

**Методы оптимизации
и их приложения**

**Труды XIII Байкальской международной школы-семинара
“Методы оптимизации и их приложения”**

ТРУДЫ

XIII Байкальской международной школы-семинара

Иркутск-Северобайкальск, 2-8 июля 2005 г.

Том 1
**Математическое
программирование**

**Иркутск
2005**

Российская академия наук
Российская ассоциация математического программирования
Институт систем энергетики им. Л.А.Мелентьева СО РАН
Иркутский государственный университет
Иркутский государственный университет путей сообщения
Иркутская государственная сельскохозяйственная академия
Институт динамики систем и теории управления СО РАН
Вычислительный центр РАН

International Association for the Promotion of Co-operation with Scientists
from the New Independent States of the Former Soviet Union (INTAS)
Российский фонд фундаментальных исследований
Иркутская областная администрация

**13-я Байкальская международная
школа-семинар**

**МЕТОДЫ ОПТИМИЗАЦИИ
И ИХ ПРИЛОЖЕНИЯ**

ТРУДЫ ШКОЛЫ-СЕМИНАРА

Том 1. Математическое программирование

2 – 8 июля 2005 г.
Иркутск, Байкал

Иркутск
2005

УДК 517.988+519.85+519.86+519.87

Математическое программирование: Труды XIII Байкальской международной школы-семинара "Методы оптимизации и их приложения", Иркутск, Байкал, 2 – 8 июля 2005 года. Том 1: Иркутск, ИСЭМ СО РАН. – 2005. – 648 с.

ISBN 5-93908-031-6.

В данном томе представлены работы, посвященные теории и методам линейного, выпуклого, нелинейного программирования, дискретной и глобальной оптимизации, многокритериальной оптимизации и теории игр, а также программам и программным комплексам для решения различных задач математического программирования.

Для научных работников, студентов и аспирантов, специализирующихся в соответствующих областях прикладной математики.

Труды подготовлены при финансовой поддержке Российского фонда фундаментальных исследований (проект 05-01-10048-г) и International Association for the Promotion of Co-operation with Scientists from the New Independent States of the Former Soviet Union (проект 04-85-832)

Ответственные за выпуск: *к.ф.-м.н. Хамисов О.В.*
к.ф.-м.н. Таурова Е.В.

ISBN 5-93908-031-6

©Институт систем энергетики
им. Л.А. Мелентьева СО РАН, 2005

Russian Academy of Sciences (RAS)
Russian Association of Mathematical Programming
Institute of Energy Systems, Siberian Branch of RAS
Irkutsk State University
Irkutsk State University of Railway Communications
Irkutsk State Agricultural Academy
Institute of System Dynamics and Control Theory, Siberian Branch of RAS
Computer Center of RAS

International Association for the Promotion of Co-operation with Scientists
from the New Independent States of the Former Soviet Union (INTAS)
Russian Foundation of Basic Research
Administration of Irkutsk Region

PROCEEDINGS OF

**13-th Baikal International
School-seminar**

**OPTIMIZATION METHODS
AND THEIR APPLICATIONS**

Volume 1. Mathematical programming

July, 2 – 8, 2005
Irkutsk, Baikal

Irkutsk
2005

Mathematical programming: Proceedings of XIII Baikal International School-seminar "Optimization methods and their applications", July, 2 – 8, Irkutsk, Baikal, 2005. Vol. 1. Irkutsk: Melentiev Energy Systems Institute SB RAS. – 2005. – 648 p.

Publication of the proceedings are supported by Russian Foundation of Basic Research (project 05-01-10048-Г) and International Association for the Promotion of Co-operation with Scientists from the New Independent States of the Former Soviet Union (project 04-85-832)

СОДЕРЖАНИЕ

ПЛЕНАРНЫЕ ДОКЛАДЫ	1
А.С. Антипин(<i>Москва</i>) К построению общей теории равновесных и игровых задач	3
V. Bulatov, T. Belykh, A. Burdukovskaja(<i>Иркутск</i>) Methods of the Chebyshev points for some problems of operations research	37
S. Dempe(<i>Germany</i>) Optimality conditions for bilevel programming problems	51
Ю.А. Кочетов, А.В.Плясунов(<i>Новосибирск</i>) Локальный поиск в комбинаторной оптимизации. Нужна ли производная?	65
СЕКЦИОННЫЕ ДОКЛАДЫ	77
ЛИНЕЙНОЕ ПРОГРАММИРОВАНИЕ	79
В.А. Березнев(<i>Москва</i>) О сложности задач параметрического линейного программирования	81
А.С. Величко(<i>Владивосток</i>) Использование точных и приближенных решений в параллельном методе прямо-двойственных отсечений для блочных задач линейного программирования	87
А.С. Герасимов, Н.К. Косовский(<i>Санкт-Петербург</i>) Алгоритм определения совместности систем линейных двучленных неравенств	93
И.И. Дикин, О.М. Попова(<i>Иркутск</i>) Контрпримеры в теории сходимости метода внутренних точек	99
В.И. Ерохин(<i>Борисоглебск</i>) Матричная коррекция несобственных задач линейного программирования по минимуму евклидовой нормы с произвольными весами и фиксированными элементами	105
В.Г. Жадан, М.В. Втюрина (<i>Москва</i>) Метод внутренней точки с наискорейшим спуском для линейной задачи дополнителности	113
В.И. Зоркальцев (<i>Иркутск</i>) Комбинированные алгоритмы внутренних точек .	119
В.И. Зоркальцев, А.Ю. Филатов(<i>Иркутск</i>) Комбинированные алгоритмы внутренних точек для задач линейного программирования	125
К.С. Кобылкин (<i>Екатеринбург</i>) Существование комитетов системы линейных неравенств	131
С.М. Пержабинский, А.Ю. Филатов(<i>Иркутск</i>) Решение систем линейных уравнений с симметричными положительно определенными матрицами методом сопряженных направлений	135
Л.Д. Попов (<i>Екатеринбург</i>) Симметрические системы и фейеровские процессы для несобственных задач линейного программирования	141
М.Ю. Хачай (<i>Екатеринбург</i>) О вычислительной сложности задачи о минимальном комитете системы линейных неравенств	147
Н.Б. Шамрай (<i>Омск</i>) Вариационный подход к решению несобственных задач линейного программирования	155
ВЫПУКЛОЕ ПРОГРАММИРОВАНИЕ	161
А.З. Ишмухаметов, Ю.Г. Карюкина (<i>Москва</i>) О некоторых конечношаговых методах оптимизации в выпуклых задачах	163
Ю.Э. Линке(<i>Иркутск</i>) О некоторых конечношаговых методах оптимизации в выпуклых задачах	169

В.Д. Скарин (<i>Екатеринбург</i>) О применении штрафных функций для коррекции несобственных задач выпуклого программирования	175
И.А. Фукин (<i>Казань</i>) Об одном алгоритме заданной точности в методе барьерных функций	181
В.И. Шмырев (<i>Новосибирск</i>) Об одной задаче кусочно-линейного выпуклого программирования	187
НЕЛИНЕЙНОЕ ПРОГРАММИРОВАНИЕ	195
А.С. Апарцин (<i>Иркутск</i>) Об одном классе минимаксных задач	197
Е.Г. Белоусов (<i>Москва</i>) Разрешимость и устойчивость задач математического программирования	203
Г.И. Забияко, Е.А. Котельников (<i>Новосибирск</i>) Примеры распараллеливания процесса решения в некоторых задачах оптимизации	209
А.В. Зыкина (<i>Омск</i>) Решение обратной нелинейной задачи дополнительности	215
Д.В. Иванов, О.В. Хамисов (<i>Иркутск</i>) Численное тестирование методов нулевого порядка в задачах невыпуклого программирования	221
О. Н. Канева (<i>Омск</i>) Двухэтапная задача стохастического программирования	231
Р.М. Ларин, Е.В. Хмель (<i>Новосибирск</i>) Решение одной задачи выбора изделий методом ветвей и границ	239
В.Г. Малинов (<i>Ульяновск</i>) Регуляризованный проекционный двухшаговый метод второго порядка с переменной метрикой для задач минимизации с ограничениями	245
М.А. Новиков (<i>Иркутск</i>) О наименьших значениях полиномов	253
Р.В. Намм, С.А. Сачков, Г. Ву (<i>Хабаровск</i>) Итерационный метод Удзавы для полуконвективной задачи Синьорини	261
Е.В. Таирова (<i>Иркутск</i>) О применении методов квадратичного программирования к нахождению допустимой точки систем квадратичных неравенств	267
А.Н. Тырсин (<i>Челябинск</i>) Устойчивая аппроксимация данных линейными регрессионными моделями	273
ГЛОБАЛЬНАЯ ОПТИМИЗАЦИЯ	279
К.А. Баркалов (<i>Нижний Новгород</i>) Новый способ учета ограничений в условиях задач глобальной оптимизации	281
V. Barsbold, Ch. Altannar and R. Enkhbat (<i>Mongolia</i>) A Global Method for solving the convex quadratic maximization problem	287
I. Vasil'ev, P. Avella (<i>Иркутск, del Sannio(Италия)</i>) Computational experiments with PBC: Parallel branch and cut library	293
О.В. Дружинина (<i>Иркутск</i>) О глобальном поиске в задаче полиэдральной отделимости	301
R. Enkhbat (<i>Mongolia</i>) On the minimization of a strongly convex function	307
А.В. Орлов (<i>Иркутск</i>) О локальном и глобальном поиске в задачах билинейного программирования	313
Е.Г. Петрова, А.С. Стрекаловский (<i>Иркутск</i>) О вариационных методах решения систем нелинейных уравнений	319
А.С. Стрекаловский (<i>Иркутск</i>) О некоторых прикладных задачах оптимизации	325
Т.В. Яковлева (<i>Иркутск</i>) О методах локального поиска для задачи с D.C. ограничением	331

МНОГОКРИТЕРИАЛЬНАЯ ОПТИМИЗАЦИЯ	337
В. А. Емеличев, К. Г. Кузьмин (<i>Минск</i>) Критерий устойчивости векторной задачи минимизации пороговых функций	339
Р.В. Ефремов, Г.К. Каменев, А.В. Лотов(<i>Москва</i>) Метод экономного описания полиэдральной аппроксимации выпуклого множества в задаче многокритериальной оптимизации	345
А.М. Леонович (<i>Минск</i>) Об устойчивости векторной траекторной задачи с обобщенными нелинейными частными критериями	353
А.И. Поспелов(<i>Москва</i>) Алгоритм аппроксимации выпуклой оболочки Эджворта - Парето в некоторых задачах целочисленной многокритериальной оптимизации	359
ТЕОРИЯ ИГР	365
С.Е. Бухтояров, В.А. Емеличев (<i>Минск</i>) Об устойчивости обобщенно-оптимальной ситуации в конечной коалиционной игре с параметрической концепцией равновесия (от лексикографической оптимальности до равновесия по Нэшу)	367
В.А. Васильев(<i>Новосибирск</i>) Информационное равновесие в модели обмена с экстерналиями	373
С.А. Осипов(<i>Запорожье</i>) Разрешение задачи дележа в кооперативной игре с помощью системы поддержки принятия решений	381
О.А. Попова(<i>Омск</i>) Равновесный подход в игровых моделях процесса обучения	385
ДИСКРЕТНАЯ ОПТИМИЗАЦИЯ	393
А.А. Агеев (<i>Новосибирск</i>) Эффективные алгоритмы для задач о размещении центров с незамкнутым и замкнутым маршрутами обслуживания	395
Е.В. Алексеева(<i>Новосибирск</i>) Поведение в среднем алгоритма локального спуска для задачи о P -медиане	401
Е.В. Алексеева, Ю.А. Кочетов, Н.А. Кочетова(<i>Новосибирск</i>) Критические параметры простейшей задачи размещения	407
С.Н. Астраков, А.И. Ерзин(<i>Кемерово, Новосибирск</i>) Моделирование взаимоотношений на графах	413
А.Е. Бабурин, Э.Х. Гимади(<i>Новосибирск</i>) Алгоритмы с оценками для некоторых обобщений задачи коммивояжера	421
А.Ф. Валеева, М.Н. Аглиуллин(<i>Уфа</i>) Алгоритм муравьиной колонии для задач двухмерной упаковки: результаты вычислительного эксперимента	429
М.А. Верхотуров, С.В. Петренко (<i>Уфа</i>) Нерегулярное размещение невыпуклых ориентированных многоугольников в односвязной невыпуклой области	435
Э.Х. Гимади, Е.Н. Гончаров(<i>Новосибирск</i>) О задаче выбора комплектации системы машин и узлов с нелинейной функцией производственных затрат	443
М.В. Девятерикова, А.А. Колоколов(<i>Омск</i>) Анализ устойчивости по целевой функции некоторых алгоритмов дискретной оптимизации	449
Г.Г. Забудский(<i>Омск</i>) О минимаксной и минисуммной задачах размещения на плоскости с запрещенными областями	455
В.В. Залюбовский(<i>Новосибирск</i>) О представлении перестановками допустимых решений одномерной задачи упаковки в контейнеры	461

Л.А. Заозерская, Е. Китринову, А.А. Колоколов (<i>Омск, Чиос(Греция)</i>) Задача оптимального размещения центров телекоммуникаций в регионе . . .	469
Д.С. Иваненко (<i>Новосибирск</i>) Гибридный алгоритм ветвей и границ с локальным поиском для смешано-целочисленной задачи размещения с ограничениями на мощности производства	477
С.Д. Иванова (<i>Омск</i>) Ширина решеточных графов специальной структуры . . .	485
В.П. Ильев (<i>Омск</i>) Оценки погрешности приближенного алгоритма для задачи о раскраске графа	491
В.С. Ишутенко, Ю.А. Кочетов (<i>Новосибирск</i>) Математическая модель оценки состава системы технических средств	497
А.А. Колоколов, А.В. Адельшин, Д.И. Ягофарова (<i>Омск</i>) Алгоритмы лексикографического перебора для решения задачи выполнимости и некоторых ее обобщений	503
А.А. Колоколов, З.Е. Нагорная, О.Н. Гуселетова, А.В. Ярош (<i>Омск</i>) Задачи дискретной оптимизации и программный комплекс для эскизного проектирования одежды	509
А.В. Кононов (<i>Новосибирск</i>) Приближенный алгоритм для задачи теории расписания с ограничением на число возобновляемых ресурсов, потребляемых одной работой	515
М.К. Кравцов, С.А. Дичковская (<i>Минск</i>) Вычислительное исследование двух приближенных алгоритмов для трехиндексной планарной проблемы выбора	521
А.А. Кузнецова, О.Н. Карпачева (<i>Ангарск</i>) Два метода локального поиска с параметрами для задачи о максимальной клике	527
Т.В. Леванова, М.А. Лореш (<i>Омск</i>) О сходимости одного алгоритма муравьиной колонии для задачи о p -медиане	535
Э.А. Мухачева, А.С. Филиппова, Ю.И. Алентьева (<i>Уфа</i>) Алгоритмы декодирования прямоугольных упаковок: краткий обзор современных подходов	543
Э. А. Мухачева, А.С. Филиппова, Д.А. Назаров, М.С. Егорова (<i>Уфа</i>) Модифицированные алгоритмы блочных декодеров в задачах прямоугольной упаковки	549
А.В. Плясунов (<i>Новосибирск</i>) Приближенный локальный поиск в задаче о P -медиане	557
А.В. Пролубников (<i>Омск</i>) Алгоритм нахождения приближенного решения задачи поиска оптимального вложения графа	563
А.Б. Рамазанов (<i>Баку</i>) Об оценке точности градиентного алгоритма в одной задаче выпуклой дискретной оптимизации и некоторые смежные вопросы	571
А.А. Романова, В.В. Сервах (<i>Омск</i>) Алгоритмы решения одной задачи построения циклического расписания	577
А.С. Талевнин (<i>Омск</i>) Приближенный вероятностный алгоритм для одной задачи аппроксимации графов	583
И.В. Уразова (<i>Омск</i>) Релаксации Лагранжа и L-структура многогранника задачи об упаковке	589
Д.В. Филимонов (<i>Омск</i>) Решение дискретной минимаксной задачи размещения с древовидной структурой связей на сети	595
P. Hansen, N. Mladenović, D. Urošević (<i>Canada, United Kingdom, Serbia and Montenegro</i>) Variable neighborhood branching for mixed integer programming	601

О.В. Хамисов (<i>Иркутск</i>) О построении новых отсечений в целочисленном линейном программировании	607
В.Н. Шевченко, А.Ю. Чирков (<i>Нижний Новгород</i>) О $(n - 1)$ -мерном приближенном решении n -мерной задачи о рюкзаке	613
ПРОГРАММЫ И ПРОГРАММНЫЕ КОМПЛЕКСЫ	619
А.Р. Ершов (<i>Иркутск</i>) Язык описания моделей системы автоматической глобальной оптимизации	621
Л.И. Крутова, С.В. Ленский (<i>Екатеринбург</i>) Экспериментальная оценка эффективности алгоритмов и программ	627
Г.А. Опарин, А.П. Новопагин (<i>Иркутск</i>) Булева модель оптимального планирования асинхронных абстрактных программ	631

ПЛЕНАРНЫЕ ДОКЛАДЫ

К ПОСТРОЕНИЮ ОБЩЕЙ ТЕОРИИ РАВНОВЕСНЫХ И ИГРОВЫХ ЗАДАЧ. 1.

А.С.Антипин

*Вычислительный Центр Российской Академии Наук. Москва
e-mail: antipin@ccas.ru*

Аннотация. В статье предпринимаются попытки очертить контуры будущей теории равновесных и игровых проблем.

Ключевые слова: равновесные задачи, игра двух лиц, неподвижные точки, равновесие по Нэшу, дефицит ресурсов, сходимость.

Введение

Равновесные и игровые задачи представляют собой математический инструментарий для описания конфликтных ситуаций постоянно возникающих в любой области повседневной жизни. Конфликт - это взаимодействие частично противоречивых факторов или интересов участников ситуации. Факторы или интересы описываются целевыми функциями, которые каждый из участников оптимизирует на своем множестве альтернатив. Найденные частные решения образуют вектор, удовлетворяющий некоторым балансовым уравнениям, характеризующим систему участников как единое целое. Последнее означает, что участники ситуации не могут независимо друг от друга выбирать свои решения, а вынуждены находить их согласованно, так чтобы удовлетворить балансовым соотношениям, которые и определяют их совокупность как единое целое в окружающей среде. Конечно, описать поведение таких систем с помощью одной задачи оптимизации невозможно, для этого требуются системы таких задач, которые известны в математике как игры n -лиц с равновесием по Нэшу. Эти системы задач оптимизации всегда можно скаляризовать и свести их к равновесным аналогам или задачам равновесного программирования, решениями которых являются неподвижные точки или просто равновесия. Перейдем к формальным постановкам и сформулируем две базовые равновесные задачи, а затем общую конструкцию, включающую в себя обе первые.

1. Постановка равновесных задач

1.1 Задача равновесного программирования представляет собой задачу вычисления неподвижной точки $v^* \in \Omega^*$ экстремального отображения [1]

$$v^* \in \operatorname{Argmin}\{\varphi(w) + \Phi(v^*, w) \mid g(w) \leq 0, w \in \Omega\}, \quad (1)$$

где $\Phi(v, w) : R^n \times R^n \rightarrow R$, $\varphi(w) : R^n \rightarrow R$, $g(w) : R^n \rightarrow R^m$, $\Omega \subset R^n$ - выпуклое замкнутое множество. переменные w и v принимают значения в одном и том же допустимом множестве $D = \{w \in R^n \mid g(w) \leq 0, w \in \Omega\}$. Предполагается, что экстремальное (маргинальное) отображение $w(v) \equiv \operatorname{Argmin}\{\varphi(w) + \Phi(v, w) \mid g(w) \leq 0, w \in \Omega\}$ переводит каждую точку допустимого множества D в выпуклое замкнутое подмножество из D . Согласно [2], такое отображение при достаточно общих условиях имеет неподвижную точку, то есть точку,

которая содержится в своем образе $v^* \in w(v^*)$, где $v^* \in D^* \subset D$; при этом пара векторов v^*, v^* всегда лежит на диагонали квадрата $D \times D$.

Как следует из (1), любое решение этой задачи удовлетворяет системе неравенств

$$\varphi(v^*) + \Phi(v^*, v^*) \leq \varphi(w) + \Phi(v^*, w), \quad g(w) \leq 0 \quad \forall w \in \Omega. \quad (2)$$

Эту систему можно рассматривать как эквивалентное определение неподвижной точки. Введем функцию (сдвиг относительно диагонали квадрата $D \times D$) $\Psi(v, w) = \varphi(w) + \Phi(v, w) - \varphi(v) - \Phi(v, v)$, которая равна нулю на диагонали квадрата $\Psi(v, w)|_{v=w} = 0 \quad \forall v \in D$. Используя $\Psi(v, w)$, задачу (1) представим в виде

$$\Psi(v^*, w) \geq 0, \quad g(w) \leq 0 \quad \forall w \in \Omega. \quad (3)$$

В [3] показано, что существует вектор $v^* \in D$ который является решением (3).

С содержательной точки зрения равновесную задачу (1) можно трактовать как игру активного участника стратегии которого описываются переменной $w \in \Omega$ против неопределенности Природы постоянные изменения которой характеризуются параметром $v \in \Omega$. В равновесной ситуации игрок не заинтересован изменять свое состояние, в противном случае он попадет в область неопределенности с непредсказуемым собственным поведением, при этом его выигрыш может измениться только в худшую сторону.

Целевая функция задачи (1) представляет собой сумму функций одна из которых зависит от одной переменной, а вторая от двух переменных. Такая структура целевой функции связана с тем, что функция $\varphi(w)$ согласно представлениям теории принятия решений описывает собственные предпочтения лица принимающего решения при выборе наилучшей альтернативы на собственном множестве альтернатив. Эти предпочтения не зависят от изменения внешних условий в которых приходится принимать решение. Функция $\Phi(v, w)$, наоборот описывает воздействие внешних условий на систему предпочтений лица принимающего решение. В изменяющейся обстановке игрок должен найти такое решение, которое устойчиво по отношению к изменяющимся условиям с одной стороны и отражает личные предпочтения участника с другой.

Равновесная задача (1) включает в себя, в частности, важный случай игры двух лиц (и естественно, n -лиц) с ненулевой суммой. Рассмотрим этот случай более детально. Для удобства рассмотрения выпишем постановку задачи в форме (2)

$$\varphi(v^*) + \Phi(v^*, v^*) \leq \varphi(w) + \Phi(v^*, w), \quad G(w) \leq 0 \quad \forall w \in \Omega. \quad (4)$$

Пусть функции и множество, формирующие эту задачу имеют следующее представление $\Phi(v, w) = f_1(z_1, x_2) + f_2(x_1, z_2)$, $\varphi(w) = \varphi_1(z_1) + \varphi_2(z_2)$, $G(w) = (g_1(z_1), g_2(z_2))$, где $G(w)$ -вектор-столбец и $v = (x_1, x_2)$, $w = (z_1, z_2)$, $\Omega = X_1 \times X_2$, тогда задачу можно переписать как

$$\varphi_1(x_1^*) + \varphi_2(x_2^*) + f_1(x_1^*, x_2^*) + f_2(x_1^*, x_2^*) \leq \varphi_1(z_1) + \varphi_2(z_2) + f_1(z_1, x_2^*) + f_2(x_1^*, z_2),$$

$$g_1(z_1) \leq 0, \quad g_2(z_2) \leq 0, \quad z_1 \in X_1, \quad z_2 \in X_2.$$

Целевая функция полученной задачи имеет сепарабельную структуру относительно переменных z_1 и z_2 , а множество $\{z_1 | g_1(z_1) \leq 0, z_1 \in X_1\} \times \{z_2 | g_2(z_2) \leq 0, z_2 \in X_2\}$ представляет собой параллелепипед. В силу этих свойств задача распадается на систему двух независимых подзадач оптимизации, причем каждая определена в своем пространстве

$$x_1^* \in \text{Argmin}\{\varphi_1(z_1) + f_1(z_1, x_2^*) \mid g_1(z_1) \leq 0, \quad z_1 \in X_1\},$$

$$x_2^* \in \operatorname{Argmin}\{\varphi_2(z_2) + f_2(x_1^*, z_2) \mid g_2(z_2) \leq 0, z_2 \in X_2\}. \quad (5)$$

Задача (5) представляет собой игру двух лиц с функциональными ограничениями. Частный случай этой задачи без функциональных ограничений был сформулирован Нэшем в [4]

$$x_1^* \in \operatorname{Argmin}\{f_1(z_1, x_2^*) \mid z_1 \in X_1\},$$

$$x_2^* \in \operatorname{Argmin}\{f_2(x_1^*, z_2) \mid z_2 \in X_2\}.$$

В игре (5) $\{x_1 \mid g_1(z_1) \leq 0, z_1 \in X_1\}$, $\{x_2 \mid g_2(z_2) \leq 0, z_2 \in X_2\}$ интерпретируются как множества стратегий первого и второго игроков, где $z_i \in X_i, i = 1, 2$ отдельные стратегии игроков, а $\varphi_1(z_1) + f_1(z_1, x_2), \varphi_2(z_2) + f_2(x_1, z_2)$ рассматриваются как их платежные функции. Содержательный смысл неподвижной точки x^* задачи (5) состоит в том, что ни один из игроков не заинтересован нарушать это состояние равновесия, поскольку никто из них не может в одностороннем порядке уменьшить значение своей платежной функции. Любое решение игры (5) будем называть также равновесным решением или равновесием. Это решение, в частности, означает состояние компромисса с суммарным выигрышем $\varphi_1(x_2^*) + f_1(x_1^*, x_2^*) + \varphi_2(x_2^*) + f_2(x_1^*, x_2^*)$.

Среди игр двух лиц наиболее изученными являются игры с нулевой суммой. Эти игры подчинены условию $\varphi_1(x_1) + f_1(x_1, x_2) + \varphi_2(x_2) + f_2(x_1, x_2) = 0, g_1(x_1) \leq 0, x_1 \in X_1, g_2(x_2) \leq 0, x_2 \in X_2$. Выполнение указанного условия возможно только в случае, если $f_1(x_1, x_2) = -f_2(x_1, x_2)$ и $\varphi_1(x_1) = \varphi_2(x_2) = 0, g_1(x_1) \leq 0, x_1 \in X_1, g_2(x_2) \leq 0, x_2 \in X_2$. Очевидно, тогда задачу можно представить как

$$f(x_1^*, z_2) \leq f(x_1^*, x_2^*) \leq f(z_1, x_2^*) \quad \forall z_1, z_2 \in X_1 \times X_2,$$

где $f(x_1, x_2) = f_1(x_1, x_2) = -f_2(x_1, x_2)$. Отсюда следует, что пара x_1^*, x_2^* является седловой точкой функции $f(x_1, x_2)$ на множестве $g_1(x_1) \leq 0, x_1 \in X_1, g_2(x_2) \leq 0, x_2 \in X_2$.

В заключении этого раздела приведем экономическую интерпретацию равновесной задачи (1) [5]. Будем предполагать, что эта задача описывает взаимодействие двух участников: один из которых "планирующий орган", а другой - "производство". Пусть переменная $v \in \Omega$ описывает набор альтернатив первого участника, а переменная $w \in \Omega$ - множество альтернатив второго. Альтернативы первого участника интерпретируются как планы для производства, которые выбирает планирующий орган. Ответ производства на выбранный план заключается в выборе такого уровня интенсивности $w \in \Omega$ работы предприятия, который минимизирует производственные расходы (издержки). В общем случае, интересы планирующего органа и производства не совпадают. Их согласование в контексте рассматриваемой ситуации состоит в совпадении планов и интенсивностей работы производства, т.е. $v^* = w^*$. В дальнейшем, содержательный смысл согласования интересов участников ситуации будет уточнен введением функции сдвига $\Psi(v, w) = \varphi(w) + \Phi(v, w) - \varphi(v) + \Phi(v, v), v, w \in \Omega \times \Omega$ для которой точка v^*, w^* является седловой, а v^* - равновесное решение (1). В этом случае интересы "планирующего органа" сводятся к максимизации планов интенсивности работы производства, т.е. функции $\Psi(v, w)$ по v , а интересы предприятия в минимизации издержек производства, т.е. функции $\Psi(v, w)$ по w .

1.2 Равновесная задача дефицита ресурсов представляет собой задачу вычисления неподвижной точки $y^* \in R_+^m$ экстремального отображения, порожденного функцией оптимального значения или функцией чувствительности, определенной на множестве ресурсов

и функцией (отображением) стоимости этих ресурсов. Опишем ситуацию более детально. Функция оптимального значения (чувствительности) хорошо известна и многократно обсуждалась в литературе [6]. Для задачи выпуклого программирования она имеет вид

$$f(y) = \min\{\varphi(w) \mid g(w) \leq y, w \in \Omega\}, \quad (6)$$

где $\varphi(w) : R^n \rightarrow R$, $g(w) : R^n \rightarrow R^m$ - выпуклые функции, $\Omega \subset R^n$ - выпуклое замкнутое множество. Допустимое множество этой задачи $D(y) = \{w \in \Omega \mid g(w) \leq y, w \in \Omega\}$ зависит от параметра $y \geq 0$. При фиксированном $y \geq 0$ имеем задачу выпуклого программирования. Функция Лагранжа этой задачи имеет вид $\mathcal{L}(w, p) = \varphi(w) + \langle p, g(w) - y \rangle$, $w \in \Omega, p \in R_+^m$. Предполагается, что при каждом значении y задача (6) имеет регулярный характер и, следовательно, существует седловая точка $w(y), p(y)$ функции Лагранжа этой задачи.

Задача выпуклого программирования обычно интерпретируется как модель производства в которой требуется выбрать вектор интенсивности работы предприятия w так чтобы обеспечить выпуск продукции с минимальными расходами ресурсов, которые оцениваются величиной $f(y)$ из (6). Интуитивно ясно, что если объемы ресурсов (компоненты вектора y) неограниченно растут, то их дефицитность (производственная нехватка) падает, соответственно, внутренние оценки обусловленности по Л.В. Канторовичу (отрицательные множители Лагранжа) тоже падают, потребление ресурсов в этом случае ничто не ограничивает и поэтому расходы предприятия по выпуску продукции стремятся к нулю. Эта идеальная ситуация неограниченного объема ресурсов в реальности не встречается. Ресурсы - это, как правило, продукция некоторого производства или сырье стоимость которого определяется расходами на восстановление нарушенной Природы, причем чем больше требуется ресурсов для производства, тем выше их стоимость. Таким образом, наряду с функцией дефицитности ресурсов $f(y)$, убывающей с ростом y имеется другая функция стоимости ресурсов $r(y)$, возрастающей с увеличением y . Интуитивно ясно, что существует равновесный вектор ресурсов $y^* \geq 0$, такой что $f(y^*) = r(y^*)$. Это равенство означает, что дефицитность (нехватка) ресурсов для выпуска продукции согласована с их внешней (рыночной) стоимостью. Другими словами, нехватка ресурсов согласована с их наличием для нужд производства.

Здесь равновесные объемы ресурсов (вектор y^*) определяются как точка пересечения скалярных "кривых" $f(y)$ и $r(y)$. Во многих случаях может оказаться более предпочтительным другой способ вычислить равновесные объемы ресурсов, а именно, как точку пересечения векторных функций $p(y)$ и $s(y)$, где $p(y)$ - градиент (субградиент) выпуклой функции чувствительности $f(y)$, а $s(y)$ - возрастающая по компонентно векторная функция, которая, в частности, может быть градиентом (субградиентом) функции стоимости ресурсов $r(y)$. Размерность вектора $s(y)$ совпадает с размерностью двойственных переменных функции Лагранжа задачи выпуклого программирования (6). Это дает нам возможность сформулировать равновесную задачу дефицита ресурсов в следующем виде

$$\begin{aligned} w(y), p(y) \in \text{Argsdl}\{\varphi(w) + \langle p, g(w) - s(y) \rangle \mid w \in \Omega, p \in R_+^m\}, \\ -p(y) = s(y), \end{aligned} \quad (7)$$

где $s(y)$ -априори заданное отображение, а $w(y), p(y)$ - седловая точка функции,

$$\mathcal{L}(w, p, s(y)) = \varphi(w) + \langle p, g(w) - s(y) \rangle \quad \forall w \in \Omega, p \in R_+^m$$

при фиксированном значении параметра $y \in R_+^m$.

Пусть $y = y^*$ решение (7), тогда эту задачу можно переформулировать в следующем виде: требуется вычислить вектор $y^* \geq 0$, который является решением седлового включения

$$w(y^*), s(y^*) \in \text{Argsdl}\{\varphi(w) + \langle p, g(w) - s(y^*) \rangle \mid w \in \Omega, p \in R_+^m\}. \quad (8)$$

где $w(y^*), p(y^*)$ -седловая точка функции

$$\mathcal{L}(w, p, s) = \varphi(w) + \langle p, g(w) - s(y^*) \rangle \quad \forall w \in \Omega, \quad p \in R_+^m$$

при этом $-p^* = s^*$ при фиксированном значении $y = y^*$. Ниже будет показано, что функция $\mathcal{L}(w, p, s)$ является функцией Лагранжа задачи выпуклого программирования

$$w(y) \in \text{Argmin}\{\varphi(w) \mid g(w) \leq s(y), \quad w \in \Omega\}$$

Напомним, что вектор $p(y)$ в (7) является градиентом (субградиентом) выпуклой функции чувствительности $f(y)$ [7],[8] а вектор $s(y)$ является в частности градиентом (субградиентом) некоторой выпуклой функции стоимости ресурсов $r(y)$. Таким образом, можно утверждать, что в равновесной задаче (8) требуется выбрать равновесный вектор ресурсов $y = y^*$ так чтобы его внутренняя и внешняя маргинальные оценки (цены) совпали

$$-p(y^*) = s(y^*).$$

С другой стороны, из условия дополняющей нежесткости выше приведенной задачи выпуклого программирования следует, что равновесное решение y^* является точкой пересечения двух скалярных "кривых" $\langle s(y), g(w(y)) \rangle$ и $|s(y)|^2$.

Типы векторных функций $s(y)$ могут быть весьма многообразны, например, $s(y) = y$, $s(y) = \langle Ay, y \rangle$, где $A \geq 0$ -симметричная матрица. Компоненты вектора $s(y)$ могут быть представлены хорошо известными производственными функциями. В частности, векторная функция $s(y)$ может быть функцией множителей Лагранжа другой задачи выпуклого программирования. Вектор $s(y)$ как это уже было сделано выше в дальнейшем для краткости будем также называть вектором ресурсов.

Отметим книгу ([7], стр.240), где кратко обсуждаются вариационные неравенства, решения которых можно представить как неподвижные точки экстремальных и седловых отображений.

1.3 Общая равновесная задача представляет собой задачу отыскания неподвижной точки седлового отображения вида

$$\begin{aligned} w(v, y), p(v, y) \in \text{Argsdl}\{\varphi(w) + \Phi(v, w) + \langle p, g(w) - s(y) \rangle \mid w \in \Omega, p \in R_+^m\}, \\ w(v, y) = v, \quad p(v, y) = s(y), \end{aligned} \quad (9)$$

где $w(v, y), p(v, y)$ -седловая точка функции

$$\mathcal{L}(v, w) = \varphi(w) + \Phi(v, w) + \langle p, g(w) - s(y) \rangle \quad \forall w \in \Omega, \quad p \in R_+^m$$

при фиксированных значениях параметров $v \in \Omega$ и $y \in R_+^m$. Седловое отображение из правой части (9) представляет собой выпуклозначное замкнутое отображение образ которого состоит из седловых точек выпукло-вогнутой седловой функции переменных $w \in \Omega$, $p \in R_+^m$, зависящей от параметров $v \in \Omega$, $y \in R_+^m$. При некоторых требованиях

непрерывности отображения и компактности его области определения согласно теореме Какутани всегда существует неподвижная точка такого отображения, т.е. решение задачи (9).

Можно показать, что в этом случае компоненты вектора p являются множителями Лагранжа задачи равновесного программирования относительно переменной $w \in \Omega$

$$w(v, y) \in \operatorname{Argmin}\{\varphi(w) + \Phi(v, w) \mid g(w) \leq s(y), w \in \Omega\} \quad (10)$$

при фиксированном значении параметра $y \in R_+^m$.

2. Теория равновесных задач

2.1 Положительная полуопределенность Введем некоторые понятия необходимые для развития теории равновесных задач. Понятие положительной полуопределенности дает возможность выделить класс функций седлового типа [9].

О п р е д е л е н и е 1 . *Функцию $\Phi(v, w)$ из $R^n \times R^n$ в R^1 назовем положительно полуопределенной на $\Omega \times \Omega$, если она удовлетворяет неравенству*

$$\Phi(w, w) - \Phi(w, v) - \Phi(v, w) + \Phi(v, v) \geq 0 \quad \forall w, v \in \Omega \times \Omega. \quad (11)$$

Класс положительно полуопределенных функций не пуст, так как он включает в себя все антисимметричные функции вида

$$\Phi(v, w) = -\Phi(w, v) \quad \forall v, w \in \Omega \times \Omega. \quad (12)$$

В случае, если $\Phi(v, w) = \langle \Phi v, w \rangle$ - билинейная функция, где Φ - квадратная матрица, то неравенство (11) принимает вид хорошо известного условия положительной полуопределенности матрицы Φ , т.е. $\langle \Phi w, w \rangle - \langle \Phi w, v \rangle - \langle \Phi v, w \rangle + \langle \Phi v, v \rangle = \langle \Phi(w - v), w - v \rangle \geq 0 \quad \forall (w - v) \in \Omega$.

Нетрудно видеть, что условие антисимметрии фактически описывает функции седлового типа. Убедимся, что условие положительной полуопределенности также можно трактовать как расширение условия антисимметрии. Действительно, введем функцию сдвига $\Psi(v, w) = \Phi(v, w) - \Phi(v, v)$, где $\Psi(v, v) = 0 \quad \forall v \in \Omega$ и представим (11) в виде

$$\Psi(w, v) + \Psi(v, w) \leq 0 \quad \forall w, v \in \Omega. \quad (13)$$

Введем определение. Если точка v^* удовлетворяет неравенству

$$\Psi(v^*, v^*) \leq \Psi(v^*, w) \quad \forall w \in \Omega, \quad (14)$$

то будем называть ее равновесным решением этого неравенства. Из (13) при $v = v^*$ имеем $\Psi(w, v^*) \leq 0 \quad \forall w \in \Omega$. Отсюда $\Psi(v, v^*) \leq \Psi(v^*, v^*) \leq \Psi(v^*, w) \quad \forall v, w \in \Omega$. Теперь можно сформулировать ключевое свойство положительной полуопределенности

С в о й с т в о 1 . *Если v^* равновесное решение или неподвижная точка экстремального отображения, порожденного правой частью неравенства (14), то пара (v^*, v^*) является седловой точкой для функции сдвига $\Psi(v, w)$.*

Это свойство оправдывает введение нового понятия (нелинейной) положительной полуопределенности. Ниже будет показано, что понятие положительной полуопределенности в исследовании седловых задач играет такую же роль как понятие выпуклости в задачах оптимизации.

Напомним, что выпуклая собственная функция $f(v)$ субдифференцируема в каждой точке области определения, при этом субдифференциал $\partial f(v)$ представляет собой непустое выпуклое замкнутое множество опорных функционалов $\nabla f(v)$. Любой из них называется субградиентом и удовлетворяет условию

$$\langle \nabla f(v), w - v \rangle \leq f(w) - f(v) \leq \langle \nabla f(w), w - v \rangle \quad \forall w, v \in \Omega. \quad (15)$$

Если функция $\Phi(v, w)$ выпуклая по w для любого v , то применяя последовательно к (11) левое и правое неравенства из (15), получим

$$\langle \nabla_2 \Phi(w, w) - \nabla_2 \Phi(v, v), w - v \rangle \geq 0 \quad \forall w, v \in \Omega, \quad (16)$$

С в о й с т в о 2 . Градиент-сужение $\nabla_2 \Phi(v, w)|_{v=w}$ положительно полуопределенной, выпуклой по w для любого v функции $\Phi(v, w)$ - монотонный оператор.

2.2 Бидифференциал и его свойства В функциональных пространствах линейные функции играют особую роль, их часто рассматривают как касательные плоскости к графикам любых функций, а затем нередко сравнивают поведение этих функций относительно их линейных приближений. В пространствах бифункций аналогичную роль берут на себя билинейные функции или седловые поверхности [10].

О п р е д е л е н и е 2 . Функцию $\Phi(v, w)$ назовем бидифференцируемой в точке $v, w \in \Omega \times \Omega$, если существует квадратная матрица $D\Phi(v, w)$, такая что

$$\{\Phi(v+h, w+k) - \Phi(v+h, w)\} - \{\Phi(v, w+k) - \Phi(v, w)\} = \langle D\Phi(v, w)h, k \rangle + \omega(v, w, h, k), \quad (17)$$

где $\omega(v, w, h, k)/|h||k| \rightarrow 0$ при $|h|, |k| \rightarrow 0$ для всех $h \in \mathbb{R}^n, k \in \mathbb{R}^n$.

Билинейную функцию $\langle D\Phi(v, w)h, k \rangle$ назовем билинейным дифференциалом функции $\Phi(v, w)$ в точке $v, w \in \Omega \times \Omega$.

Функцию $\Phi(v, w)$ назовем би-дифференцируемой на квадрате $\Omega \times \Omega$, если она дифференцируема во всех точках этого квадрата.

Формально введенный дифференциал имеет простой геометрический смысл и дает возможность оценить отклонение функции по первой переменной двух приращений по второй (двойное приращение), а именно, $\{\Phi(v+h, v+k) - \Phi(v+h, v)\}$ и $\{\Phi(v, v+k) - \Phi(v, v)\}$ при переходе из точки v, v в точку $v+h, v+k$. Таким образом, введенный бидифференциал порожден разностью по переменной v двух приращений по переменной w . Однако, эта разность может быть представлена как разность двух средних значений функции $\Phi(v, w)$ на главной и поперечной диагоналях прямоугольника приращений

$$\begin{aligned} (1/2)(\Phi(v+h, w+k) + \Phi(v, w)) - (1/2)(\Phi(v+h, w) + \Phi(v, w+k)) &= \\ &= \langle (1/2)D\Phi(v, w)h, k \rangle + (1/2)\omega(v, w, h, k). \end{aligned}$$

Тогда введенный дифференциал можно трактовать как билинейную аппроксимацию приращений средних значений.

Нетрудно проверить, что если функция $\Phi(v, w) = \varphi(v) + \varphi(w)$ сепарабельна относительно своих переменных, то бидифференциал такой функции равен нулю, т.е. $D\Phi(v, v) \equiv 0 \forall v \in \Omega$.

Билинейный дифференциал представляет собой седловую касательную поверхность и описывает некоторые особенности поведения двойного приращения функции $\Phi(v, w)$ по отношению к этому дифференциалу.

Возникает вопрос, как соотносится введенный билинейный дифференциал с классическим в случае дифференцируемости $\Phi(v, w)$? В [10] показано, что если $\Phi(v, w)$ дважды непрерывно дифференцируемая функция, то матрица $D\Phi(v, v)$ совпадает с сужением матрицы смешанных производных на главную диагональ квадрата $\frac{\partial^2 \Phi(v, w)}{\partial w \partial v} |_{v=w}$ второго дифференциала и там же установлено свойство.

С в о й с т в о 3 . Матрица бидифференциала нормализованной функции седловой задачи является антисимметричной.

2.3 Симметричность и антисимметричность матрицы бидифференциала. Убедимся, что матрица бидифференциала $D\Phi(v, v)$, определенная на диагонали квадрата $\Omega \times \Omega$ является симметричной, если функция $\Phi(v, w)$ симметричная, т.е. подчинена условию

$$\Phi(v, w) = \Phi(w, v) \quad \forall v, w \in \Omega \times \Omega \quad (18)$$

и антисимметричной, если $\Phi(v, w)$ антисимметричная функция. Действительно, пусть $\Phi(v, w)$ – симметричная функция, рассмотрим еще раз представление (17)

$$\begin{aligned} & \Phi(v + \varepsilon h, v + \varepsilon k) - \Phi(v + \varepsilon h, v) - \Phi(v, v + \varepsilon k) + \Phi(v, v) = \\ & = \varepsilon^2 \langle D\Phi(v, v)h, k \rangle + \omega(v, \varepsilon h, \varepsilon k) \end{aligned} \quad (19)$$

для всех $v + \varepsilon h \in \Omega$, $v + \varepsilon k \in \Omega$, $\varepsilon \geq 0$. Поменяем местами переменные, тогда

$$\begin{aligned} & \Phi(v + \varepsilon k, v + \varepsilon h) - \Phi(v, v + \varepsilon h) - \Phi(v + \varepsilon k, v) + \Phi(v, v) = \\ & = \varepsilon^2 \langle D^T \Phi(v, v)k, h \rangle + \omega(v, \varepsilon k, \varepsilon h). \end{aligned} \quad (20)$$

В силу симметрии левые части (19) и (20) равны, следовательно, равны их правые части

$$\langle D\Phi(v, v)h, k \rangle = \langle D^T \Phi(v, v)h, k \rangle + (\omega(v, \varepsilon h, \varepsilon k) - \omega(v, \varepsilon k, \varepsilon h)) / \varepsilon^2.$$

Отсюда, при $\varepsilon \rightarrow 0$ получим

$$D\Phi(v, v) = D^T \Phi(v, v) \quad \forall v \in \Omega.$$

Теперь, пусть функция $\Phi(v, w)$ - антисимметрична, т.е. выполнено условие (12). Тогда сложим неравенства (19) и (20). В силу условия антисимметрии, получим

$$0 = \langle (D\Phi(v, v) + D^T \Phi(v, v))h, k \rangle + (\omega(v, \varepsilon h, \varepsilon k) + \omega(v, \varepsilon k, \varepsilon h)) / \varepsilon^2.$$

Отсюда, при $\varepsilon \rightarrow 0$ имеем

$$D\Phi(v, v) = -D^T \Phi(v, v) \quad \forall v \in \Omega.$$

Из последнего равенства, в частности, следует $\langle D\Phi(v, v)h, h \rangle = 0 \quad \forall h \in \mathbb{R}^n$.

С в о й с т в о 4 . Если функция $\Phi(v, w)$ симметричная, то $D\Phi(v, v)$ - симметричная матрица для всех $v \in \Omega$, если же $\Phi(v, w)$ антисимметрична, то $D\Phi(v, v)$ - антисимметричная матрица для всех $v \in \Omega$.

Обратное утверждение, вообще говоря, не верно. Действительно, пусть $\Phi(v, w)$ - симметричная функция, тогда функция вида $\Phi(v, w) + \varphi(v)$ не является симметричной (однако $\Phi(v, w) + \varphi(v) + \varphi(w)$ - симметричная функция), но смешанные производные, т.е. матрицы бидифференциалов у этих функций совпадают.

2.4 Положительная полуопределенность матрицы бидифференциала. В классическом анализе идея производной часто используется для того чтобы понятие выпуклости выразить через касательную плоскость. Эта мысль находит свое выражение в системе хорошо известных неравенств, которые в сущности и составляют основной инструментарий выпуклого анализа.

Попробуем развить аналогичный подход для характеристики понятия положительной полуопределенности.

О п р е д е л е н и е 3 Функцию $\Phi(v, w)$ назовем:

- 1) сильно положительно полуопределенной,
 - 2) положительно полуопределенной,
 - 3) липшицево положительно полуопределенной,
- если при $h = k$ в разложении бидифференциала (17)

$$\{\Phi(v + h, w + h) - \Phi(v + h, w)\} - \{\Phi(v, w + h) - \Phi(v, w)\} = \langle D\Phi(v, w)h, h \rangle + \omega(v, w, h, h),$$

правая часть этого разложения удовлетворяет последовательно условиям

- 1) $\langle D\Phi(v, w)h, h \rangle + \omega(v, w, h, h) \geq \mu|h|^2$,
 - 2) $\langle D\Phi(v, w)h, h \rangle + \omega(v, w, h, h) \geq 0$,
 - 3) $|\langle D\Phi(v, w)h, h \rangle + \omega(v, w, h, h)| \leq L|h|^2$
- для любого фиксированного v и всех $h \in R^n$.

Из условия 2) сразу же следует определение 1. Отметим, что условие 2) носит более общий характер, чем неравенство (11) из определения 1), поскольку неравенство (11) определено для точек расположенных симметрично относительно диагонали квадрата, в то время как условия 2) справедливы для любых точек из $\Omega \times \Omega$

Покажем, что свойства функции $\Phi(v, w)$ определяются в основном свойствами матрицы бидифференциала, причем присутствие величины $\omega(v, v, h, h)$ третьего порядка малости не отражается на этом взаимоотношении.

Действительно, пусть функция $\Phi(v, w)$ положительно полуопределенная (сильно положительно полуопределенна), тогда из (17) при $v = w$ и $h = k$, где приращение имеет вид εh , получим

$$\begin{aligned} (1/\varepsilon^2)(\Phi(v + \varepsilon h, v + \varepsilon h) - \Phi(v + \varepsilon h, v) - \Phi(v, v + \varepsilon h) + \Phi(v, v)) = \\ = \langle D\Phi(v, v)h, h \rangle + \omega(v, v, \varepsilon h, \varepsilon h)/\varepsilon^2 \geq 0 \quad (\geq \mu|h|^2) \quad \forall h \in R^n. \end{aligned}$$

Отсюда, при $\varepsilon \rightarrow 0$ следует, что $\langle D\Phi(v, v)h, h \rangle \geq 0$ ($\langle D\Phi(v, v)h, h \rangle \geq \mu|h|^2$) $\forall h \in R^n$ и $v \in \Omega$.

С в о й с т в о 5 . Если функция $\Phi(v, w)$ положительно полуопределенная (сильно положительно полуопределенная), то матрица бидифференциала $D\Phi(v, v) \geq 0$ -положительная полуопределенная ($D\Phi(v, v) \geq \mu|h|^2$ -сильно положительно полуопределенная) для всех $v \in \Omega$.

Пусть матрица $D\Phi(v, w)$ бидифференциала (17) положительно полуопределенна (сильно положительно определена) в каждой точке некоторой окрестности диагонали квадрата $\Omega \times \Omega$ Тогда, для фиксированной точки v, v и для фиксированного значения вектора h выпишем формулу о конечном приращении

$$\Phi(v + h, v + h) - \Phi(v + h, v) - \Phi(v, v + h) + \Phi(v, v) = \langle D\Phi(v, v + \theta h)h, h \rangle \geq 0, \quad (21)$$

где $0 \leq \theta \leq 1$ - некоторое число. Отсюда имеем

С в о й с т в о 6 . Если матрица $D\Phi(v, w) \geq 0$ для всех v, w из некоторой окрестности диагонали квадрата $\Omega \times \Omega$, тогда $\Phi(v, w)$ положительно полуопределенная (сильно положительно определенная) функция.

Однако матрица бидифференциала $D\Phi(v, w)$ может не быть положительно полуопределенной в некоторой окрестности диагонали квадрате $\Omega \times \Omega$, но она может быть сильно положительно определенной на его диагонали. Этого свойства также достаточно для положительной определенности функции $\Phi(v, w)$.

Действительно из разложения

$$\begin{aligned} (1/\varepsilon^2)(\Phi(v + \varepsilon h, v + \varepsilon h) - \Phi(v + \varepsilon h, v) - \Phi(v, v + \varepsilon h) + \Phi(v, v)) = \\ \geq (\mu|h|^2 + (1/\varepsilon^2)\omega(v, v, \varepsilon h, \varepsilon h)), \end{aligned}$$

где h фиксированный элемент. При $\varepsilon \rightarrow 0$ всегда найдется такое значение $\varepsilon_0 > 0$, что для всех $\varepsilon_0 > \varepsilon$ величина $\mu|h|^2 + \omega(v, v, \varepsilon h, \varepsilon h)/\varepsilon^2 \geq 0$. Следовательно, неравенство (11) выполняется.

С в о й с т в о 7 . Если матрица бидифференциала сильно положительно полуопределенная для всех $v \in \Omega$, тогда функция $\Phi(v, w)$ положительно полуопределенная.

3. Расщепление равновесных задач

3.1 Симметричные равновесные задачи. В предыдущих разделах отдельно были рассмотрены классы симметричных, антисимметричных и положительно полуопределенных бифункций. В соответствие с этим разделением рассмотрим отдельно классы симметричных, антисимметричных и положительно полуопределенных равновесных задач. Для простоты рассуждений функциональные ограничения задачи опустим.

Итак, рассмотрим равновесную задачу без функциональных ограничений

$$\varphi(v^*) + \Phi(v^*, v^*) \leq \varphi(w) + \Phi(v^*, w) \quad \forall w \in \Omega. \quad (22)$$

Сначала будем предполагать, что целевая функция $\Phi(v, w)$ - симметричная, т.е. подчинена условию (18). С учетом симметрии (22) можно представить в виде

$$\varphi(v^*) + \Phi(v^*, v^*) \leq \varphi(w) + \Phi(w, v^*) \quad \forall w \in \Omega.$$

Сопоставляя оба неравенства, видим, что точка v^* является минимумом функции $\varphi(w) + \Phi(v, w)$ причем в качестве переменной оптимизации можно использовать любую переменную функции $\Phi(v, w)$.

Установим главные свойства симметричных равновесных задач, а именно, покажем, что эти равновесные задачи сводятся к задачам оптимизации. Пусть $\Phi(v, w)$ симметричная, положительно полуопределенная функция, тогда из (11) имеем

$$\Phi(w, w) - 2\Phi(v, w) + \Phi(v, v) \geq 0 \quad \forall w, v \in \Omega \times \Omega. \quad (23)$$

При $v = v^*$ получим

$$2\Phi(v^*, w) \leq \Phi(w, w) + \Phi(v^*, v^*) \quad \forall w \in \Omega. \quad (24)$$

Сопоставляя это неравенство с (22), имеем

$$2(\varphi(v^*) - \varphi(w)) + 2\Phi(v^*, v^*) \leq \Phi(w, w) + \Phi(v^*, v^*) \quad \forall w \in \Omega.$$

Отсюда

$$\varphi(v^*) + (1/2)\Phi(v^*, v^*) \leq \varphi(w) + (1/2)\Phi(w, w) \quad \forall w \in \Omega. \quad (25)$$

Следовательно, можно сформулировать

С в о й с т в о 8 Если функция $\Phi(v, w)$ симметричная и положительно полуопределенная, то равновесная задача (22) сводится к задаче оптимизации вида

$$v^* \in \text{Argmin}\{\varphi(w) + (1/2)\Phi(w, w) \mid w \in \Omega\}. \quad (26)$$

Это значит, что если v^* - равновесное решение (22), то пара v^*, v^* - минимум функции $2\varphi(w) + \Phi(w, w)$ на диагонали квадрата $\Omega \times \Omega$, при этом ни откуда не следует, что эта функция выпуклая, хотя функция $\varphi(w) + (1/2)\Phi(v^*, w)$ выпуклая по w для любого v .

3.2 Антисимметричные равновесные задачи. Покажем, что равновесная задача с антисимметричной функцией $\Phi(v, w)$ сводится к седловой задаче. Введем функцию сдвига относительно диагонали квадрата $\Omega \times \Omega$, т.е.

$$\begin{aligned} \Psi(v, w) &= \varphi(w) + \Phi(v, w) - \varphi(v) - \Phi(v, v) = \\ &= \varphi(w) - \varphi(v) + \Phi(v, w) \quad \forall v, w \in \Omega, \end{aligned}$$

где $\Phi(v, v) = 0 \quad \forall v \in \Omega$. В терминах введенной функции переформулируем исходную равновесную задачу в виде

$$0 = \Psi(v^*, v^*) \leq \Psi(v^*, w) \quad \forall w \in D. \quad (27)$$

Эквивалентность (22) и (27) очевидна, поскольку их целевые функции отличаются только на константу $\varphi(v^*)$.

Покажем, что v^*, v^* является седловой точкой для $\Psi(v, w)$ на $\Omega \times \Omega$. Действительно, из (27) имеем

$$0 = \varphi(v^*) - \varphi(v^*) + \Phi(v^*, v^*) \leq \varphi(w) - \varphi(v^*) + \Phi(v^*, w) \quad \forall w \in \Omega.$$

Умножим полученное неравенство на минус единицу, тогда

$$0 \geq -\varphi(w) + \varphi(v^*) - \Phi(v^*, w) = -\varphi(w) + \varphi(v^*) + \Phi(w, v^*) \quad \forall w \in \Omega.$$

Перепишем полученное неравенство в форме

$$\Psi(v, v^*) \leq 0 \quad \forall v \in \Omega.$$

Сопоставляя это неравенство с (27), получим

$$\Psi(v, v^*) \leq \Psi(v^*, v^*) \leq \Psi(v^*, w) \quad \forall v, w \in \Omega \times \Omega. \quad (28)$$

Из (28) следует

С в о й с т в о 9 Если v^* равновесное решение задачи (22), и функция $\Phi(v, w)$ антисимметричная, то точка v^*, v^* является седловой для функции сдвига $\Psi(v, w) = \varphi(w) - \varphi(v) + \Phi(v, w)$.

3.3 Положительно полуопределенные равновесные задачи. В рассмотренных выше случаях функция $\Phi(v, w)$ была либо симметричной, либо антисимметричной, но в обоих примерах она была положительно полуопределенной. Теперь рассмотрим случай, когда функция $\Phi(v, w)$ только положительно полуопределенная, при этом не предполагаем никакой симметрии. Используя функцию сдвига $\Psi(v, w) = \varphi(w) + \Phi(v, w) - \varphi(v) - \Phi(v, v)$ представим задачу (22) в виде

$$0 \leq \Psi(v^*, w) \quad \forall w \in \Omega. \quad (29)$$

Из неравенства положительной полуопределенности (11) при $v = v^*$ имеем

$$\Phi(w, w) - \Phi(w, v^*) \geq \Phi(v^*, w) - \Phi(v^*, v^*) \quad \forall w \in \Omega.$$

Сопоставим полученное неравенство с (29), тогда

$$\Phi(w, w) - \Phi(w, v^*) = \Psi(w, v^*) \geq 0 \quad \forall w \in \Omega. \quad (30)$$

Сравнивая (29) и (30), получим

$$\Psi(v, v^*) \leq \Psi(v^*, v^*) \leq \Psi(v^*, w) \quad \forall v, w \in \Omega \times \Omega. \quad (31)$$

Таким образом, можно утверждать

С в о й с т в о 10 Если v^* равновесное решение задачи (22), и функция $\Phi(v, w)$ положительно полуопределенная, то точка v^*, v^* является седловой для функции сдвига $\Psi(v, w) = \varphi(w) + \Phi(v, w) - \varphi(v) - \Phi(v, v)$.

При этом отметим, что если функция $\Phi(v, w)$ была выпуклой по w для любого v , то и функция $\Psi(v, w)$ остается выпуклой по w для любого v .

3.4 Расщепление задач. Рассмотрим важный вопрос, что происходит с равновесной задачей (22) при разложении ее целевой функции базисе симметричных и антисимметричных функций пространства бифункций. Бифункция $\Phi(v, w)$ - эта вещественная функция двух переменных одинаковой размерности $v, w \in \Omega \times \Omega$, где $\Omega \in R^n$ - конечномерное евклидовое пространство. В частности, $\Omega \times \Omega$ может совпадать с $R^n \times R^n$. Совокупность всех бифункций образует линейное пространство с обычными операциями сложения двух вещественных функций и умножения их на скаляр.

В пространстве бифункций выделим два базисных подпространства симметричных и антисимметричных функций. Эти подпространства определяются следующими условиями:

$$\Phi(v, w) - \Phi(w, v) = 0 \quad \forall w \in \Omega, \quad \forall v \in \Omega, \quad (32)$$

и

$$\Phi(v, w) + \Phi(w, v) = 0 \quad \forall w \in \Omega, \quad \forall v \in \Omega. \quad (33)$$

Введенные подпространства можно рассматривать как систему "координат" в которой любая бифункция может быть разложена в системе базисных пространств. Действительно, пара точек с координатами w, v и v, w расположена симметрично относительно диагонали квадрата $\Omega \times \Omega$, т.е. относительно линейного многообразия $v = w$. Это дает возможность ввести понятие транспонированной функции [11]. Если каждой точке с координатами v, w поставить в соответствие значение функции $\Phi(., .)$ вычисленной в точке w, v , т.е. $v, w \rightarrow \Phi(w, v)$, то получим транспонированную функцию $\Phi^T(v, w) = \Phi(w, v)$. В терминах этой функции условия симметричности (32) и антисимметричности (33) имеют вид

$$\Phi(v, w) = \Phi^T(v, w), \quad \Phi(v, w) = -\Phi^T(v, w).$$

Используя очевидные соотношения $\Phi(v, w) = (\Phi^T(v, w))^T$, $(\Phi_1(v, w) + \Phi_2(v, w))^T = \Phi_1^T(v, w) + \Phi_2^T(v, w)$, нетрудно проверить, что любую вещественную функцию $\Phi(v, w)$ всегда можно представить в виде суммы

$$\Phi(v, w) = S(v, w) + K(v, w), \quad (34)$$

где функция $S(v, w)$ - симметричная, а $K(v, w)$ - антисимметричная. Это разложение единственно, причем

$$S(v, w) = \frac{1}{2} (\Phi(v, w) + \Phi^T(v, w)), \quad K(v, w) = \frac{1}{2} (\Phi(v, w) - \Phi^T(v, w)). \quad (35)$$

Функции $S(v, w)$ и $K(v, w)$ можно трактовать как "координаты" разложения $\Phi(v, w)$ в базисе симметричных и антисимметричных функций.

Согласно формулам (34), (35) условие положительной полуопределенности (11) можем записать как

$$S(w, w) + K(w, w) - S(w, v) - K(w, v) - S(v, w) - K(v, w) + S(v, v) + K(v, v) \geq 0$$

для всех $w, v \in \Omega \times \Omega$.

Учитывая (32), (33), отсюда имеем

$$S(w, w) - 2S(w, v) + S(v, v) \geq 0 \quad \forall w, v \in \Omega \times \Omega. \quad (36)$$

Таким образом, можно утверждать следующее

С в о й с т в о 11 Если функция $\Phi(v, w)$ положительно полуопределенная, то ее симметрическая часть $S(v, w)$ в разложении (34), так же положительно полуопределенная функция.

Положим $v = v^*$ в (36), тогда

$$S(w, v^*) \leq (1/2)S(w, w) + (1/2)S(v^*, v^*) \quad \forall w \in \Omega. \quad (37)$$

Используя полученное неравенство, оценим аналогичную величину в (22), тогда

$$\varphi(v^*) + (1/2)S(v^*, v^*) + K(v^*, v^*) \leq \varphi(w) + (1/2)S(w, w) + K(v^*, w) \quad \forall w \in \Omega. \quad (38)$$

Введем обозначение для $\varphi'(v, w) = \varphi(w) + (1/2)S(w, w)$ и запишем (38) в форме

$$\varphi'(v^*) + K(v^*, v^*) \leq \varphi'(w) + K(v^*, w) \quad \forall w \in \Omega, \quad (39)$$

Функции $\varphi(w) + \Phi(v, w)$ и $\varphi'(w) + K(v, w)$ - это очень разные функции, например, если первая из них выпуклая по w для любого v , то вторая функция, вообще говоря, не будет выпуклой, но тем не менее точка v^* будет равновесной для задачи (25). Следовательно имеем

С в о й с т в о 12 Положительно полуопределенную равновесную задачу (22) с точкой равновесия v^* всегда можно привести к антисимметричной равновесной задаче вида (39) для которой v^* также является точкой равновесия.

Таким образом, равновесная точка является инвариантом при преобразовании исходной положительно полуопределенной равновесной задачи к ее антисимметричной форме.

Покажем, что в состоянии равновесия задачи, порожденные функциями $\varphi(w) + \Phi(v, w)$ и $\varphi'(w) + K(v, w)$, имеют одно и тоже необходимое условие. Действительно, дифференцируя (34) по w , получим

$$\nabla_2 \Phi(v, w) = \nabla_2 S(v, w) + \nabla_2 K(v, w) \quad \forall v, w \in \Omega \times \Omega.$$

Используя условие $2\nabla_2 S(v, w)|_{v=w} = \nabla S(v, v)$, которое справедливо для симметричных функций [11]

$$\nabla_2 \Phi(v, w) = (1/2)\nabla S(v, v) + \nabla_2 K(v, w) \quad \forall v, w \in \Omega \times \Omega.$$

Отсюда, нетрудно видеть что в равновесном состоянии необходимые условия минимума у обеих задач (22) и (39) совпадают и имеют вид

$$\langle \nabla \varphi(v^*) + \nabla_2 S(v^*, v^*) + \nabla_2 K(v^*, v^*), w - v^* \rangle \geq 0 \quad \forall w \in \Omega$$

или

$$\langle \nabla \varphi(v^*) + (1/2)\nabla S(v^*, v^*) + \nabla_2 K(v^*, v^*), w - v^* \rangle \geq 0 \quad \forall w \in \Omega \quad (40)$$

Последнее означает, что в случае, если функция $\Phi(v, w)$ не является положительно полуопределенной, а функция $\Phi'(v, w) = \varphi(w) + (1/2)S(w, w) + K(v, w)$ при этом является выпуклой по w для любого v , то задачу (22) всегда можно привести к положительно полуопределенной форме (39). Если учесть, что выпуклость функции $\Phi'(v, w)$ по w можно получить всегда за счет выбора регуляризирующего члена $\varphi(w)$, то имеем

С в о й с т в о 13 Если функция $\Phi'(v, w) = \varphi(w) + (1/2)S(w, w) + K(v, w)$ выпукла по w для любого v , то неположительно полуопределенную равновесную задачу (22) можно привести к положительно полуопределенному виду (39), при этом оператор $\nabla_2 \Phi(v, w)|_{v=w}$ согласно (16) будет монотонным.

Последнее обстоятельство лежит в основе различных примеров неположительно полуопределенных равновесных задач [12], которые порождают вариационные неравенства со свойством монотонности. На основе этих примеров утверждается, что класс положительно полуопределенных задач более узкий, чем соответствующий им класс вариационных

неравенств. В действительности, в случае дифференцируемости $\Phi(v, w)$ по w они эквивалентны.

3.5 Комплексная структура равновесных задач. Представление $\Phi(v, w)$ в базисе симметричных и антисимметричных функций $S(v, w)$ и $K(v, w)$ порождает расщепление равновесной задачи на сумму двух задач одна из которых задача оптимизации, а другая - седловая задача. Очевидно, что такое расщепление равновесной задачи имеет структуру комплексного числа: задача оптимизации соответствует вещественной части комплексного числа, а седловая задача его мнимой части. Это особенно наглядно проявляется для билинейных равновесных задач, где задача оптимизации порождается симметричной матрицей с чисто вещественным спектром, а седловая задача порождается антисимметричной матрицей с чисто мнимым спектром. В этом контексте седловую поверхность с антисимметричной матрицей можно рассматривать как графический образ мнимой единицы. В этом представлении положительно полуопределенная функция $\Phi(v, w)$ соответствует комплексному числу лежащему в правой полуплоскости, т.е. в полуплоскости с положительной вещественной частью. Понятие положительной полуопределенности, таким образом, можно трактовать как нелинейное обобщение идеи положительного числа на нелинейные бифункции.

3.6 Связь выпуклости и положительной полуопределенности. Понятие положительной полуопределенности тесно связано с понятием выпуклости, для этого достаточно сопоставить соответствующие неравенства

$$\frac{\Phi(w, v) + \Phi(v, w)}{2} \leq \frac{1}{2} \Phi(w, w) + \frac{1}{2} \Phi(v, v),$$

и

$$\Phi\left(\frac{w+v}{2}, \frac{v+w}{2}\right) \leq \frac{1}{2} \Phi(w, w) + \frac{1}{2} \Phi(v, v).$$

Если функция $\Phi(v, w) = \langle \Phi v, w \rangle$ билинейна, где матрица Φ - симметрична и положительно полуопределена, то в классе этих функций оба неравенства эквивалентны.

В нелинейном случае положительная полуопределенность гарантирует выпуклость функции $\Phi(v, w)$ на главной диагонали, при условии, что функция выпукла на поперечной диагонали. Действительно, из выпуклости и положительной полуопределенности мы имеем

$$\Phi\left(\frac{v+w}{2}, \frac{w+v}{2}\right) \leq \frac{1}{2} \Phi(v, w) + \frac{1}{2} \Phi(w, v) \leq \frac{1}{2} \Phi(v, v) + \frac{1}{2} \Phi(w, w).$$

Отсюда

$$\Phi\left(\frac{v+w}{2}, \frac{v+w}{2}\right) \leq \frac{1}{2} \Phi(v, v) + \frac{1}{2} \Phi(w, w).$$

Утверждение доказано.

С в о й с т в о 14 Если функция $\Phi(v, w)$ выпукла на любой поперечной диагонали, т.е. для всех $w = -v + c$, где $c \in R^1$, то она выпукла и на главной диагонали, т.е. при всех $w = v$

Это утверждение дополняет условия симметрии и антисимметрии функции $\Phi(v, w)$, которые формулируются в точках, лежащих на поперечных диагоналях.

Представляет несомненный интерес утверждение, которое гарантирует выпуклость функции $\Phi(v, w)$ на главной диагонали при условии, что она выпукла по w при любого v .

Итак, для некоторой точки $v' = \frac{v+w}{2}$ условие выпуклости функции $\Phi(v, w)$ по переменной w запишем как

$$\Phi\left(\frac{v+w}{2}, \frac{v+w}{2}\right) \leq \frac{1}{2} \Phi\left(\frac{v+w}{2}, v\right) + \frac{1}{2} \Phi\left(\frac{v+w}{2}, w\right) \quad \forall v, w \in \Omega \times \Omega. \quad (41)$$

С другой стороны условия положительной полуопределенности функции $\Phi(v, w)$ в вершинах квадратов $(\frac{v+w}{2}, \frac{v+w}{2}); (\frac{v+w}{2}, v); (v, \frac{v+w}{2}); (v, v)$ и $(w, w); (w, \frac{v+w}{2}); (\frac{v+w}{2}, w); (\frac{v+w}{2}, \frac{v+w}{2})$ имеют вид

$$\Phi\left(\frac{v+w}{2}, \frac{v+w}{2}\right) - \Phi\left(\frac{v+w}{2}, v\right) - \Phi\left(v, \frac{v+w}{2}\right) + \Phi(v, v) \geq 0, \quad (42)$$

$$\Phi(w, w) - \Phi\left(w, \frac{v+w}{2}\right) - \Phi\left(\frac{v+w}{2}, w\right) + \Phi\left(\frac{v+w}{2}, \frac{v+w}{2}\right) \geq 0. \quad (43)$$

Предполагая, что $\Phi(v, w)$ симметричная функций, отсюда имеем

$$2\Phi\left(\frac{v+w}{2}, v\right) \leq \Phi\left(\frac{v+w}{2}, \frac{v+w}{2}\right) + \Phi(v, v),$$

$$2\Phi\left(\frac{v+w}{2}, w\right) \leq \Phi(w, w) + \Phi\left(\frac{v+w}{2}, \frac{v+w}{2}\right).$$

Сопоставляя полученные оценки с (41), имеем

$$\Phi\left(\frac{v+w}{2}, \frac{v+w}{2}\right) \leq \frac{1}{2} \Phi(v, v) + \frac{1}{2} \Phi(w, w) \quad \forall v, w \in \Omega \times \Omega. \quad (44)$$

С в о й с т в о 15 Если симметричная функция $\Phi(v, w)$ выпукла по w для любого v , то она выпукла на главной диагонали.

Доказанное свойство следует также из неравенства (16) в случае дифференцируемости $\Phi(v, w)$ по w и при условии, что эта функция симметричная. Однако, неравенство (16) справедливо не только для симметричных функций, поэтому возникает вопрос о выпуклости общих функций на главной диагонали без условия их симметрии. Выясним, когда это возможно. Для этого нам придется существенно расширить понятия выпуклости, для несимметричных, в общем случае, функций $\Phi(v, w)$.

О п р е д е л е н и е 4 . Функцию $\Phi(v, w)$ назовем выпуклой на ортогональных сечениях или просто на сечениях, если она удовлетворяет неравенству

$$\Phi\left(\frac{v+w}{2}, \frac{v+w}{2}\right) \leq \frac{1}{4} \left(\Phi\left(\frac{v+w}{2}, v\right) + \Phi\left(\frac{v+w}{2}, w\right) \right) + \frac{1}{4} \left(\Phi\left(v, \frac{v+w}{2}\right) + \Phi\left(w, \frac{v+w}{2}\right) \right) \quad (45)$$

для всех $w, v \in \Omega \times \Omega$.

В частности, если функция $\Phi(v, w)$ симметричная, то из (45) немедленно следует условие обычной выпуклости (44), более того, если $\Phi(v, w)$ выпуклая на сечениях в смысле (41), то она выпукла в смысле (45), поэтому последнее условие можно трактовать как расширение понятия выпуклости на несимметричный случай.

Покажем, что в общем случае несимметричная функция выпуклая на сечениях будет выпуклой на главной диагонали. Из (42),(43) мы имеем

$$\Phi\left(\frac{v+w}{2}, v\right) + \Phi\left(v, \frac{v+w}{2}\right) \leq \Phi\left(\frac{v+w}{2}, \frac{v+w}{2}\right) + \Phi(v, v),$$

$$\Phi(w, \frac{v+w}{2}) + \Phi(\frac{v+w}{2}, w) \leq \Phi(w, w) + \Phi(\frac{v+w}{2}, \frac{v+w}{2}).$$

Подставляя левые части этих неравенств в (45), получим выпуклость функции $\Phi(v, v)$

$$\Phi(\frac{v+w}{2}, \frac{v+w}{2}) \leq \frac{1}{2} \Phi(v, v) + \frac{1}{2} \Phi(w, w) \quad \forall v, w \in \Omega \times \Omega.$$

С в о й с т в о 16 Если функция $\Phi(v, w)$ выпуклая на сечениях, то она выпуклая на главной диагонали.

3.7 Симметризация и антисимметризация равновесных задач. Ранее, было отмечено, что разложение функции $\Phi(v, w)$ в базисе симметричных и антисимметричных функций порождает расщепление равновесной задачи на две ее проекции - задачу оптимизации и седловую задачу. Обе эти задачи определены в одном и том же пространстве и поэтому естественно предполагать, что при некоторых ограничениях решение исходной задачи будет являться и решением задач-проекций. Этот факт может оказаться полезным для построения методов решения равновесных задач, потому что обе задачи обладают дополнительными свойствами и являются более простыми. Итак, рассмотрим простейшую равновесную задачу

$$\varphi(v^*) + \Phi(v^*, v^*) \leq \varphi(w) + \Phi(v^*, w) \quad \forall w \in \Omega. \quad (46)$$

В силу разложения (34), эту задачу можно представить как

$$\varphi(v^*) + S(v^*, v^*) + K(v^*, v^*) \leq \varphi(w) + S(v^*, w) + K(v^*, w) \quad \forall w \in \Omega. \quad (47)$$

Учитывая, что $K(v^*, v^*) = 0$, дополнительно потребуем, чтобы область изменения переменных w была подчинена ограничению $K(v^*, w) \leq 0$, тогда последнее неравенство можно представить как систему неравенств

$$\varphi(v^*) + S(v^*, v^*) \leq \varphi(w) + S(v^*, w), \quad K(v^*, w) \leq 0 \quad w \in \Omega.$$

Полученная система определяет собой симметричную равновесную задачу со связанным антисимметричным ограничением. Ранее было показано, что симметричные равновесные задачи могут быть сведены к задачам оптимизации вида

$$v^* \in \text{Argmin}\{\varphi(w) + (1/2)S(w, w) \mid K(v^*, w) \leq 0, \quad w \in \Omega. \quad (48)$$

Причем, если функция $\Phi(v, w)$ была положительно полуопределенной, то как следует из (26) точка (v^*, v^*) есть минимум (48), если v^* -равновесие для (46). Если же функция $\Phi(v, w)$ не была положительно полуопределенной, то как следует из (40) равновесное решение задачи (46) удовлетворяет необходимому условию минимума задачи оптимизации (48).

Заметим, что функциональное ограничение в задаче (48) порождено антисимметричной функцией, допустимое множество которой всегда содержит диагональ квадрата, т.е. прямую " $v = w$ ". Именно, на этой прямой определена целевая функция и, именно, на этой прямой лежит равновесная точка. В этой ситуации функциональное ограничение задачи становится лишним и его нужно просто отбросить. Тогда исходная задача (46) фактически сведется к ее симметризации

$$\varphi(v^*) + (1/2)(\Phi(v^*, v^*) + \Phi^T(v^*, v^*)) \leq \varphi(w) + (1/2)(\Phi(v^*, w) + \Phi^T(v^*, w)) \quad \forall w \in \Omega.$$

В невыпуклом и в неположительно полуопределенном случае этот факт выглядит очень привлекательным с точки зрения сведения общих равновесных задач к симметричным (т.е. оптимизационным) задачам. Если $w \in \Omega$ подчинено условию $\Phi^T(v^*, w) - \Phi^T(v^*, v^*) \leq 0$, то v^* - решение симметричной задачи

$$\varphi(v^*) + (1/2)(\Phi(v^*, v^*)) \leq \varphi(w) + (1/2)(\Phi(v^*, w)), \quad \Phi^T(v^*, w) - \Phi^T(v^*, v^*) \leq 0 \quad \forall w \in \Omega.$$

В зависимости от распределения свойств выпуклости среди симметричной и антисимметричной компонент целевой функции исходной равновесной задачи, последнюю может оказаться полезным преобразовать не к задаче оптимизации, а к седловой задаче. Действительно, из (47) имеем

$$\varphi(v^*) + K(v^*, v^*) \leq \varphi(w) + K(v^*, w) + (S(v^*, w) - S(v^*, v^*)), \quad w \in \Omega.$$

Потребуем дополнительно, чтобы переменные w были подчинены ограничению вида $S(v^*, w) - S(v^*, v^*) \leq 0$, тогда задача сводится к системе

$$\varphi(v^*) + K(v^*, v^*) \leq \varphi(w) + K(v^*, w), \quad S(v^*, w) - S(v^*, v^*) \leq 0, \quad w \in \Omega, \quad (49)$$

которая представляет собой антисимметричную равновесную задачу с симметричными связанными ограничениями. Как было показано ранее такая задача является седловой задачей. Действительно, в силу антисимметрии можем написать

$$\varphi(w) + K(w, w) - \varphi(v^*) - K(w, v^*) - \varphi(w) - K(v^*, w) + \varphi(v^*) + K(v^*, v^*) = 0.$$

Сопоставляя полученное неравенство с (49), имеем

$$\varphi(w) + K(w, w) - \varphi(v^*) - K(w, v^*) \geq 0. \quad (50)$$

Ограничение задачи (49) в силу симметрии запишем как $S(v, v^*) - S(v^*, v^*) \leq 0$, $w \in \Omega$. Используя обозначения $\Psi(v, w) = \varphi(w) - K(v, w) + \varphi(v) + K(v, v)$, систему неравенств (49) и (50) можем представить в виде

$$\Psi(v, v^*) \leq \Psi(v^*, v^*) \leq \Psi(v^*, w)$$

где

$$S(v, v^*) - S(v^*, v^*) \leq 0, \quad S(v^*, w) - S(v^*, v^*) \leq 0, \quad v \in \Omega, \quad w \in \Omega,$$

т.е. получена седловая задача с симметричными функциональными ограничениями. Во многих случаях эта задача может оказаться относительно простой для ее решения. Идея расщепления равновесных задач никак не связана с идеей выпуклости, последнее означает, что любую не выпуклую (в каком либо смысле) равновесную задачу всегда можно свести к одной из ее проекций, которая существенно проще исходной задачи, и найти решение задачи-проекции, которое может оказаться решением исходной задачи.

4. Выпуклые положительно полуопределенные равновесные задачи

4.1 Прямая и дуальная задачи. Напомним, что равновесную задачу

$$\varphi(v^*) + \Phi(v^*, v^*) \leq \varphi(w) + \Phi(v^*, w) \quad \forall w \in \Omega, \quad (51)$$

где $\Phi(v, w)$ положительно полуопределенная (в частности, антисимметричная) функция, будем называть положительно полуопределенной равновесной задачей. Если дополнительно функция $\varphi(w) + \Phi(v, w)$ выпукла по w для любого v , то такую задачу будем называть выпуклой положительно полуопределенной или для краткости просто выпуклой равновесной задачей.

Введем понятие дуальной равновесной задачи. Используя свойство положительной полуопределенности, положим $v = v^*$ в (11), тогда

$$\begin{aligned} \varphi(w) + \Phi(w, w) - \varphi(v^*) - \Phi(w, v^*) &\geq \\ &\geq \varphi(w) + \Phi(v^*, w) - \varphi(v^*) - \Phi(v^*, v^*) \quad \forall w \in \Omega. \end{aligned} \quad (52)$$

Сопоставляя (51) и (52), получим

$$\varphi(v) + \Phi(v, v) - \varphi(v^*) - \Phi(v, v^*) \geq 0 \quad \forall v \in \Omega. \quad (53)$$

Наряду с равновесной задачей (51), которую будем называть прямой, введем еще одну равновесную задачу, которую назовем дуальной [13],[12]: в этой задаче требуется определить вектор v^* такой что условие (53) выполнено, т.е.

$$v^* \in \operatorname{Argmax}\{\varphi(v^*) + \Phi(v, v^*) - \varphi(v) - \Phi(v, v) \mid v \in \Omega\}. \quad (54)$$

Из (52) и (53) следует

С в о й с т в о 17 Множество решений прямой задачи (51) вложено в множество решений дуальной задачи (53).

В общем случае дуальная задача может быть сформулирована не зависимо от прямой и возникает вопрос о том, как связаны решения этих двух задач.

4.2 Седловая постановка. Прямая задача (51) в терминах функции сдвига $\Psi(v, w) = \varphi(w) + \Phi(v, w) - \varphi(v) - \Phi(v, v)$ как уже отмечалось в (29) имеет вид

$$0 \leq \Psi(v^*, w) \quad \forall w \in \Omega. \quad (55)$$

Дуальная задач (53) с помощью этой же функции может быть представлена как

$$\Psi(v, v^*) \leq 0 \quad \forall v \in \Omega. \quad (56)$$

Очевидно, что обе задачи сводятся к решению системы [1]

$$\Psi(v, v^*) \leq \Psi(v^*, v^*) \leq \Psi(v^*, w) \quad \forall v, w \in \Omega \times \Omega \quad (57)$$

из которой следует важное свойство.

С в о й с т в о 18 Если v^* равновесное решение задачи (51), функция $\Phi(v, w)$ положительно полуопределенная, то точка v^*, v^* является седловой для функции сдвига $\Psi(v, w) = \varphi(w) + \Phi(v, w) - \varphi(v) - \Phi(v, v)$.

Отметим при этом, что функция $\Psi(v, w)$ выпукла по w и не является вогнутой по v .

Установим факт обратный к сформулированному утверждению, а именно, одна из компонент седловой точки функции сдвига всегда является неподвижной точкой. Действительно, пусть v^*, w^* седловая точка функции сдвига $\Psi(v, w)$

$$\Psi(v, w^*) \leq \Psi(v^*, w^*) \leq \Psi(v^*, w) \quad \forall v, w \in \Omega \times \Omega.$$

Положим в этом неравенстве $w = v^*, v = w^*$, тогда получим $\Psi(v^*, w^*) = 0$. Правое неравенство выше приведенной системы перепишем в форме

$$0 \leq \varphi(w) + \Phi(v^*, w) - \varphi(v^*) - \Phi(v^*, v^*) \quad \forall w \in \Omega.$$

Отсюда имеем

$$\varphi(v^*) + \Phi(v^*, v^*) \leq \varphi(w) + \Phi(v^*, w) \quad \forall w \in \Omega.$$

Таким образом, можно сформулировать следующее

С в о й с т в о 19 Если v^*, w^* седловая точка функции сдвига $\Psi(v, w)$, то v^* - неподвижная точка задачи (51).

Два последних утверждения можно рассматривать как равновесный аналог теоремы Куна-Таккера из оптимизации.

4.3 Свойства выпуклости множества равновесных решений. Покажем, что множество решений выпуклой положительно полуопределенной равновесной задачи представляет собой выпуклое замкнутое множество. Предполагая выпуклость функции $\Psi(v, w)$ по $w \in \Omega$ для любого $v \in \Omega$ покажем, что множество решений дуальной задачи всегда выпукло и замкнуто. Действительно, пусть v_1^* и v_2^* решения дуальной задачи (56), т.е.

$$\Psi(v, v_i^*) \leq 0 \quad \forall v \in \Omega, \quad i = 1, 2.$$

Убедимся, что любая точка отрезка $v^*(\alpha) = \alpha v_1^* + (1 - \alpha)v_2^*$ также является решением дуальной задачи. Действительно, с учетом выпуклости $\Psi(v, w)$ по w , имеем

$$\begin{aligned} \Psi(v, v^*(\alpha)) &= \Psi(v, \alpha v_1^* + (1 - \alpha)v_2^*) \leq \\ &\alpha \Psi(v, v_1^*) + (1 - \alpha)\Psi(v, v_2^*) \leq 0, \end{aligned}$$

т.е.

$$\Psi(v, v^*(\alpha)) \leq 0 \quad \forall v \in \Omega.$$

Таким образом установлено

С в о й с т в о 20 Множество решений дуальной задачи (56) выпукло, замкнуто.

Покажем, что любое решение дуальной задачи является решением прямой [12]. Пусть v^* - решение дуальной задачи, т.е. $\Psi(v, v^*) - \Psi(v, v) \leq 0 \quad \forall v \in \Omega$. Тогда, если $v(\alpha) = \alpha v^* + (1 - \alpha)v$, то

$$\Psi(v(\alpha), v^*) \leq \Psi(v(\alpha), v(\alpha)).$$

Отсюда

$$\Psi(v(\alpha), v^*) \leq \Psi(v(\alpha), \alpha v^* + (1 - \alpha)v) \leq$$

$$\leq \alpha \Psi(v(\alpha), v^*) + (1 - \alpha) \Psi(v(\alpha), v),$$

или

$$(1 - \alpha) \Psi(v(\alpha), v^*) \leq (1 - \alpha) \Psi(v(\alpha), v).$$

При $\alpha \rightarrow 1$ с учетом непрерывности функции $\Psi(v, w)$, получим

$$\Psi(v^*, v^*) \leq \Psi(v^*, v) \quad \forall v \in \Omega,$$

т.е. v^* - решение прямой задачи (51). Таким образом установлено

С в о й с т в о 21 Множество решений дуальной задачи (56) вложено в множество решений прямой задачи (55).

Из свойств 17 и 21 следует, что множества решений прямой и дуальной задач совпадают. Из этого факта, в частности, вытекает

С в о й с т в о 22 Множество неподвижных точек прямой задачи (51) является выпуклым и замкнутым.

5. Методы решения равновесных задач

5.1 Экстрапроксимальный метод для решения задачи равновесного программирования. Рассмотрим экстрапроксимальный подход для задачи равновесного программирования

$$v^* \in \operatorname{Argmin}\{\varphi(w) + \Phi(v^*, w) \mid g(w) \leq 0, w \in \Omega\}, \quad (58)$$

где при фиксированном значении параметра v мы имеем задачу выпуклого программирования относительно переменной w .

Обозначим допустимое множество этой задачи как $D = \{w \in R^n \mid g(w) \leq 0, w \in \Omega\}$, тогда задачу (58) можем представить в виде неравенства

$$\varphi(v^*) + \Phi(v^*, v^*) \leq \varphi(w) + \Phi(v^*, w) \quad \forall w \in D. \quad (59)$$

Используя структуру этого неравенства введем функцию сдвига

$$\Psi(v, w) = \varphi(w) + \Phi(v, w) - \varphi(v) - \Phi(v, v) \quad \forall v, w \in D \times D.$$

Нетрудно видеть, что на диагонали квадрата $D \times D$, т.е. при $v = w$ эта функция тождественно равна нулю: $\Psi(v, v) = 0$. В терминах введенной функции равновесная задача (59) принимает вид

$$\Psi(v^*, v^*) \leq \Psi(v^*, w) \quad \forall w \in D.$$

Введем

О п р е д е л е н и е 5 . Равновесное решение $v^* \in D$ задачи (59) или (58) назовем решением седлового типа, если пара v^*, v^* является седловой точкой функции сдвига

$$\Psi(v, v^*) \leq \Psi(v^*, v^*) \leq \Psi(v^*, w) \quad \forall v, w \in D \times D. \quad (60)$$

Введенное свойство является ключевым для обоснования сходимости большинства итеративных и непрерывных методов решения равновесных задач. Это свойство является конструктивным, поскольку его проверка сводится к доказательству теоремы существования седловой точки функции сдвига.

Имея в виду дальнейшие ссылки выпишем левое неравенство этой системы отдельно

$$\varphi(v^*) + \Phi(v, v^*) - \varphi(v) - \Phi(v, v) \leq 0 \quad \forall v \in D. \quad (61)$$

Отметим, что функция $\Psi(v, w)$ здесь выпуклая по w и, вообще говоря, не является вогнутой по v . Равновесные задачи решения которых подчинены этому условию составляют довольно обширный класс прикладных задач. Покажем, что введенные ранее положительно полуопределенные функции (11) порождают равновесные задачи решениями которых являются неподвижные точки седлового типа в смысле (60). Положим $v = v^*$ в (11) и представим это неравенство в виде

$$\begin{aligned} & \varphi(w) + \Phi(w, w) - \varphi(v^*) - \Phi(w, v^*) \geq \\ & \geq \varphi(w) + \Phi(v^*, w) - \varphi(v^*) - \Phi(v^*, v^*) \quad \forall w \in D. \end{aligned}$$

Сопоставляя полученное неравенство с (59), имеем (61). Используя функцию сдвига, неравенства (61), (59) в свою очередь можно записать в виде системы неравенств (60). Другими словами, любое решение положительно полуопределенной равновесной задачи является равновесием седлового типа. Положительно полуопределенные равновесные задачи представляют собой подкласс равновесных задач с седловой структурой решений.

Рассмотрим равновесную задачу (58) принимая во внимание ее функциональные ограничения, которые теперь мы будем учитывать с помощью функции Лагранжа. Функция Лагранжа этой задачи, зависящая от параметра v , имеет вид

$$\mathcal{L}(v, w, p) = \varphi(w) + \Phi(v, w) + \langle p, g(w) \rangle \quad \forall w \in \Omega, p \geq 0. \quad (62)$$

Будем предполагать, что в равновесном состоянии, т.е. при $v = v^*$ функция $L(v^*, w, p)$ имеет седловую точку

$$\mathcal{L}(v^*, w^*, p) \leq \mathcal{L}(v^*, w^*, p^*) \leq \mathcal{L}(v^*, w, p^*) \quad \forall w \in \Omega, p \geq 0. \quad (63)$$

Введенная функция Лагранжа дает возможность учитывать функциональные ограничения задачи в итеративных формулах методов не используя операцию проектирования, которая в общем случае может оказаться весьма трудоемкой. Однако, условие для равновесного решения быть точкой седлового типа должно быть сформулировано с учетом функциональных ограничений, которые в этом случае будут вовлечены в вычислительный процесс. Сформулируем аналог условия (60) для равновесных задач с функциональными ограничениями.

О п р е д е л е н и е 6 . *Равновесное решение $v^* \in \Omega$ задачи (58) назовем решением седлового типа, если пара v^*, v^* является седловой точкой функции сдвига*

$$\Psi_0(v, w, p) = \Psi(v, w) + \langle p, g(w) - g(v) \rangle, \quad v, w \in \Omega, p \geq 0$$

при $p = p^*$, т.е.

$$\begin{aligned} & \Psi(v, v^*) + \langle p^*, g(v^*) - g(v) \rangle \leq \\ & \leq \Psi(v^*, v^*) + \langle p^*, g(v^*) - g(v^*) \rangle \leq \\ & \leq \Psi(v^*, w) + \langle p^*, g(w) - g(v^*) \rangle \quad \forall v, w \in \Omega \times \Omega. \end{aligned} \quad (64)$$

Левое неравенство этой системы выпишем отдельно

$$\Psi(v, v^*) + \langle p^*, g(v^*) - g(v) \rangle \leq 0. \quad (65)$$

Вернемся к функции Лагранжа и систему неравенств (63) представим в виде проксимальных уравнений

$$\begin{aligned} v^* &\in \operatorname{Argmin}\{\varphi(w) + \Phi(v^*, w) + \langle p^*, g(w) \rangle \mid w \in \Omega\}, \\ p^* &\in \operatorname{Argmax}\{\langle p, g(v^*) \rangle \mid p \geq 0\}, \end{aligned} \quad (66)$$

Последнее включение в полученной системе можно переписать также в эквивалентной форме операторного уравнения

$$\begin{aligned} v^* &\in \operatorname{Argmin}\{\varphi(w) + \Phi(v^*, w) + \langle p^*, g(w) \rangle \mid w \in \Omega\}, \\ p^* &= \pi_+(p^* + \alpha g(v^*)), \end{aligned} \quad (67)$$

где $\pi(\dots)$ - оператор проектирования некоторого вектора на положительный ортант R_+^n , $\alpha > 0$ -параметр типа длины шага.

Для решения системы (67) используем экстрапроксимальный метод [5]. Предполагая, что приближение v^n, p^n найдено, вычислим следующую итерацию по формулам первый полушаг (прогнозный)

$$\bar{p}^n = \pi_+(p^n + \alpha g(v^n)),$$

$$\bar{v}^n \in \operatorname{Argmin}\{1/2|w - v^n|^2 + \alpha(\mathcal{L}(v^n, w, \bar{p}^n) \mid w \in \Omega\},$$

второй полушаг (основной)

$$p^{n+1} = \pi_+(p^n + \alpha g(\bar{v}^n)),$$

$$v^{n+1} \in \operatorname{Argmin}\{1/2|w - v^n|^2 + \alpha(\mathcal{L}(\bar{v}^n, w, \bar{p}^n) \mid w \in \Omega\}, \quad (68)$$

где $\mathcal{L}(v, w, p)$ - функция Лагранжа (62). Длина шага α в процессе (68) определяется из условия $0 < \alpha < \alpha_0$, где константа α_0 будет определена ниже. Для задач равновесного программирования этот подход был исследован в [5],[14].

Для доказательства сходимости представим процесс в виде неравенств, которые являются эквивалентной формой удобной для получения различных оценок. В частности, будет использовано неравенство вида

$$\frac{1}{2}|z^* - x|^2 + \alpha_n f(z^*) \leq \frac{1}{2}|z - x|^2 + \alpha_n f(z) - \frac{1}{2}|z - z^*|^2 \quad \forall z \in Z, \quad (69)$$

которому подчинена любая функция вида $\frac{1}{2}|z - x|^2 + \alpha_n f(z)$, где $f(z)$ выпуклая не обязательно дифференцируемая функция определенная на выпуклом множестве Z , где $z \in Z$ и z^* - точка минимума функции $\varphi(z) = \frac{1}{2}|z - x|^2 + \alpha_n f(z)$ на Z для некоторого x [9].

Поскольку целевые функции во втором и четвертом уравнениях процесса (68) имеют структуру функции $\frac{1}{2}|z - x|^2 + \alpha_n f(z)$, то эти уравнения можно переписать в виде

$$\begin{aligned} &|\bar{v}^n - v^n|^2 + 2\alpha(\varphi(\bar{v}^n) + \Phi(v^n, \bar{v}^n) + \langle \bar{p}^n, g(\bar{v}^n) \rangle) \leq \\ &\leq |w - v^n|^2 + 2\alpha(\varphi(w) + \Phi(v^n, w) + \langle \bar{p}^n, g(w) \rangle) - |\bar{v}^n - w|^2, \end{aligned} \quad (70)$$

и

$$\begin{aligned} & |v^{n+1} - w^n|^2 + 2\alpha(\varphi(v^{n+1}) + \Phi(\bar{v}^n, v^{n+1}) + \langle \bar{p}^n, g(w^{n+1}) \rangle) \leq \\ & \leq |w - w^n|^2 + 2\alpha(\varphi(w) + \Phi(\bar{v}^n, w) + \langle \bar{p}^n, g(w) \rangle) - |w - w^{n+1}|^2. \end{aligned} \quad (71)$$

Первое и третье уравнения из (68) в соответствии с определением оператора проектирования запишем в виде

$$\langle \bar{p}^n - p^n - \alpha g(v^n), p - \bar{p}^n \rangle \geq 0 \quad \forall p \geq 0, \quad (72)$$

и

$$\langle p^{n+1} - p^n - \alpha g(\bar{v}^n), p - p^{n+1} \rangle \geq 0 \quad \forall p \geq 0. \quad (73)$$

Будем предполагать так же, что функция $\Phi(v, w)$ удовлетворяет условию типа Липшица в форме

$$|\{\Phi(w + h, v + k) - \Phi(w + h, v)\} - \{\Phi(w, v + k) - \Phi(w, v)\}| \leq |\Phi| |h| |k| \quad (74)$$

для всех $w, w + h \in \Omega, v, v + k \in \Omega$, где $|\Phi|$ - константа. Класс функций, удовлетворяющий этому условию не пуст [1]. Кроме того, предполагается, что векторная функция $g(w)$ удовлетворяет условию Липшица

$$|g(w + h) - g(w)| \leq |g| |h| \quad (75)$$

для всех $w, w + h \in \Omega$, где $|g|$ - константа.

Получим оценку отклонения вектора v^{n+1} и \bar{v}^n друг от друга. С этой целью в неравенстве (70) положим $w = v^{n+1}$, а в неравенстве (71) $w = \bar{v}^n$

$$\begin{aligned} & |\bar{v}^n - v^n|^2 + 2\alpha(\varphi(\bar{v}^n) + \Phi(v^n, \bar{v}^n) + \langle \bar{p}^n, g(\bar{v}^n) \rangle) \leq \\ & \leq |v^{n+1} - v^n|^2 + 2\alpha(\varphi(v^{n+1}) + \Phi(v^n, v^{n+1}) + \langle \bar{p}^n, g(v^{n+1}) \rangle) - |v^{n+1} - \bar{v}^n|^2, \end{aligned}$$

и

$$\begin{aligned} & |v^{n+1} - v^n|^2 + 2\alpha(\varphi(v^{n+1}) + \Phi(\bar{v}^n, v^{n+1}) + \langle \bar{p}^n, g(v^{n+1}) \rangle) \leq \\ & \leq |\bar{v}^n - v^n|^2 + 2\alpha(\varphi(\bar{v}^n) + \Phi(\bar{v}^n, \bar{v}^n) + \langle \bar{p}^n, g(\bar{v}^n) \rangle) - |\bar{v}^n - v^{n+1}|^2. \end{aligned}$$

Затем сложим оба неравенства

$$|\bar{v}^n - v^{n+1}|^2 + \alpha[\Phi(v^n, \bar{v}^n) - \Phi(v^n, v^{n+1}) - \Phi(\bar{v}^n, \bar{v}^n) + \Phi(\bar{v}^n, v^{n+1})] \leq 0$$

Отсюда с учетом (74) имеем

$$|\bar{v}^n - v^{n+1}| \leq \alpha |\Phi| |v^n - \bar{v}^n|. \quad (76)$$

Аналогично получим оценку для векторов \bar{p}^n и p^{n+1} из (68)

$$|\bar{p}^n - p^{n+1}| \leq \alpha |g| |v^n - \bar{v}^n|. \quad (77)$$

Покажем, что процесс (68) сходится монотонно по норме к одному из равновесных решений.

Т е о р е м а 1 Если равновесное решение задачи (58) удовлетворяет условию (64), функции $\varphi(w) + \Phi(v, w)$, $g(w)$ выпуклы по w для любого v , кроме того выполняются условия (74), (75), $\Omega \subseteq R^n$ - выпуклое замкнутое множество, то последовательность v^n , порожденная методом (68) с параметром $0 < \alpha < (\sqrt{2(|\Phi|^2 + |g|^2)})^{-1}$, сходится монотонно по норме к одному из равновесных решений, т.е. $v^n, p^n \rightarrow v^*, p^*$ при $n \rightarrow \infty$.

Д о к а з а т е л ь с т в о. Положим $w = v^*$ в (71), тогда

$$\begin{aligned} & |v^{n+1} - v^n|^2 + 2\alpha(\varphi(v^{n+1}) + \Phi(\bar{v}^n, v^{n+1}) + \langle \bar{p}^n, g(v^{n+1}) \rangle) \leq \\ & \leq |v^* - v^n|^2 + 2\alpha(\varphi(v^*) + \Phi(\bar{v}^n, v^*) + \langle \bar{p}^n, g(v^*) \rangle) - |v^{n+1} - v^*|^2. \end{aligned}$$

Положим $w = v^{n+1}$ в (70)

$$\begin{aligned} & |\bar{v}^n - v^n|^2 + 2\alpha(\varphi(\bar{v}^n) + \Phi(v^n, \bar{v}^n) + \langle \bar{p}^n, g(\bar{v}^n) \rangle) \leq \\ & \leq |v^{n+1} - v^n|^2 + 2\alpha(\varphi(v^{n+1}) + \Phi(v^n, v^{n+1}) + \langle \bar{p}^n, g(v^{n+1}) \rangle) - |v^{n+1} - \bar{v}^n|^2. \end{aligned}$$

Сложим полученные неравенства

$$\begin{aligned} & |v^{n+1} - v^*|^2 + |v^{n+1} - \bar{v}^n|^2 + |\bar{v}^n - v^n|^2 + 2\alpha\{\Phi(\bar{v}^n, \bar{v}^n) + \varphi(\bar{v}^n) - \Phi(\bar{v}^n, v^*) - \varphi(v^*)\} - \\ & - 2\alpha\{\Phi(\bar{v}^n, \bar{v}^n) - \Phi(\bar{v}^n, v^{n+1}) - \Phi(v^n, \bar{v}^n) + \Phi(v^n, v^{n+1})\} + \\ & + 2\alpha\langle \bar{p}^n, g(\bar{v}^n) - g(v^*) \rangle \leq |v^n - v^*|^2. \end{aligned}$$

Отсюда, с учетом (74) имеем

$$\begin{aligned} & |v^{n+1} - v^*|^2 + |v^{n+1} - \bar{v}^n|^2 + (1 - 2\alpha^2|\Phi|^2)|\bar{v}^n - v^n|^2 + \\ & + 2\alpha\{\Phi(\bar{v}^n, \bar{v}^n) + \varphi(\bar{v}^n) - \Phi(\bar{v}^n, v^*) - \varphi(v^*)\} + 2\alpha\langle \bar{p}^n, g(\bar{v}^n) - g(v^*) \rangle \leq |v^n - v^*|^2. \end{aligned} \quad (78)$$

Положим $w = \bar{v}^n$ в (65) и выпишем его в виде

$$\varphi(v^*) + \Phi(\bar{v}^n, v^*) - \varphi(\bar{v}^n) - \Phi(\bar{v}^n, \bar{v}^n) + \langle p^*, g(v^*) - g(\bar{v}^n) \rangle \leq 0,$$

или

$$\varphi(\bar{v}^n) + \Phi(\bar{v}^n, \bar{v}^n) - \varphi(v^*) - \Phi(\bar{v}^n, v^*) \geq \langle p^*, g(v^*) - g(\bar{v}^n) \rangle.$$

Учитывая полученную оценку в (78), имеем

$$\begin{aligned} & |v^{n+1} - v^*|^2 + |v^{n+1} - \bar{v}^n|^2 + (1 - 2\alpha^2|\Phi|^2)|\bar{v}^n - v^n|^2 + \\ & + 2\alpha\langle \bar{p}^n - p^*, g(\bar{v}^n) - g(v^*) \rangle \leq |v^n - v^*|^2. \end{aligned} \quad (79)$$

Рассмотрим неравенства (72) и (73). Положим $p = p^*$ в (73)

$$\langle p^{n+1} - p^n, p^* - p^{n+1} \rangle - \alpha\langle g(\bar{v}^n), p^* - p^{n+1} \rangle \geq 0 \quad (80)$$

и $p = p^{n+1}$ в (72)

$$\langle \bar{p}^n - p^n, p^{n+1} - \bar{p}^n \rangle + \alpha\langle g(\bar{v}^n) - g(v^n), p^{n+1} - \bar{p}^n \rangle - \alpha\langle g(\bar{v}^n), p^{n+1} - \bar{p}^n \rangle \geq 0, \quad (81)$$

Второе слагаемое в этом неравенстве оценим с помощью (75),(77), затем неравенства (80) и (81) сложим

$$\begin{aligned} & \langle p^{n+1} - p^n, p^* - p^{n+1} \rangle + \langle \bar{p}^n - p^n, p^{n+1} - \bar{p}^n \rangle + \\ & + \alpha^2 |g|^2 |\bar{v}^n - v^n|^2 - \alpha \langle g(\bar{v}^n), p^* - \bar{p}^n \rangle \geq 0, \end{aligned}$$

С помощью тождества

$$|x_1 - x_3|^2 = |x_1 - x_2|^2 + 2\langle x_1 - x_2, x_2 - x_3 \rangle + |x_2 - x_3|^2$$

разложим два первых скалярных произведения в сумму квадратов

$$\begin{aligned} & |p^{n+1} - p^*|^2 + |p^{n+1} - \bar{p}^n|^2 + |\bar{p}^n - p^n|^2 - \\ & - 2\alpha^2 |g|^2 |\bar{v}^n - v^n|^2 + 2\alpha \langle g(\bar{v}^n), p^* - \bar{p}^n \rangle \leq |p^n - p^*|^2. \end{aligned} \quad (82)$$

Далее сложим (79) и (82). Учитывая, что $\langle p^{n+1}, g(v^*) \rangle \leq 0$, $\langle p^*, g(v^*) \rangle = 0$, а также оценку (61) при $v = \bar{v}^n$ получим

$$\begin{aligned} & |v^{n+1} - v^*|^2 + |p^{n+1} - p^*|^2 + |v^{n+1} - \bar{v}^n|^2 + |p^{n+1} - \bar{p}^n|^2 + \\ & + ((1 - 2\alpha^2(|\Phi|^2 + |g|^2))|\bar{v}^n - v^n|^2 + |\bar{p}^n - p^n|^2 \leq |v^n - v^*|^2 + |p^n - p^*|^2. \end{aligned} \quad (83)$$

Просуммируем неравенство (83) от $n = 0$ до $n = N$

$$\begin{aligned} & |v^{N+1} - v^*|^2 + |p^{N+1} - p^*|^2 + 2d \sum_{k=0}^{k=N} |\bar{u}^k - v^k|^2 + \sum_{k=0}^{k=N} |v^{k+1} - \bar{u}^k|^2 + \sum_{k=0}^{k=N} |p^{k+1} - \bar{p}^k|^2 + \\ & + \sum_{k=0}^{k=N} |\bar{p}^k - p^k|^2 \leq |v^0 - v^*|^2 + |p^0 - p^*|^2, \end{aligned}$$

где $d = 1 - 2\alpha^2(|\Phi|^2 + |g|^2) > 0$. Из полученного неравенства следует ограниченность траектории:

$$|v^{N+1} - v^*|^2 + |p^{N+1} - p^*|^2 \leq |v^0 - v^*|^2 + |p^0 - p^*|^2,$$

а также, сходимость рядов:

$$\sum_{k=0}^{\infty} |\bar{u}^k - v^k|^2 < \infty, \quad \sum_{k=0}^{\infty} |v^{k+1} - \bar{u}^k|^2 < \infty, \quad \sum_{k=0}^{\infty} |p^{k+1} - \bar{p}^k|^2 < \infty, \quad \sum_{k=0}^{\infty} |\bar{p}^k - p^k|^2 < \infty,$$

и, следовательно, стремление к нулю величин

$$|\bar{u}^n - v^n|^2 \rightarrow 0, \quad |v^{n+1} - \bar{u}^n|^2 \rightarrow 0, \quad |p^{n+1} - \bar{p}^n|^2 \rightarrow 0, \quad |\bar{p}^n - p^n|^2 \rightarrow 0, \quad n \rightarrow \infty.$$

Так как последовательность v^n, p^n ограничена, то существует элемент v', p' , такой, что $v^{n_i} \rightarrow v', p^{n_i} \rightarrow p'$ при $n_i \rightarrow \infty$, и при этом

$$|v^{n_i+1} - \bar{u}^{n_i}|^2 \rightarrow 0, \quad |\bar{u}^{n_i} - v^{n_i}|^2 \rightarrow 0, \quad |p^{n_i+1} - \bar{p}^{n_i}|^2 \rightarrow 0, \quad |\bar{p}^{n_i} - p^{n_i}|^2 \rightarrow 0.$$

Рассмотрим неравенства (68) для всех $n_i \rightarrow \infty$ и, перейдя к пределу, получим:

$$p' = \pi_+(p' + \alpha g(v')),$$

$$\varphi(v' + \Phi(v', v')) + \langle p', g(v') \rangle \leq \varphi(w + \Phi(v', w)) + \langle p', g(w) \rangle$$

для всех $w \in \Omega$.

Поскольку эти соотношения эквивалентны (67), то $v' = v^* \in \Omega^*$, $p' = p^* \geq 0$, т.е. любая предельная точка последовательности v^n, p^n является решением задачи. Условие монотонности убывания величины $|v^n - v^*| + |p^n - p^*|$ обеспечивает единственность предельной точки, т.е. сходимость $v^n \rightarrow v^*$, $p^n \rightarrow p^*$ при $n \rightarrow \infty$. Теорема доказана.

Доказанная теорема является одновременно обоснованием доказательства сходимости игрового экстрапроксимального метода для решения игры двух лиц с функциональными ограничениями (5). Метод этой игры имеет вид:

первый полушаг (прогнозный)

$$\bar{p}_1^n = \pi_+(p_1^n + \alpha g_1(x_1^n)),$$

$$\bar{x}_1^n \in \text{Argmin}\{1/2|z_1 - x_1^n|^2 + \alpha(\varphi_1(z_1) + f_1(z_1, x_2^n) + \langle \bar{p}_1^n, g_1(z_1) \rangle) \mid z_1 \in X_1\},$$

$$\bar{p}_2^n = \pi_+(p_2^n + \alpha g_2(x_2^n)),$$

$$\bar{x}_2^n \in \text{Argmin}\{1/2|z_2 - x_2^n|^2 + \alpha(\varphi_2(z_2) + f_2(x_1^n, z_2) + \langle \bar{p}_2^n, g_2(z_2) \rangle) \mid z_2 \in X_2\},$$

второй полушаг (основной)

$$p_1^{n+1} = \pi_+(p_1^n + \alpha g_1(\bar{x}_1^n)),$$

$$x_1^{n+1} \in \text{Argmin}\{1/2|z_1 - x_1^n|^2 + \alpha(\varphi_1(z_1) + f_1(z_1, \bar{x}_2^n) + \langle \bar{p}_1^n, g_1(z_1) \rangle) \mid z_1 \in X_1\},$$

$$p_2^{n+1} = \pi_+(p_2^n + \alpha g_2(\bar{x}_2^n)),$$

$$x_2^{n+1} \in \text{Argmin}\{1/2|z_2 - x_2^n|^2 + \alpha(\varphi_2(z_2) + f_2(\bar{x}_1^n, z_2) + \langle \bar{p}_2^n, g_2(z_2) \rangle) \mid z_2 \in X_2\}.$$

Длина шага α в этом процессе определяется из условия $0 < \alpha < \alpha_0$ где α_0 - константа. Экстраградиентный подход для решения игр двух лиц исследован в [15].

5.2 Экстрапроксимальный метод с перекосом для решения равновесной задачи дефицита ресурсов. В задаче требуется вычислить вектор $y \geq 0$, который является решением системы уравнений

$$w(y), p(y) \in \text{Argsdl}\{\varphi(w) + \langle p, g(w) - s(y) \rangle \mid w \in \Omega, p \in R_+^n\}.$$

$$-p(y) = s(y) \tag{84}$$

где $s(y)$ -априори заданное отображение, а $w(y), p(y)$ - седловая точка функции

$$\mathcal{L}(w, p, s(y)) = \varphi(w) + \langle p, g(w) - s(y) \rangle \quad \forall w \in \Omega, \quad p \in R_+^m$$

при фиксированном значении параметра $y \in R_+^m$.

Пусть $y = y^*$ решение (84), тогда эту задачу можно переформулировать в следующем виде: требуется вычислить вектор $y^* \geq 0$, который является решением седлового включения

$$w(y^*), s(y^*) \in \text{Argsdl}\{\varphi(w) + \langle p, g(w) - s(y^*) \rangle \mid w \in \Omega, p \in R_+^m\}. \tag{85}$$

где $w(y^*), p(y^*)$ -седловая точка функции

$$\mathcal{L}(w, p, s) = \varphi(w) + \langle p, g(w) - s(y^*) \rangle \quad \forall w \in \Omega, \quad p \in R_+^m$$

при этом $p^* = s^*$ при фиксированном значении $y = y^*$.

Наряду с функцией $\mathcal{L}(w, p, s)$ рассмотрим ее сужение $\mathcal{L}(w, p, p)$ на "плоскость" $s = (1/2)p$

$$\mathcal{L}(w, p, p) = \varphi(w) + \langle p, g(w) - (1/2)p \rangle \quad \forall w \in \Omega, \quad p \in R_+^m.$$

Введенная функция является выпуклой по $w \in \Omega$ и вогнутой (точнее сильно вогнутой) по переменной $p \in R_+^m$. Такая функция, как правило, всегда имеет седловую точку. Обозначим ее как $w^* \in \Omega$ и $p^* \in R_+^m$. По определению эта точка удовлетворяет системе неравенств

$$\begin{aligned} & \varphi(w^*) + \langle p, g(w^*) - (1/2)p \rangle \leq \\ & \leq \varphi(w^*) + \langle p^*, g(w^*) - (1/2)p^* \rangle \leq \\ & \leq \varphi(w) + \langle p^*, g(w) - (1/2)p^* \rangle \quad \forall w \in \Omega, p \in R_+^m. \end{aligned} \quad (86)$$

Левое неравенство системы представляет собой задачу оптимизации

$$p^* \in \text{Argmax}\{\langle p, g(w^*) - (1/2)p \rangle \mid p \geq 0\}.$$

Необходимое условие максимума этой задачи выпишем в виде вариационного неравенства

$$\langle g(w^*) - p^*, p - p^* \rangle \leq 0, \quad p \geq 0. \quad (87)$$

Положим в этом неравенстве сначала $p = 0$, а затем $p = 2p^*$, тогда получим

$$\langle g(w^*) - p^*, p^* \rangle = 0. \quad (88)$$

Из этого неравенства также следует, что

$$g(w^*) - p^* \leq 0 \quad (89)$$

(если предположить, что это не так, то приходим к противоречию с (87)).

Правое неравенство системы (86) с учетом (88) представим в следующем виде

$$\varphi(w^*) \leq \varphi(w) + \langle p^*, g(w) - p^* \rangle, \quad w \in \Omega.$$

Дополнительно потребуем что переменная w подчинена условию $\langle p^*, g(w) - g(w^*) \rangle \leq 0$. Тогда выше введенное неравенство приводится к задаче оптимизации с одним скалярным ограничением

$$\varphi(w^*) \leq \varphi(w), \quad \langle p^*, g(w) - p^* \rangle \leq 0, \quad w \in \Omega.$$

Поскольку точка минимума этой задачи w^* согласно (89) принадлежит множеству $\{w \mid g(w) \leq p^*, w \in \Omega\}$, которое вложено в допустимое множество выше приведенной задачи, то точка w^* будет являться решением задачи оптимизации

$$w \in \text{Argmin}\{\varphi(w) \mid g(w) \leq p^*, w \in \Omega\}. \quad (90)$$

Из (87) и (88) следует, что пара w^*, p^* есть седловая точка функции Лагранжа задачи (90), т.е. p^* - ее двойственное решение. Другими словами, двойственное решение этой задачи совпадает с вектором ресурсов правой части ограничений. Чтобы получить окончательное решение задачи (85), необходимо решить уравнение $p^* = s(y)$ относительно переменной $y \geq 0$.

Таким образом, доказано следующее

С в о й с т в о 23 . Решение равновесной задачи (85) сводится к поиску седловой точки w^*, p^* функции

$$\mathcal{L}(w, p) = \varphi(w) + \langle p, g(w) - (1/2)p \rangle \quad \forall w \in \Omega, \quad p \in R_+^m \quad (91)$$

и вычислению решения (существование которого предполагается) уравнения

$$p^* = s(y).$$

В частности, векторная функция $s(y) = s_2(y_2)$ может быть функцией множителей Лагранжа другой задачи выпуклого программирования, например

$$w_2^* \in \text{Argmax}\{\varphi_2(w_2) \mid g_2(w_2) \geq y_2, w_2 \in \Omega_2\}. \quad (92)$$

Последнее означает, что вычислив седловую точку функции Лагранжа (91), используем ее вектор множителей Лагранжа $p^* = s_2$, чтобы решить задачу оптимизации

$$\mathcal{L}(w_2, p_2) = \varphi_2(w_2) + \langle s_2, g_2(w_2) - (1/2)p_2 \rangle \quad \forall w_2 \in \Omega_2.$$

Вектор y_2 может быть вычислен по формуле $y_2 = g(w^*)$. Если обе задачи (90) и (92) моделируют производственную деятельность предприятий, то равновесная задача дефицита ресурсов согласовывает объемы поставок ресурсов одного предприятия другому в терминах внутренних маргинальных цен (расценок).

Перейдем к обсуждению вопросов вычисления седловой точки функции (91). Седловая точка функции $\mathcal{L}(w, p)$ удовлетворяет системе неравенств (86). Необходимые условия минимума и максимума этой функции по своим переменным выпишем в форме проксимальных уравнений

$$w^* = \text{Argmin}\{(1/2)|w - w^*|^2 + \alpha(\varphi(w) + \langle p^*, g(w) - (1/2)p^* \rangle) \mid w \in \Omega\}.$$

$$p^* = \text{Argmax}\{-(1/2)|p - p^*|^2 + \alpha(\langle p, g(w^*) - (1/2)p \rangle) \mid p \in R_+^m\}. \quad (93)$$

Чтобы решить систему проксимальных уравнений используем экстрапроксимальный итеративный процесс [5],[14]

$$\begin{aligned} \bar{w}^n &\in \text{argmin}\{(1/2)|w - w^n|^2 + \alpha(\varphi(w) + \langle p^n, g(w) \rangle) \mid w \in \Omega\}, \\ p^{n+1} &\in \text{argmin}\{(1/2)|p - p^n|^2 - \alpha(\langle p, g(\bar{w}^n) - (1/2)p \rangle) \mid p \geq 0\}, \\ w^{n+1} &\in \text{argmin}\{(1/2)|w - w^n|^2 + \alpha(\varphi(w) + \langle p^{n+1}, g(w) \rangle) \mid w \in \Omega\}, \end{aligned} \quad (94)$$

Поскольку целевые функции процесса (94) имеют структуру функции $\frac{1}{2}|z - x|^2 + \alpha_n f(z)$, все уравнения из (94) перепишем в эквивалентной форме (69)

$$\begin{aligned} &|\bar{w}^n - w^n|^2 + 2\alpha(\varphi(\bar{w}^n) + \langle p^n, g(\bar{w}^n) \rangle) \leq \\ &\leq |w - w^n|^2 + 2\alpha(\varphi(w) + \langle p^n, g(w) \rangle) - |\bar{w}^n - w|^2, \end{aligned} \quad (95)$$

$$\begin{aligned} &|p^{n+1} - p^n|^2 - 2\alpha\langle p^{n+1}, g(\bar{w}^n) - (1/2)p^{n+1} \rangle \leq \\ &\leq |p - p^n|^2 - 2\alpha\langle p, g(\bar{w}^n) - (1/2)p \rangle - |p - p^{n+1}|^2, \end{aligned} \quad (96)$$

$$|w^{n+1} - w^n|^2 + 2\alpha(\varphi(w^{n+1}) + \langle p^{n+1}, g(w^{n+1}) \rangle) \leq$$

$$\leq |w - w^n|^2 + 2\alpha(\varphi(w) + \langle p^{n+1}, g(w) \rangle) - |w - w^{n+1}|^2. \quad (97)$$

Оценим величину $|w^{n+1} - \bar{w}^n|$ отклонения двух векторов из (94) при этом используем условие Липшица

$$|g(w_1) - g(w_2)| \leq |g||w_1 - w_2| \quad \forall w_1, w_2 \in \Omega, \quad (98)$$

где $|g|$ -константа.

Положим $w = v^{n+1}$ в первом неравенстве этой системы и $w = \bar{v}^n$ в третьем неравенстве, затем сложим оба неравенства

$$\begin{aligned} & |\bar{w}^n - w^n|^2 + 2\alpha(\varphi(\bar{w}^n) + \langle p^n, g(\bar{w}^n) \rangle) \leq \\ & \leq |w^{n+1} - w^n|^2 + 2\alpha(\varphi(w^{n+1}) + \langle p^n, g(w^{n+1}) \rangle) - |\bar{w}^n - w^{n+1}|^2, \\ & |w^{n+1} - w^n|^2 + 2\alpha(\varphi(w^{n+1}) + \langle p^{n+1}, g(w^{n+1}) \rangle) \leq \\ & \leq |\bar{w}^n - w^n|^2 + 2\alpha(\varphi(\bar{w}^n) + \langle p^{n+1}, g(\bar{w}^n) \rangle) - |\bar{w}^n - w^{n+1}|^2. \end{aligned}$$

Сложим полученные неравенства, тогда с учетом (98) имеем

$$2|\bar{w}^n - w^{n+1}|^2 \leq 2\alpha\langle p^n - p^{n+1}, g(\bar{w}^n) - g(w^{n+1}) \rangle \leq 2\alpha|p^n - p^{n+1}||g||\bar{w}^n - w^{n+1}|.$$

Отсюда

$$|\bar{w}^n - w^{n+1}| \leq \alpha|g||p^n - p^{n+1}|. \quad (99)$$

Докажем теорему

Т е о р е м а 2 Если седловая точка задачи (85) существует, функции $\varphi(w)$ и $g(w)$ выпуклы, $g(w)$ удовлетворяет условию Липшица (98) с константой $|g|$, Ω -выпуклое замкнутое множество, то последовательность w^n, p^n экстрапроксимального метода (94) с выбором длины шага из условия $\alpha < 1/(\sqrt{2}|g|)$, сходится монотонно по норме к одному из равновесных решений, т.е. $w^n, p^n \rightarrow w^*, p^*$ при $n \rightarrow \infty$ для всех w^0, p^0 .

Доказательство. Положим $w = w^*$ в (97) и $w = w^{n+1}$ в (95), тогда

$$\begin{aligned} & |w^{n+1} - w^n|^2 + 2\alpha(\varphi(w^{n+1}) + \langle p^{n+1}, g(w^{n+1}) \rangle) \leq \\ & \leq |w^* - w^n|^2 + 2\alpha(\varphi(w^*) + \langle p^{n+1}, g(w^*) \rangle) - |w^{n+1} - w^*|^2 \end{aligned}$$

и

$$\begin{aligned} & |\bar{w}^n - w^n|^2 + 2\alpha(\varphi(\bar{w}^n) + \langle p^n, g(\bar{w}^n) \rangle) \leq \\ & \leq |w^{n+1} - w^n|^2 + 2\alpha(\varphi(w^{n+1}) + \langle p^n, g(w^{n+1}) \rangle) - |\bar{w}^n - w^{n+1}|^2. \end{aligned}$$

Сложим полученные неравенства, предварительно прибавив нуль в виде $\langle p^{n+1}, g(\bar{w}^n) - \langle p^{n+1}, g(\bar{w}^n) \rangle = 0$, тогда

$$\begin{aligned} & |w^{n+1} - w^*|^2 + |w^{n+1} - \bar{w}^n|^2 + |\bar{w}^n - w^n|^2 + 2\alpha(\varphi(\bar{w}^n) - \varphi(w^*)) + \\ & + 2\alpha(\langle p^n, g(\bar{w}^n) \rangle - \langle p^{n+1}, g(\bar{w}^n) \rangle) - \langle p^n, g(w^{n+1}) \rangle + \langle p^{n+1}, g(w^{n+1}) \rangle + \\ & + \langle p^{n+1}, g(\bar{w}^n) \rangle - \langle p^{n+1}, g(w^*) \rangle \leq |w^n - w^*|^2. \end{aligned}$$

Отсюда с учетом (99)

$$|w^{n+1} - w^*|^2 + |w^{n+1} - \bar{w}^n|^2 + |\bar{w}^n - w^n|^2 + 2\alpha(\varphi(\bar{w}^n) - \varphi(w^*)) +$$

$$2\alpha(\langle p^n - p^{n+1} \rangle + 2\alpha(\langle p^{n+1}, g(\bar{w}^n) \rangle - \langle p^{n+1}, g(w^*) \rangle)) \leq |w^n - w^*|^2.$$

Положим $w = \bar{w}^n$ в (86), тогда

$$\varphi(w^*) + \langle p^*, g(w^*) \rangle \leq \varphi(\bar{w}^n) + \langle p^*, g(\bar{w}^n) \rangle$$

и сложим два последних неравенства

$$\begin{aligned} & |w^{n+1} - w^*|^2 + |w^{n+1} - \bar{w}^n|^2 + |\bar{w}^n - w^n|^2 + \\ & + 2\alpha^2|g|^2|p^{n+1} - p^n|^2 + \langle p^{n+1} - p^*, g(\bar{w}^n) - g(w^*) \rangle \leq |w^n - w^*|^2. \end{aligned} \quad (100)$$

К полученной оценке вернемся позже, а сейчас рассмотрим неравенство (96), где положим $p = p^*$, тогда

$$\begin{aligned} & |p^{n+1} - p^n|^2 - 2\alpha\langle p^{n+1}, g(\bar{w}^n) - (1/2)p^{n+1} \rangle \leq \\ & \leq |p^* - p^n|^2 - 2\alpha\langle p^*, g(\bar{w}^n) - (1/2)p^* \rangle - |p^* - p^{n+1}|^2. \end{aligned}$$

С другой стороны в (86) положим $p = p^{n+1}$

$$\langle p^{n+1}, g(w^*) - (1/2)p^{n+1} \rangle \leq \langle p^*, g(w^*) - (1/2)p^* \rangle.$$

Сложив оба неравенства, получим

$$|p^{n+1} - p^*|^2 + |p^{n+1} - p^n|^2 + 2\alpha\langle p^{n+1} - p^*, g(w^*) - g(\bar{w}^n) \rangle \leq |p^n - p^*|^2. \quad (101)$$

Окончательно просуммируем оценки (100) и (101)

$$\begin{aligned} & |w^{n+1} - w^*|^2 + |w^{n+1} - \bar{w}^n|^2 + |\bar{w}^n - w^n|^2 + |p^{n+1} - p^*|^2 + \\ & + (1 - 2\alpha^2|g|^2)|p^{n+1} - p^n|^2 \leq |w^n - w^*|^2 + |p^n - p^*|^2. \end{aligned}$$

Из этого неравенства, в частности, следует монотонность убывания величины $|w^n - w^*|^2 + |p^n - p^*|^2$ с ростом n . Просуммируем полученное неравенство неравенство от $n = 0$ до $n = N$:

$$\begin{aligned} & |w^{N+1} - w^*|^2 + |p^{N+1} - p^*|^2 + \sum_{k=0}^{k=N} (|w^{k+1} - \bar{w}^k|^2 + |\bar{w}^k - w^k|^2) + \\ & + \varepsilon \sum_{k=0}^{k=N} |p^{k+1} - p^k|^2 \leq |w^n - w^*|^2 + |p^n - p^*|^2, \end{aligned}$$

где $\varepsilon = 1 - 2\alpha^2|g|^2$. Из полученного неравенства следует ограниченность последовательности

$$|w^{N+1} - w^*|^2 + |p^{N+1} - p^*|^2 \leq |w^0 - w^*|^2 + |p^0 - p^*|^2,$$

сходимость рядов $\sum_{k=0}^{\infty} (|w^{k+1} - \bar{w}^k|^2 + |\bar{w}^k - w^k|^2) < \infty$, $\sum_{k=0}^{\infty} |p^{k+1} - p^k|^2 < \infty$, и, следовательно, стремление к нулю величин $|w^{n+1} - \bar{w}^n|^2 \rightarrow 0$, $|\bar{w}^n - w^n|^2 \rightarrow 0$, $|p^{n+1} - p^n|^2 \rightarrow 0$ при $n \rightarrow \infty$.

Так как последовательность w^n, p^n ограничена, то существует такой элемент w', p' что $w^{n_i}, p^{n_i} \rightarrow w', p'$, при $n_i \rightarrow \infty$ и при этом $|w^{n_i+1} - \bar{w}^{n_i}|^2 \rightarrow 0$, $|w^{n_i+1} - w^{n_i}|^2 \rightarrow 0$, $|\bar{w}^{n_i} - p^{n_i}|^2 \rightarrow 0$. Рассмотрим неравенства (95) и (96) при всех $n_i \rightarrow \infty$ и, переходя к пределу, получим предельные неравенства

$$\varphi(w') + \Phi(\bar{v}', w') \leq \varphi(w) + \Phi(\bar{v}', w) \quad \forall w \in \Omega.$$

Очевидно, что это неравенство совпадает с (84) и, следовательно, $w', p' = w^*, p^* \in \Omega \times \Omega$. Таким образом, любая предельная точка последовательности w^n, p^n является решением равновесной задачи. В силу монотонности убывания величины $|w^n - w^*|^2 + |p^n - p^*|^2$ эта последовательность может иметь только одну предельную точку, т.е. w^n, p^n монотонно по норме сходится к одному из равновесных решений задачи: $w^n, p^n \rightarrow w^*, p^*$ при $n \rightarrow \infty$. Теорема доказана.

Список литературы

- [1] Антипин А.С. *Градиентный и экстраградиентный подходы в билинейном равновесном программировании*. Москва ВЦ РАН, -2002.-132.
- [2] Aubin J.-P., Frankowska H. *Set Valued Analysis*. Boston etc.: Birkhauser, 1990.
- [3] Fan Ky. *A minimax Inequality and Applications. Inequalities III* New York. Acad. Press, 1972. 103-113.
- [4] Nash J.F. Jr. *Equilibrium points in n-person games*. Proc. Nat.Acad.Sci. USA, 1950, 36, 48-49.
- [5] Antipin A. S. *Equilibrium programming problem: prox-regularization and prox-methods*. In the book "Recent Advances in Optimization". Lecture Notes in Economics and Mathematical Systems. Springer. 1997
- [6] Zlobec S. *Stable Parametric Programming*. Kluwer Academic Publishers. Dordrecht etc. 2001.
- [7] Гольштейн Е.Г., Третьяков Н.В. *Модифицированные функции Лагранжа*. Наука. Москва. 1989.
- [8] Васильев Ф.П., Иваницкий А.Ю. *Линейное программирование*. Факториал. Москва. 2003.
- [9] Антипин А.С. *О сходимости и оценках скорости сходимости проксимальных методов к неподвижным точкам экстремальных отображений*. Журнал вычисл. мат. и мат. физ. 1995. Т. 35. No.5, 688-704.
- [10] Антипин А.С. *Метод внутренней линеаризации для задач равновесного программирования*. Журнал вычисл. мат.и мат. физ. 2000. Т.40. No.8, 1142-1162.
- [11] Антипин А.С. *Расщепление градиентного подхода для решения экстремальных включений*. Журнал вычисл. мат.и мат. физ. 1998. Т.38. No.7, 1118-1132.
- [12] Konnov I. *Combind Relaxation Methods for Variational Inequalities*. Springer-Verlag. Berlin etc. 2001.
- [13] Hadjisavvas N., Schaible S. *On strong pseudomonotonicity and (semi) strict quasimonotonicity*. Journal of Optimization Theory and Applications. 1993. V. 79, 139-155.
- [14] Антипин А.С. *Равновесное программирование: проксимальные методы*. Журнал вычисл. мат.и мат. физ. 1997. Т.37. No.11, 1327-1339.
- [15] Antipin A. S. *Extragradient approach to the solution of two person non-zero sum games*. In the book "Optimization and Optimal Control". World Scienttific. New Jersey etc. 2003. 1-28.

TO GLOBAL THEORY FOR EQUILIBRIUM AND GAME PROBLEMS. 1.

A.S. Antipin

Computing Center of Russian Academy of Sciences, Moscow
e-mail: antipin@ccas.ru

Abstract. The equilibrium programming problem and equilibrium model of deficit resources are considered in the paper. The general theory for equilibrium problems is developed. The convergence for extraproximal methods for solving equilibrium and saddle point problems are justified.

Key words: equilibrium programming problem, two-person game, equilibrium model of deficit resources, Nash equilibrium, extraproximal method, convergence.

METHODS OF THE CHEBYSHEV POINTS FOR SOME PROBLEMS OF OPERATIONS RESEARCH¹

V. Bulatov, T. Belykh, A. Burdukovskaja

Institute of Energy systems, Irkutsk
e-mail: bulatov@sei.isem.irk.ru

Abstract. We consider different kinds of the Chebyshev points method in linear and convex programming. Some applications of these methods in applied energy problems and more general problem of system viability are presented. Relation of the Chebyshev points method to global and integer programming problems is given.

Key words: method of centers, convex programming, viability, global optimization.

Introduction

In the last few decades many mathematicians dealing with convex programming problems focus their attention on methods characterized by co-called polynomial complexity. Such methods converge in some sense to the optimal solution with the degree of geometrical progression, and the rate of convergence depends only on the dimension of the considered problem.

In particular, central cutting plane methods falls into this class. The idea of such methods consists in successive localization of a solution by a family of simple structured sets with a measure tending to zero. These methods are the well known ellipsoid method, methods of mass center of simplices (the simplex imbedding methods) [1,4], ellipsoid inscribing methods and others.

In the paper fast modifications and novel versions of the basic central cutting methods are presented.

In the first chapter we study an algorithm for solving convex programming problems based on successive localization of the optimal solution by simplices with volume tending to zero. The mass center of a simplex is used as the simplex center. If a mass center does not belong to the feasible set then k cutting planes are introduced. The planes are associated with discrepancies of the constraints, which form the feasible set.

In a number of important cases it is possible to embed a simplex truncated by k planes into the simplex of the minimal possible volume. For these cases an estimate of convergence rate (of volume ratio of two successive simplices) was obtained.

The second chapter is devoted to other versions of the central cutting plane methods. As in the first chapter, presented algorithms use a sequence of simplices containing the solution of the considered problem. However the Chebyshev points of simplices are taken as their centers here. In this case volumes of simplices do not need to decrease whereas their inscribed balls, cubes, or parallelepipeds reduce in volume.

In the second chapter other modifications of the Chebushev point method are presented as well. As distinct from the methods considered in the first chapter convergence degree of the Chebyshev point methods depends on structure of a problem. Methods of Chebyshev centers or mass centers of convex sets find numerous applications. For example, the third chapter

¹Supported by RFBR grant N 03-01-00518

contains some new versions of branch-and-bound methods for global optimizations of additive and concave functions.

The same technique is useful in creation of new schemes of branch-and-bound methods for integer programming problems. In the third chapter we also consider the simplest inner and outer approximations of the accessibility sets for linear systems of difference equations. Here we again use maximal volume cubes, balls and parallelepipeds inscribed in implicitly determined sets.

In the third chapter we also introduce the concept of dynamic system viability originally considered by admiral S.O. Makarov in the 1870s and comprehensively studied in [3].

In the last chapter it is also shown that the Chebyshev point methods are useful for finding equilibrium points in n-person games and for solving some stochastic programming problems with probability constraints, as applied to optimization of agricultural productivity in the conditions of irrigated farming. The last paragraph of the report is devoted to extension of the Chebyshev point methods to the problems of linear system optimal control with convex constraints on state variables along a whole trajectory.

1. THE CHEBYSHEV POINTS METHODS IN CONVEX PROGRAMMING

1.1. The Chebyshev Points of Convex Sets in E^n

Let the system of linear inequalities

$$\begin{cases} h_1(x) = a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n - b_1 \leq 0, \\ \dots\dots\dots \\ h_m(x) = a_{m1}x_1 + a_{m2}x_2 + \dots + a_{mn}x_n - b_m \leq 0, \end{cases} \quad (1.1)$$

determines a bounded set in E^n and

$$L = \min_x \max_{1 \leq i \leq m} h_i(x) = \max_{1 \leq i \leq m} h_i(x^*). \quad (1.2)$$

System (1.1) is obviously solvable if $L \leq 0$. In this case $|L|$ is a stability measure of solution x^* of system (1.1) in the sense of (1.2). If $L > 0$, then system (1.1) does not have solutions and $|L|$ is its inconsistency measure. A point $x^* = (x_1^*, \dots, x_n^*)$ is called a Chebyshev point of system (1.1) if $\min_x \max_{1 \leq i \leq m} h_i(x) = \max_{1 \leq i \leq m} h_i(x^*) = L \leq 0$. If $L > 0$, then x^* is called a Chebyshev approximation of system (1.1). Let us introduce a new variable x_{n+1} and move the extended space E^{n+1} . Relate to problem (1.2) the following linear programming problem

$$\min x_{n+1} \quad (1.3)$$

subject to

$$\sum_j a_{ij}x_j - b_i \leq x_{n+1}, \quad \forall i = \overline{1, m}. \quad (1.4)$$

It is clear that problem (1.3), (1.4) is equivalent to problem (1.1), (1.2). In the case when (1.4) satisfies the Haar condition the solution of the linear programming problem is unique and determines a unique Chebyshev point corresponding to (1.2). Let us give a geometrical interpretation of the Chebyshev points of a system of linear inequalities $Ax \leq b$, where A is an $(m \times n)$ matrix $m > n$. Assume that rows a^i of the matrix A in (1.1) are normalized and pair $\{x^*, x_{n+1}^*\}$ solves problem (1.3), (1.4). Then x^* is the center of the ball of the

maximum radius $r^* = |x_{n+1}^*|$, inscribed into the polyhedron $R = \{x : Ax \leq b\}$. Moreover, if $\|a^i\| = \sum_{j=1}^n |a_{ij}|$ then r^* is the radius of the maximum volume cube, which belongs to R .

If $\|a^i\| = \sqrt{\sum_j a_{ij}^2}$ then r^* is the radius of the maximum volume ball, which belongs to R . If $\|a^i\| = \max_{1 \leq j \leq n} |a_{ij}|$ then r^* is the radius of the maximum volume rhombus, which belongs to R .

Let $\|a^i\| \neq 1$ and the Chebyshev point x^* should belong to a convex compact set D , $x^* \in D$. Then the solution $\{x^*, x_{n+1}^*\}$ of the following linear programming problem

$$\min\{x_{n+1} : a^i x - b_i \leq \|a^i\| x_{n+1} : x \in D, i = \overline{1, m}\} \quad (1.5)$$

is called a Chebyshev point of system (1.1) over set D .

Similarly to (1.1), (1.2) one could introduce the concept of the Chebyshev point for a system of inequalities defined by convex or quasiconvex functions. Let the system of inequalities

$$\begin{aligned} g_1(x_1, \dots, x_n) = g_1(x) &\leq 0, \\ &\dots\dots\dots \\ g_m(x_1, \dots, x_n) = g_m(x) &\leq 0, \end{aligned}$$

determines the bounded set

$$R = \{x : g_j(x) \leq 0, j = \overline{1, m}\}$$

and

$$L = \min_x \max_{1 \leq j \leq m} g_j(x) = \max_{1 \leq j \leq m} g_j(x^*). \quad (1.2')$$

As in the linear case problem (1.2') is related with the following equivalent convex program:

$$\min\{x_{n+1} : g_j(x) \leq x_{n+1}, j = \overline{1, m}\}. \quad (1.3')$$

Problem (1.3') is simpler than the general convex program due to triviality of finding a feasible solution: for arbitrary $x^0 \in E^n$ and

$$x_{n+1}^0 = (\varepsilon + \max_{1 \leq j \leq m} g_j(x^0)), \quad \varepsilon > 0,$$

(x^0, x_{n+1}^0) is a feasible point for (1.3').

Many papers are devoted to methods of finding the Chebyshev points of convex sets in E^n [4,5,7,8,9 and others].

1.2. Chebyshev Point Methods in Convex Programming

As far as the authors know, first constructions connected to applications of the Chebyshev points in iterative processes of solving convex programming problems were suggested in [4].

Also in [4] some proposals on deleting nonactive constraints at each step of the iterative process were suggested and a theorem stating that any additional constraint is active finitely many times was proved.

Consider the problem

$$\min\{\varphi(x) : x \in R\}, \quad (1.6)$$

$$R = \{x : Ax \leq b\}. \quad (1.7)$$

The current step of the iterative process of solving (1.6), (1.7) (see [4]) has the form:

$$\min\{x_{n+1} : \nabla\varphi(x^j)^T(x - x^j) \leq x_{n+1}, x \in R, j = \overline{1, k}\}. \quad (1.8)$$

At each step in (1.8) a linear programming problem is solved, note that the dimension of the auxiliary problem increases since every time a linear constraint is added. This drawback characterizes almost all cutting plane methods. Geometrically, problem (1.8) is equivalent to finding the Chebyshev point x^* of the system

$$\nabla\varphi(x^j)^T(x - x^j) \leq 0, \quad j = \overline{1, k},$$

under the additional constraint $x^* \in R = \{x : Ax \leq b\}$.

In [4] a theorem on convergence of the sequence $\{x_{n+1}^k\}$ to zero and a theorem on existence of a subsequence $\{x^{lj}\} \subset \{x^j\}$ convergent to a solution of problem (1.6), (1.7) are proved.

Later, the Chebyshev point methods were developed by Kiev mathematics S.I. Zukhovitskii, R.A. Polyak, M.E. Primak, and others.

In [5] methods of the type (1.8) are used for finding equilibrium points in n -person games, reduced to finding the normalized equilibrium points. The convergence of the methods is proved and geometrical estimates of the convergence rate are given. A chronic drawback of the cutting plane methods (and in particular, the Chebyshev point methods) is the increase (every step adds a linear constraint) of the dimension in an auxiliary linear programming problems.

Many papers (see, for example, [6,12]) are devoted to reductions of solving the initial problem to solving a sequence of linear programming problems with fixed dimensions. As a rule, the proof of the modified methods convergence needs uniform or strong convexity of the objective function [6] or existence of a strongly convex function, which may not coincide with the objective [3].

Use of points other than the Chebyshev points for simplices allows on to construct a polynomial time algorithms, the convergence rate of which depends only on the space dimension [10]. Moreover, by inscribing a sequence of maximum volume ellipsoids into a sequence of polyhedrons containing a solution of the initial problem one can reach an convergence rate, which does not depend on the space dimension [11]. The auxiliary problems in this case are essentially harder.

Consider now the general convex programming problem

$$\min\{\varphi(x) : x \in R\}, \quad (1.9)$$

$$R = \{x : q_j(x) \leq 0, \quad j = \overline{1, m}\}, \quad (1.10)$$

where $\varphi(x), q_j(x)$ are pseudoconvex functions, R satisfies regularity conditions. Assume that R^0 is a convex polyhedron, $x^* \in R^0$, and a starting point $x^0 \in R^0$ is given. Find

$$\alpha^0 = \begin{cases} \nabla\varphi(x^0), & \text{if } \max_j q_j(x^0) = q_{j_0}(x^0) \leq 0 \\ \nabla q_{j_0}(x^0), & \text{if } \max_j q_j(x^0) = q_{j_0}(x^0) > 0 \end{cases}$$

and determine $R^1 = \{x, x_{n+1} : \alpha^{0T}(x - x^0) \leq \|\alpha^0\| x_{n+1}, x \in R^0\}$. Then $x^1 = \operatorname{argmin}\{x_{n+1} : (x, x_{n+1}) \in R^1\}$.

Assume that x^k, x_{n+1}^k have been found and R^k has been constructed. Determine

$$\alpha^k = \begin{cases} \nabla\varphi(x^k), & \text{if } \max_j q_j(x^k) \leq 0 \\ \nabla q_{jk}(x^k), & \text{if } \max_j q_j(x^k) = q_{jk}(x^k) > 0 \end{cases} \quad (1.11)$$

and the convex polyhedron

$$R^{k+1} = \{x, x_{n+1} : x \in R^k, \alpha^{kT}(x - x^k) \leq \|\alpha^k\| x_{n+1}\}. \quad (1.12)$$

Then the next approximation $\{x^{k+1}\}$ is found from solving the linear programming problem

$$x^{k+1} = \operatorname{argmin}\{x_{n+1} : x, x_{n+1} \in R^{k+1}\}. \quad (1.13)$$

Convergence of the iterative process (1.11)-(1.13) is proved in [12]. The Chebyshev point methods are assigned to the class of central cutting methods, but in contrast to other wellknown methods such as the ellipsoid method [10] and the simplex imbedding method [1] the convergence rate of the Chebyshev point methods depends on problem structure, initial data. Apparently, the further investigations in this field can be performed in the three directions.

A. Constructing an iterative processes of the type (1.11)-(1.13) with fixed number (say no more, than $(n + 1)$) of additional constraints, i.e. with fixed number of inequalities determining the set R^k .

B. Constructing an iterative processes of the type (1.11)-(1.13) with additional constraints of special kinds like $x_i \leq \bar{x}_i$ or $x_i \geq \underline{x}_i$. This will give a possibility to keep the structure of the linear programming problem and, hence, reduce computational efforts at the next iteration.

C. Constructing an iterative processes of the type (1.11)-(1.13) with such a convergence rate that does not depend on initial data, structure of the problem or other intermediate information.

1.3. Some Modifications of the Chebyshev Point Method

Let us pass now to a new version of the Chebyshev point method for solving the general convex programming problem of the type (1.9), (1.10). Let

$$R^k = \{x : A^k x \leq b^k\} \ni x^* \quad (1.14)$$

be a simplex containing x^* , x^k be the Chebyshev center of R^k with

$$\sum_{j=1}^n a_{ij}^k x_j - b_i^k = \|a^{ki}\| x_{n+1}, \quad i = \overline{1, n+1}, \quad (1.15)$$

a^{ki} be a i - row of matrix A^k , and $\alpha^{kT}(x - x^k) = 0$ be a cutting plane,

$$\alpha^k = \begin{cases} \nabla\varphi(x^k), & \text{if } \max_j q_j(x^k) = q_{jk}(x^k) \leq 0, \quad \forall j = \overline{1, m} \\ \nabla q_{jk}(x^k), & \text{if } \max_j q_j(x^k) = q_{jk}(x^k) > 0 \end{cases} \quad (1.16)$$

Find

$$\bar{x}^k = \operatorname{arg min}\{\alpha^{kT} x : x \in R^k\}. \quad (1.17)$$

The auxiliary problem (1.17) is a special linear programming problem over a simplex. Exactly n constraints, which form R^k are active at the point x^k . Adding to them the constraint

$\alpha^{kT}(x - x^k) \leq 0$, we obtain a new simplex R^{k+1} . The vertex \bar{x}^k will be called a base vertex of the truncated simplex. Find a new approximation x^{k+1} (which is the Chebyshev point of R^{k+1}) from the system (1.15) of linear equations. It is not difficult to see that $R^{k+1} \ni x^* = \arg \min\{\varphi(x) : x \in R\}$, $\forall k = 1, 2, \dots$. Moreover, the radii of the Chebyshev spheres (providing that the Haar condition is true monotonously decrease, i.e. $x_{n+1}^k > x_{n+1}^{k+1}$ and

$$\lim_{k \rightarrow \infty} x_{n+1}^k = 0 \quad \text{and} \quad \lim_{k \rightarrow \infty} \varphi(x^k) = \varphi^*$$

Let us make some comments about computational complexity of the iteration. At each step of the iterative process two problems are solved: (1) the problem of minimization of a linear function over a simplex (2) solving the linear algebraic equations system of order $(n+1) \times (n+1)$ of the type (1.15) (finding the Chebyshev point). The sequence of matrices determining the simplices and systems of linear algebraic equations, differ from step to step only by one row, i.e. the computational complexity of the iteration is of order n^2 .

Let us pass now to the investigation of the computational part of the method. Let R^k be the k -th iteration simplex determined by its vertices $\{x^{0k}, x^{1k}, \dots, x^{nk}\}$, x^k be the Chebyshev point of R^k , $\{x : \alpha^{kT}(x - x^k) \leq 0\}$ — be the cutting half-space. Then, it is not difficult to see that the base vertex \bar{x}^k solving system (1.15) is determined by the formula

$$\bar{x}^k = \arg \min_{0 \leq i \leq n} \alpha^{kT}(x^{ik} - x^k). \quad (1.18)$$

A vertex $x^{ik}, i \in 0, \dots, n$, is called an undeleted (by plane $\alpha^{kT}(x - x^k) = 0$) vertex if $\alpha_i = \alpha^{kT}(x^{ik} - x^k) < 0$ and deleted if $\alpha_i \geq 0$. From this definition it follows that $\sum_{i=0}^n \alpha_i = 0$ and the number of undeleted vertices k satisfies $1 \leq k \leq n$.

From geometrical considerations it is clear that $\bar{x}^k \in R^{k+1}$, where R^{k+1} is the next iteration simplex. Moreover, the next Chebyshev point \bar{x}^{k+1} belongs to the halfline $x = \bar{x}^k + \lambda(x^k - \bar{x}^k)$, $\lambda > 0$ and has equivalent distances to any plane (as well as to $\alpha^{kT}(x - x^k) = 0$) connected to \bar{x}^k .

Assume that vector α^k and row a^{jk} of matrix A^k in (1.11) are normalized by condition (1.5). Then x^k is the center of a ball inscribed into R^k and the condition of equidistance of x^{k+1} from planes of the simplex R^{k+1} has the following form:

$$a^{jkT}(\bar{x}^k + \lambda(x^k - \bar{x}^k)) - b_{jk} = \alpha^{kT}(\bar{x}^k + \lambda(x^k - \bar{x}^k)) - \beta_k, \quad (1.19)$$

where $\beta_k = \alpha^{kT}\bar{x}^k$, α^{jkT} is an arbitrary row of matrix A^k such that $a^{jkT}\bar{x}^k - b_{jk} = 0$. There are exactly n such rows. Since $\alpha^{jkT}\bar{x}^k - b_{jk} = 0$. we have from (1-19)

$$\lambda^k = \frac{-\alpha^{kT}\bar{x}^k + \beta_k}{|x_{n+1}^k| + \alpha^{kT}(x^k - \bar{x}^k)}.$$

Then the Chebyshev point x^{k+1} of the simplex R^{k+1} that satisfy (1.15) is determined by the formula

$$x^{k+1} = \bar{x}^k + \frac{-\alpha^{kT}\bar{x}^k + \beta_k}{|x_{n+1}^k| + (\alpha^k)^T(x^k - \bar{x}^k)}(x^k - \bar{x}^k). \quad (1.20)$$

Hence, the algorithm for solving problem (1.9), (1-10) consists in the following. Let R^k be a simplex, which contains the solution of problem (1.9), (1.10), x^k be its Chebyshev point,

$|x_{n+1}^k|$ be the radius of the maximum volume inscribed ball. Define the cutting half-space by formula (1.16). Then determine (according to (1.18)) the base vertex \bar{x}^k , which solves system (1.15) n constraints active at point \bar{x}^k and an additional constraint corresponding to the cutting half-space form the new simplex R^{k+1} , and its Chebyshev point is determined by formula (1.19).

By construction $|x_{n+1}^k| \geq |x_{n+1}^{k+1}|$. Assume that simplices R^k are contained in a ball of diameter $|D|$ and contain a ball of radius $|r|$. Clearly, the latter means that we restrict our efforts to an ε -optimal solution. Then the following theorem is true.

Theorem 1. The sequence r^k satisfies the condition

$$\frac{r_k}{r_{k-1}} \leq 1 + \frac{|r|}{|R|}.$$

Proof. Repeat the technique described in [1] for finding a common point of system of linear inequalities. By virtue of (1.5)

$$|x_{n+1}^{k+1}| = -a^{jkT} \left(\bar{x}^k + \frac{-\alpha^{kT}\bar{x}^k + \beta_k}{|x_{n+1}^k| + \alpha^{kT}(x^k - \bar{x}^k)}(x^k - \bar{x}^k) \right) + b_{jk}.$$

or due to $-a^{jkT}\bar{x}^k + b_{jk} = 0$ we have

$$\begin{aligned} |x_{n+1}^{k+1}| &= \frac{-\alpha^{kT}\bar{x}^k + \beta_k}{|x_{n+1}^k| + \alpha^{kT}(x^k - \bar{x}^k)} |x_{n+1}^k| = |x_{n+1}^k| \frac{-\alpha^{kT}\bar{x}^k + \beta_k}{|x_{n+1}^k| - \alpha^{kT}\bar{x}^k + \beta_k} = \\ &= |x_{n+1}^k| \frac{1}{1 + \frac{|x_{n+1}^k|}{[-\alpha^{kT}\bar{x}^k + \beta_k]}} \end{aligned}$$

or

$$\frac{|x_{n+1}^{k+1}|}{|x_{n+1}^k|} \leq \frac{1}{1 + \frac{|r|}{|R|}}.$$

2. APPLICATIONS OF THE CHEBYSHEV POINTS METHODS

2.1. The Chebyshev Points in Investigation of Viability of Systems

Main concepts discussed here are taken from the book [2], chapter 4 "Operations research and viability of controlled systems".

The term viability was introduced by Russian scientist and admiral S.O.Makarov in the 1870s in ship-building. In the ship-building this term reflected the possibility of a ship to resist some damages being able to flow. In a more general context viability means ability of a system to resist some perturbations.

In creating various technical schemes and objects essential attention is paid to a functional destination of the object called in [2] objective destination. It is usually determined by a state vector x . It is useful also to separate control variables u and perturbation variables v .

We will assume that elements x, u, v belong to some sets X, U, V . Under the objective destination we mean that some functions and operators depending on state, control and perturbation variables will take desirable values, i.e. have some prescribed properties. This means that an operator defined on set $R = X \times U \times V$ is given as well as the set Q of its

desirable values. Then the objective destination of the system consists in the fulfillment of the inclusion

$$F(x, u, v) \in Q$$

with $x, u, v \in R$.

It is obvious that there can exist $v^* \in V$ such that $\forall x \in X, u \in U F(x, u, v^*) \notin Q$, i.e. under some perturbations the system loses its objective destination. The idea is to choose a control and a corresponding to this control state x in a such way that the system can keep its objective destination over "maximal" in some sense subset of the set of perturbations.

Let us pass now to formal constructions. For a fixed $u \in U$ form set $V(u) \in V$ such that for any $v \in V(u)$ there exists $x \in X$ such that $F(x, u, v) \in Q$.

By definition $V(u)$ is a set of safe perturbations, i.e. perturbations for which there exists a control $u \in U$ and a state $x \in X$ that guarantee the objective destination of the system.

Obviously, the more "powerful" set of the safe controls, the higher viability of the system.

Let a measure μ be introduced on V . Then $\mu(V(u))$, the measure of the set of safe perturbations corresponds to a control u . To connect $V(u)$ and V introduce the real function

$$J(u) = \frac{\mu(V(u))}{\mu(V)}$$

If $J(u) = 1$ then sets $V(u)$ and V coincide with respect to the measure, i.e. the control suppress almost all perturbations. Conversely, if $J(u) = 0$ then the set of safe perturbations has zero measure and for the given control almost all perturbations destroy the objective destination of the system. The best control providing the most viability of the system, solves the problem

$$\min\{J(u) : u \in U\}$$

In order to realize this approach it is necessary to have an exact description of $V(u)$ (the set of safe perturbations), which usually implies essential computational difficulties.

The essence of the approach consists in approximation of $V(u)$ by a more simple set $\tilde{V}(w) \subset V(u)$ with easily calculating measure. In applications the sets of perturbations have as a rule simple structure (cubes, balls, ellipsoids). Therefore it is convenient to construct the parametric family $\tilde{V}(w)$, $w \in W \subset E^m$ from the same objects.

Since

$$J(w) = \frac{\mu(\tilde{V}(w))}{\mu(V)} \leq \frac{\mu(V(u))}{\mu(V)} = J(u),$$

$J(w)$ is a lower bound for viability.

Let us give some examples.

1. Consider the model of development of a system in the form

$$d(x) + C(x)v \leq b, \quad x \in R_x, \quad v \in R_v.$$

Here $d(x)$ and $C(x)$ are an m -vector and an $m \times s$ matrix with continuous elements $d_i(x), c_{ij}(x), R_v = \{v : 0 \leq v_i \leq 1, i = \overline{1, s}\}$.

We will search for $\tilde{V}(w)$ in the form

$$\tilde{V}(w) = \{v : w \leq v \leq 1 - w\}.$$

The problem of finding the maximal cube which is contained in the set of safe perturbations $V(x) = \{v \in R_0 : d(x) + C(x)v \leq b\}$ has the form

$$\begin{aligned} \max\{w : d(x) + C(x)v + N(x)w \leq b; \\ w \leq v \leq 1 - w, x \in R_x\} \end{aligned}$$

where

$$N_i(x) = \sum_{j=1}^s |c_{ij}(x)|, \quad i = \overline{1, m}, \quad N(x) = \{N_1(x), \dots, N_m(x)\}$$

or the standard form

$$\begin{aligned} \min\{v_{s+1} : d_i(x) + \sum_{j=1}^s c_{ij}v_j - b_i \leq \sum_{j=1}^s |c_{ij}(x)|v_{s+1}; \\ v_j - 1 \leq v_{s+1}, \quad -v_j \leq v_{s+1}, \quad x \in R_x \quad i = \overline{1, m}, \quad j = \overline{1, s}\}. \end{aligned}$$

2. A linear model of system development has the form

$$Ax + Cv \leq b, \quad v \in V = \{v : 0 \leq v \leq 1\} \quad (2.1)$$

where A, C are matrices of sizes $m \times n$ and $m \times s$.

Relate to linear model (2.1) the following linear programming problem

$$\begin{aligned} \min\{v_{s+1} : \sum_{j=1}^n a_{ij}x_j + \sum_{j=1}^s c_{ij}v_j - b_i \leq \sum_{j=1}^s |c_{ij}|v_{s+1}, \quad \forall i = \overline{1, m}, \\ v_j - 1 \leq v_{s+1}, \quad -v_j \leq v_{s+1}, \quad j = \overline{1, s}\}. \end{aligned} \quad (2.2)$$

The following theorem [2] obtained by L.T.Ashepkov is true.

Theorem 2. Let $\{x^*, v^*, v_{s+1}^*\}$ be a solution of the auxiliary problem (2.2). Then
1⁰. If $v_{s+1}^* < 0$, then the system of inequalities

$$Ax^* + Cv \leq b$$

is consistent for all v from a cube $\tilde{V}(v)$ inscribed into the polyhedron of safe perturbations

$$V(x) = \{v \in V : Ax + Cv \leq b, \quad x \in E^n\};$$

v^* is the center of the cube, $2|v_{s+1}^*|$ is the length of its edge.

2⁰. If $v_{s+1}^* = 0$, then there does not exist a pair x, v which satisfies the strict inequalities

$$Ax + Cv - b < 0, \quad 0 < v < 1.$$

3⁰. If $v_{s+1}^* > 0$, then the system is inconsistent.

Under substitution of vector b by the vector $b + 2v_{s+1}^*N$ the inequalities $Ax + Cv \leq b + 2v_{s+1}^*N$ are fulfilled for $x = x^*$ and any $v \in \tilde{V}(v^*, |v_{s+1}^*|)$.

2.2. Application of the Chebyshev Points in Global Optimization and Integer Programming

Consider the problem

$$\min\left\{\sum_i \varphi_i(x_i) : x \in R\right\} \quad (2.3)$$

where $R = \{x : Ax \leq b\}$.

Applying the above technique, inscribe into R the maximum volume parallelepiped with edges parallel to coordinate axes:

$$\max\{\lambda_1, \lambda_2, \dots, \lambda_n\} \quad (2.4)$$

subject to

$$\sum_{j=1}^n a_{ij}x_j + \sum_{j=1}^n |a_{ij}| \prod_j \lambda_j \leq b_i, \quad \forall i = \overline{1, m}, \lambda_j \geq 0. \quad (2.5)$$

Assume that $\{\lambda_1^*, \dots, \lambda_n^*, x_1^*, \dots, x_n^*\}$ solves problem (2.4), (2.5). Then x^* is the center of the maximal parallelepiped \underline{R}^* and $2\lambda_1^*, \dots, 2\lambda_n^*$ are lengths of its edges. Define the set $\underline{R}^0 = \{x : \alpha_i^0 \leq x_i \leq \beta_i^0\}$ where $\alpha_i^0 = x_i^* - \lambda_i^*$, $\beta_i^0 = x_i^* + \lambda_i^*$ and find

$$\varphi(x^0) = \min\left\{\sum_{i=1}^n \varphi_i(x_i) : x \in \underline{R}^0\right\}. \quad (2.6)$$

It is clear that problem (2.6) is equivalent to n one-dimensional problems $\min\{\varphi_i(x_i) : \alpha_i^0 \leq x_i \leq \beta_i^0\}$. Let φ^* solves problem (2.6). Since $\underline{R}^0 \subset R$ we have $\varphi^* \leq \varphi(x^0) = \underline{\varphi}^0$.

Let us find now a lower bound for φ^* . To do this imbed R in the minimal volume parallelepiped. Obviously it is equivalent to solving $2n$ linear programming problems: find

$$\underline{x}^{0i} = \operatorname{argmin}\{x_i : x \in R\}, \quad i = \overline{1, n}, \quad (2.7)$$

$$\overline{x}^{0i} = \operatorname{argmax}\{x_i : x \in R\}, \quad i = \overline{1, n}. \quad (2.8)$$

The set $\overline{R}^0 = \{x : \underline{x}^0 \leq x \leq \overline{x}^0\}$ contains R , therefore

$$\overline{\varphi}_0 = \min\left\{\sum_{i=1}^n \varphi_i(x_i); x \in \overline{R}^0\right\} \leq \varphi^*.$$

Since $\overline{R}^0 \supset R \supset \underline{R}^0$, we obtain the two sided estimate

$$\overline{\varphi}_0 \leq \varphi^* \leq \underline{\varphi}^0.$$

As $\underline{x}^{01}, \dots, \underline{x}^{0n}$, $\overline{x}^{01}, \dots, \overline{x}^{0n}$ are vertices of R we can sharpen the upper bound φ^0 of φ^* calculating the values $\varphi(x) = \sum \varphi_i(x_i)$ at these points and choosing after all the minimal one. This is especially important if $\varphi(x)$ is a concave function (then it reaches its minimum at a vertex). If $\underline{\varphi}^0 - \overline{\varphi}_0 \leq \varepsilon$ then any point where the upper or lower bound is achieved is taken as an approximate solution. Otherwise, perform the following subdivision of R . Determine $2n$ polyhedrons

$$\overline{R}^{0j} = \{x : x \in R, x_i \geq \underline{x}_i^0\}, \quad \forall i = \overline{1, n},$$

and

$$\underline{R}^{0j} = \{x : x \in R, x_i \leq \overline{x}_i^0\}, \quad \forall i = \overline{1, n}.$$

Obviously $R = \cup_j \underline{R}^{0j} \cup_j \overline{R}^{0j} \cup \underline{R}^0$.

In every set $\underline{R}^{0j}, \overline{R}^{0j}, j = \overline{1, n}$, inscribe (similar to (2.4), (2,5)) the maximum volume parallelepiped or cube $\underline{R}^{1j}, j = \overline{1, 2n}$. Solve $2n$ problems

$$\varphi^{1j} = \min\{\sum \varphi_i(x_i) : x \in R^{1j}\}, j = \overline{1, 2n}.$$

Improve the upper bound for φ^* :

$$\underline{\varphi}^1 = \min\{\underline{\varphi}^0, \varphi^{11}, \dots, \varphi^{12n}\}.$$

It is not difficult to see that $\underline{R}^0 \subset \underline{R}^1 = \cup_j \underline{R}^{1j} \cup \underline{R}^0 \subset R$ hence, $\varphi^0 \geq \varphi^1 = \min\{\sum_{i=1}^n \varphi_i(x_i) : x \in \underline{R}^1\} \geq \varphi^*$.

Improve now the lower bound for φ^* . To do this imbed every set $\overline{R}^{0j}, \underline{R}^{0j}, j = \overline{1, 2n}$, into the minimum volume parallelepiped \overline{R}^{1j} solving $(2n - 2) \times 2n$ linear programming problems of the type (2.7), (2,8).

Let

$$\overline{\varphi}^{1j} = \min\{\sum_{i=1}^n \varphi_i(x_i) : x \in \overline{R}^{1j}\}, j = \overline{1, 2n},$$

$$\overline{\varphi}^1 = \min\{\overline{\varphi}^{11}, \dots, \overline{\varphi}^{12n}\}.$$

Obviously $\overline{\varphi}^1 = \min\{\sum_{i=1}^n \varphi_i(x_i) : x \in \cup_{j=1}^{2n} \overline{R}^{1j} \cap \overline{R}^0\}$. Since $\overline{R}^0 \supset \overline{R}^1 = \cup_{j=1}^{2n} \overline{R}^{1j} \cap \overline{R}^0 \supset R \supset \underline{R}^1 \supset \overline{R}^0$ we have $\underline{\varphi}^0 \geq \underline{\varphi}^1 \geq \varphi^* \geq \overline{\varphi}^1 \geq \overline{\varphi}^0$.

Continue this process and obtain a sequence of imbedded sets

$$\overline{R}^0 \supset \overline{R}^1 \supset \dots \supset \overline{R}^k \supset \dots \supset R \supset \dots \supset \underline{R}^k \supset \dots \supset \underline{R}^1 \supset \underline{R}^0.$$

Each of these sets is a union of parallelepipeds over which the additive objective function can be easily minimized. Moreover, the volumes $|R^k|$ of sets \overline{R}^k satisfy the strict inequalities

$$|\overline{R}^0| > |\overline{R}^1| > \dots > |\overline{R}^k| > \dots > |R| > \dots > |\underline{R}^k| \dots > |\underline{R}^1| > |\underline{R}^0|.$$

Therefore, the method has the two-sided estimate

$$\underline{\varphi}^k \geq \varphi^* \geq \overline{\varphi}^k,$$

which can be used as a constructive stopping rule. Consider now concave separable programming. Since a concave function reaches its minimum at a vertex, the minimization of $\sum_{i=1}^n \varphi_i(x_i)$ becomes essentially simpler and demands only two evaluations of each function $\varphi_i(x_i)$.

In order to improve the upper bound one can use also simplicial subdivisions which make possible not to solve the problems of the type (2.7), (2,8) at every iteration for constructing a sequence of the minimum volume parallelepipeds.

In some cases the suggested approach can be used for minimization non-separable functions. Consider, for instance, the quadratic programming problem:

$$\min\{\varphi_i(x_i) = c^T x - x^T B x : Ax \leq b\} \quad (2.9)$$

here the matrix B is positively defined. One of the important properties of quadratic functions is that it can be represented in a separable form by a linear transformation. Let $U = (u^1, u^2, \dots, u^n)$ be the matrix of normalized eigenvectors of B ,

$$U^T B U = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_n) > 0.$$

After the substitution $x = U y$, $f(y) = \varphi(U y)$ we have

$$f(y) = p^T y - y^T (U^T B U) y = \sum_{i=1}^n f_i(y_i)$$

here

$$f_i(y_i) = p_i y_i - \lambda_i y_i^2, \quad p = U^T c.$$

Hence, problem (2.9) is equivalent to the problem

$$\min \left\{ \sum_{i=1}^n f_i(y_i) : y \in R \right\}$$

where

$$R = \{y : A U y \leq b\}.$$

The above-described technique can be used for solving the latter problem.

Turn now to integer programming. Consider the problem

$$\min \left\{ \sum_{i=1}^n \varphi_i(x_i) : x \in R, x_i \text{ is integer} \right\} \quad (2.10)$$

$$R = \{x : A x \leq b\}.$$

Main operations of the method consist in solving linear programming problems of the type (2.7), (2.8) or (2.4), (2.5) for inner and outer approximation of the set R by a union of parallelepipeds and problems of global optimization of $\sum \varphi_i(x_i)$ on a sequence of parallelepipeds.

The first of these operations is invariant with respect to the optimization problems. Obviously solving the second one is even easier due to the presence of the integer net. Moreover, the problem of maximization of $\varphi_i(x_i)$ over the interval $\underline{x}_i^k \leq x_i \leq \bar{x}_i^k$ (which corresponds to the outer approximation) always has a solution if the initial is solvable. It is also clear that the problem of minimization of $\varphi_i(x_i)$ over inner approximation is not always solvable since a union of inner parallelepipeds may not contain an integer point. However, as soon as the approximation becomes more exact the inner problem becomes also solvable. In other details the technique for solving problem (2.10), (2.11) repeats the one described above.

Список литературы

- [1] Aleksandrov, I. A., Antsiferov, E.G., and Bulatov, V.P. *To Centered Cuttings Methods*, Proceedings of the Conference on Mathematical Programming, Sverdlovsk, IMM UNC AN SSSR, 1981, pp.162-163 (in Russian).

- [2] Antsiferov, E.G., Ashchepkov, L.T., and Bulatov, V.P. *Methods of Optimization. V.I Mathematical programming*, Novosibirsk, Nauka, 1990 (in Russian).
- [3] Bulatov, V.P. *Approximation Methods for Solving Some Mathematical Programming Problems*, in Applied Mathematics, Irkutsk, 1969, pp. 10-15 (in Russian).
- [4] Bulatov, V.P. *Approximation Methods for Solving Some Extremal Problems*, PhD. Dissertation, University of Tomsk, 1967 (in Russian).
- [5] Zukhovitskii, S.I., Polyak, R.A., and Primak, M.E. *Two methods of Joining Equilibrium Points of Concave n-Person Games*, Reports of Academy of Sciences of USSR, v.185, N 1, 1969, pp. 24-27 (in Russian).
- [6] Nenakhov, E.I. and Primak, M.E. *Estimation of the Efficiency of the Dual Simplex Method for Finding a Solution of Linear Inequalities Systems*, Kibernetika, Kiev, 1989, N2, pp.1-4, (in Russian).
- [7] Nenakhov, E.I. and Primak, M.E. *On Convergence of the Chebyshev Center Method and Some Its Applications*, Kibernetika, 1986, N2 (in Russian).
- [8] Primak, M.E. *On Convergence of the Modified Chebushev Center Method for Solving Concave Programming Problems*, Kibernetika, N5, 1977, pp.119-121 (in Russian).
- [9] Primak, M.E. *On Convergence of the Cutting Method with Refinement in Each Stage*, Kibernetika, N1, 1980, pp.119-121 (in Russian).
- [10] Khachiyan, L.G. *A Polynomial Algorithm in Linear Programming*, Doklady Akademii Nayk SSSR, 1979, v.244, N5, pp.1093-1096 (in Russian).
- [11] Khachiyan, L.G., Tarasov, S.P., and Erlikh, I.I. *Method of Inscribed Ellipsoids*, Doklady Akademii Nayk SSSR, 1987, v.298, N5, pp.1081-1085 (in Russian).
- [12] Elzinga, J. and Moore, T.G. *A Central Cutting Plane Algorithm for the Convex Programming Problems*, Mathematical Programming, v.8, 1975, pp. 134-145.

OPTIMALITY CONDITIONS FOR BILEVEL PROGRAMMING PROBLEMS

S. Dempe

*Department of Mathematics and Computer Science, Technical University Bergakademie
Freiberg, Germany
e-mail:*

Abstract. Bilevel programming problems are hierarchical optimization problems of two players: the so-called leader who makes the first choice and the so-called follower reacting optimally on the leader's selection. Goal of the leader is to find a best selection depending both on his own constraints and on the reaction of the follower. This paper starts with recalling some definitions and properties of the problem as well as possible reformulations. Then, some necessary and sufficient optimality conditions are developed.

Key words: bilevel programming, necessary and sufficient optimality conditions.

Introduction

Consider a parametric optimization problem

$$\Psi(x) := \underset{y}{\operatorname{Argmin}} \{f(x, y) : g(x, y) \leq 0\}, \quad (1)$$

where $f, g_i : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}$, $i = 1, \dots, p$. Then, the bilevel programming problem reads as

$$\min_{x, y} \{F(x, y) : y \in \Psi(x), x \in X\}, \quad (2)$$

with $F : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}$ and a closed set $X \subseteq \mathbb{R}^n$.

To give an interpretation of the problem consider two decision makers which cannot act independently but are forced to pay attention to the selection of the other one. Moreover, they do not make their choices at the same time as in non-cooperative games but according to a certain hierarchy. After being aware of the decision x of the first decision maker (the so-called leader) the second one (the follower) makes his choice by solving an optimization problem, say (1), parameterized in x and announces this decision to the first decision maker. Being fully informed about all the decisions of the follower, the leader has to find best values $x \in X$ such that this selection together with the response of the follower minimizes the function $F(x, y)$.

This problem has many applications, see e. g. the bibliography [6]. For more information about the problem see e. g. the monographs [3, 5]. Strongly speaking it should be remarked that formulation (2) reflects the optimistic position where the leader assumes that the follower will cooperate in the sense that he allows the leader to take that solution out of $\Psi(x)$ which is a best one from the leader's point of view. The opposite position is the pessimistic one where the leader is not able to count for cooperation of the follower and is thus forced to bound the damage caused by an arbitrary selection of the follower, see [5].

The following definition concerns local optimal solutions of the problem (2). Here,

$$M = \{(x, y)^\top : x \in X, y \in \Psi(x)\}$$

is the feasible set of problem (2).

Definition 1 A point $(\bar{x}, \bar{y}) \in M$ is a locally optimal solution of problem (2) provided that there is an $\varepsilon > 0$ such that $F(x, y) \geq F(\bar{x}, \bar{y})$ for all $(x, y) \in M$ with $\|(x, y) - (\bar{x}, \bar{y})\| \leq \varepsilon$.

The bilevel programming problem is a nonconvex, nondifferentiable optimization problem with implicitly determined feasible set. The problem is \mathcal{NP} -hard [8].

1. Existence of optimal solutions

Using this definition it is possible to determine assumptions guaranteeing the existence of locally optimal solutions [5]. Besides others we will need the following two assumptions:

(C) The set

$$\{(x, y)^\top : x \in X, g(x, y) \leq 0\}$$

is nonempty and bounded.

(MFCQ) The Mangasarian-Fromowitz constraint qualification is satisfied at a point (\bar{x}, \bar{y}) if there is a direction d such that

$$\nabla_y g_i(\bar{x}, \bar{y})d < 0, \forall i \in \{j : g_j(\bar{x}, \bar{y}) = 0\}.$$

A point-to-set mapping $\Gamma : \mathbb{R}^p \rightarrow 2^{\mathbb{R}^q}$ maps points $w \in \mathbb{R}^p$ to sets $\Gamma(w) \subseteq \mathbb{R}^q$.

Definition 2 A point-to-set mapping $\Gamma : \mathbb{R}^p \rightarrow 2^{\mathbb{R}^q}$ is said to be upper semicontinuous at a point $\bar{w} \in \mathbb{R}^p$ if for each open set $A \supseteq \Gamma(\bar{w})$ there is an open set $V \ni \bar{w}$ such that $\Gamma(w) \subseteq A$ for all $w \in V$. The point-to-set mapping Γ is lower semicontinuous at $\bar{w} \in \mathbb{R}^p$ provided that for each open set $A \subseteq \mathbb{R}^q$ with $\Gamma(\bar{w}) \cap A \neq \emptyset$ there is an open set $V \ni \bar{w}$ with $\Gamma(w) \cap A \neq \emptyset$ for all $w \in V$.

Theorem 1 ([10]) A locally optimal solution of the bilevel programming problem exists provided $M \neq \emptyset$, the point-to-set mapping $\Psi(\cdot)$ is upper semicontinuous at all points $x \in X$ and assumption (C) is satisfied.

It should be mentioned that the point-to-set mapping $\Psi(\cdot)$ is upper semicontinuous at a point $\bar{x} \in X$ if (C) and (MFCQ) are satisfied at all points (\bar{x}, \bar{y}) with $\bar{y} \in \Psi(\bar{x})$ [2].

If, moreover, the strong sufficient optimality condition of second order is satisfied, we get even stronger results.

(SSOC) The strong second-order sufficient optimality condition for problem (1) is satisfied at a point $(\bar{x}, \bar{y}) \in M$ if:

1. The set

$$\Lambda(\bar{x}, \bar{y}) := \{\lambda : \lambda \geq 0, \lambda^\top g(\bar{x}, \bar{y}) = 0, \nabla_y L(\bar{x}, \bar{y}, \lambda) = 0\}$$

is not empty and

2. for all $\bar{\lambda} \in \Lambda(\bar{x}, \bar{y})$ and for all $d \neq 0$ with

$$\nabla_y g_i(\bar{x}, \bar{y})d = 0 \forall i : \bar{\lambda}_i > 0$$

there is

$$d^\top \nabla_{yy}^2 L(\bar{x}, \bar{y}, \bar{\lambda})d > 0.$$

Here, $L(x, y, \lambda) = f(x, y) + \lambda^\top g(x, y)$ is the Lagrange function of problem (1).

It is well-known that validity of (MFCQ) at a locally optimal solution (\bar{x}, \bar{y}) of (1) implies that $\Lambda(\bar{x}, \bar{y}) \neq \emptyset$ is bounded. If (MFCQ) and (SSOC) are satisfied at a point $(\bar{x}, \bar{y}) \in M$, then this point is strongly stable in the sense of Kojima [11], i.e. there exist open neighborhoods U of x and V of y and a continuous function $x : U \rightarrow V$ with $y(x)$ being the unique locally optimal solution of (1) in V for all $x \in U$.

This implies that the bilevel programming problem (2) can be reduced to the problem

$$\min_x \{F(x, y(x)) : x \in X\}, \quad (3)$$

provided that (1) is a convex parametric optimization problem for which (MFCQ) and (SSOC) are satisfied for $y \in \Psi(x)$. Problem (3) has then a continuous, but implicitly determined objective function and, consequently, an optimal solution whenever $X \neq \emptyset$ is compact.

Problem (3) is a Lipschitz optimization problem provided that the assumption (CRCQ) is also valid in addition to the above assumptions (MFCQ) and (SSOC) [17].

(CRCQ) The constant rank constraint qualification is satisfied for the problem (1) at the point (\bar{x}, \bar{y}) with $g(\bar{x}, \bar{y}) \leq 0$ if there exists an open neighborhood V of (\bar{x}, \bar{y}) such that for each subset $J \subseteq \{i : g_i(\bar{x}, \bar{y}) = 0\}$ the set of gradients

$$\{\nabla_y g_i(\bar{x}, \bar{y}) : i \in J\}$$

has constant rank on V .

2. Reformulations

Applying the Karush-Kuhn-Tucker conditions to the lower level problem (1) in (2) we derive the problem

$$\begin{aligned} F(x, y) &\rightarrow \min_{x, y, u} \\ \nabla_y L(x, y, u) &= 0 \\ g(x, y) &\leq 0 \\ u &\geq 0 \\ u^\top g(x, y) &= 0 \\ x &\in X \end{aligned} \quad (4)$$

provided that the lower level problem satisfies the (MFCQ) at all feasible points for all $x \in X$ and that it is a convex optimization problem for fixed $x \in X$. Problem (4) is called a mathematical program with equilibrium constraints [12, 15]. Optimality conditions for such problems can be found e. g. in the paper [18].

Example [13]: This example shows that the convexity assumption is crucial.

Consider the problem

$$\min_{x, y} \{(x - 2) + (y - 1)^2 : y \in \Psi(x)\}$$

where $\Psi(x)$ is the set of optimal solutions of the following unconstrained optimization problem on the real axis:

$$-x \exp\{-(y + 1)^2\} - \exp\{-(y - 1)^2\} \rightarrow \min_y$$

Then, the necessary optimality conditions for the lower level problem are

$$x(y + 1) \exp\{-(y + 1)^2\} + (y - 1) \exp\{-(y - 1)^2\} = 0$$

which has three solutions for $0.344 \leq x \leq 2.903$. The global optimum of the lower level problem is uniquely determined for all $x \neq 1$ and it has a jump at the point $x = 1$. Here the global optimum of the lower level problem can be found at the points $y = \pm 0.957$. The point $(x^0; y^0) = (1; 0.957)$ is also the global optimum of the optimistic bilevel problem.

But if the lower level problem is replaced with its necessary optimality conditions and the necessary optimality conditions for the resulting problem are solved then three solutions: $(x, y) = (1.99; 0.895)$, $(x, y) = (2.19; 0.42)$, $(x, y) = (1.98; -0.98)$ are obtained. Surprisingly, the global optimal solution of the bilevel problem is not obtained with this approach. The reason for this is that the problem

$$\min\{(x - 2) + (y - 1)^2 : x(y + 1) \exp\{-(y + 1)^2\} + (y - 1) \exp\{-(y - 1)^2\} = 0\}$$

has a much larger feasible set than the bilevel problem. And this feasible set has no jump at the point $(x, y) = (1; 0.957)$ but is equal to a certain connected curve in \mathbb{R}^2 . And on this curve the objective function has no stationary point at the optimal solution of the bilevel problem. \square

Problem (4) has an equivalent formulation using a nondifferentiable constraint [18]:

$$\begin{aligned} F(x, y) &\rightarrow \min_{x, y, u} \\ \nabla_x L(x, y, u) &= 0 \\ \min\{-g_i(x, y), u_i\} &= 0, \quad i = 1, \dots, p \\ x &\in X \end{aligned} \tag{5}$$

Problem (4) is a smooth optimization problem having more variables than the original one (2). Unfortunately, no regularity condition as e.g. the (MFCQ) can be valid at a feasible point of this problem.

To pose a second reformulation of the bilevel programming problem consider the optimal value function of the lower level problem

$$\varphi(x) = \min_y \{f(x, y) : g(x, y) \leq 0\}$$

for $x \in X$. Then,

$$\begin{aligned} F(x, y) &\rightarrow \min_{x, y} \\ g(x, y) &\leq 0 \\ f(x, y) &\leq \varphi(x) \\ x &\in X \end{aligned} \tag{6}$$

is a second optimization problem being equivalent to (2) [19]. This problem has, in general, a nondifferentiable constraint. To formulate it no assumption on the original problem (2) is necessary.

3. Optimality conditions

Optimality conditions for bilevel programming problems can be based on different reformulations of the problem as well as on the original formulation (2) itself.

3.1. Conditions using the contingent cone

The contingent (or Bouligand) cone of the set M is defined via

$$C_M(x, y) := \left\{ (u, v)^\top : \exists \{t_k\}_{k=1}^\infty \subset \mathbb{R}_+, \exists \{(u^k, v^k)^\top\}_{k=1}^\infty \subset \mathbb{R}^n \times \mathbb{R}^m, \right. \\ \left. \text{with } (x, y)^\top + t_k(u^k, v^k)^\top \in \text{grph } \Psi \ \forall k, \ x + t_k u^k \in X, \right. \\ \left. \lim_{k \rightarrow \infty} t_k = 0, \ \lim_{k \rightarrow \infty} (u^k, v^k)^\top = (u, v)^\top \right\}$$

where $\text{grph } \Psi = \{(x, y)^\top : y \in \Psi(x)\}$ denotes the graph of the point-set-mapping Ψ .

Theorem 2 *If the point $(\bar{x}, \bar{y})^\top \in M$ is a locally optimal solution of the bilevel programming problem (2), then*

$$\nabla F(\bar{x}, \bar{y})(d, r)^\top \geq 0$$

for all

$$(d, r)^\top \in C_M(x, y).$$

On the other hand if $(\bar{x}, \bar{y})^\top \in M$ and

$$\nabla F(\bar{x}, \bar{y})(d, r)^\top > 0$$

for all

$$(d, r)^\top \in C_M(x, y),$$

then the point $(\bar{x}, \bar{y})^\top$ is a locally optimal solution of (2).

Proof: i) Let $(\bar{x}, \bar{y})^\top \in M$ be a locally optimal solution of problem (2). Assume that the proposition of part a) of the theorem is not satisfied. Then, there exists a direction $(d, r)^\top$ with

$$(d, r)^\top \in C_M(\bar{x}, \bar{y})$$

and

$$\nabla F(\bar{x}, \bar{y})(d, r)^\top < 0. \tag{7}$$

Hence, by definition there are two sequences $\{t_k\}_{k=1}^\infty \subset \mathbb{R}_+$ and $\{(u^k, v^k)^\top\}_{k=1}^\infty \subset \mathbb{R}^n \times \mathbb{R}^m$ with $(\bar{x}, \bar{y})^\top + t_k(u^k, v^k)^\top \in \text{grph } \Psi$, $\bar{x} + t_k u^k \in X$, $\forall k \in \mathbb{N}$, $\lim_{k \rightarrow \infty} t_k = 0$, $\lim_{k \rightarrow \infty} (u^k, v^k)^\top = (d, r)^\top$.

Thus, using the definition of the derivative we get

$$F(\bar{x} + t_k u^k, \bar{y} + t_k v^k) = F(\bar{x}, \bar{y}) + t_k \nabla F(\bar{x}, \bar{y})(u^k, v^k)^\top + o(t_k)$$

for each sufficiently large k , where $\lim_{k \rightarrow \infty} \frac{o(t_k)}{t_k} = 0$. Since

$$\lim_{k \rightarrow \infty} \left\{ \nabla F(\bar{x}, \bar{y})(u^k, v^k)^\top + \frac{o(t_k)}{t_k} \right\} = \nabla F(\bar{x}, \bar{y})(d, r)^\top < 0$$

by the assumption this implies

$$\nabla F(\bar{x}, \bar{y})(u^k, v^k)^\top + \frac{o(t_k)}{t_k} < 0$$

for all sufficiently large k and, hence,

$$F(\bar{x} + t_k u^k, \bar{y} + t_k v^k) < F(\bar{x}, \bar{y})$$

for large k . This leads to a contradiction to local optimality.

ii) Now, let $\nabla F(\bar{x}, \bar{y})(d, r)^\top > 0$ for all $(d, r)^\top \in C_M(\bar{x}, \bar{y})$ and assume that there is a sequence $(x^k, y^k) \in M$ converging to $(\bar{x}, \bar{y})^\top$ with $F(x^k, y^k) < F(\bar{x}, \bar{y})$ for all k . Then,

$$\left(\frac{x^k - \bar{x}}{\|(x^k, y^k) - (\bar{x}, \bar{y})\|}, \frac{y^k - \bar{y}}{\|(x^k, y^k) - (\bar{x}, \bar{y})\|} \right)^\top$$

converges to some $(d, r)^\top \in C_M(\bar{x}, \bar{y})$. Using differential calculus, it is now easy to verify that

$$\nabla F(\bar{x}, \bar{y})(d, r)^\top \leq 0$$

contradicting our assumption. □

3.2. The linear case

If bilevel programming problems with linear lower level problems are under consideration, a stronger formulation of the optimality condition is possible [7]. For this, consider a linear parametric optimization problem

$$\max_y \{c^\top y : Ay = b, y \geq 0\} \quad (8)$$

with an (m, n) -matrix A and parameters in the right-hand side as well as in the objective function. Let $\Psi_L(b, c)$ denote the set of optimal solutions of (8). A special optimistic bilevel programming problem reads as

$$\min_{y, b, c} \{f(y) : Bb = \tilde{b}, Cc = \tilde{c}, y \in \Psi_L(b, c)\}. \quad (9)$$

Using linear programming duality, problem (9) has a reformulation as

$$\begin{aligned} f(y) &\longrightarrow \min_{y, b, c, u} \\ Ay &= b \\ y &\geq 0 \\ A^\top u &\geq c \\ y^\top (A^\top u - c) &= 0 \\ Bb &= \tilde{b} \\ Cc &= \tilde{c}. \end{aligned} \quad (10)$$

It should be noted that the objective function in the upper level problem does not depend on the parameters of the lower level one. This makes a more precise definition of a locally optimal solution of problem (9) necessary:

Definition 3 *A point \bar{y} is a locally optimal solution of problem (9) if there exists an open neighborhood U of \bar{y} such that $f(\bar{y}) \leq f(y)$ for all y, b, c with $Bb = \tilde{b}$, $Cc = \tilde{c}$ and $y \in U \cap \Psi_L(b, c)$.*

The main result of this definition is the possibility to drop the explicit dependence of the solution of problem (9) on c . This dependence rests on solvability of the dual problem and is guaranteed for index sets I in the set $\mathcal{I}(y)$.

Let the following index sets be determined at some point \bar{y} :

1. $I(\bar{y}) = \{i : \bar{y}_i = 0\}$,
2. $I(u, c) = \{i : (A^\top u - c)_i > 0\}$
3. $\mathcal{I}(\bar{y}) = \{I(u, c) : A^\top u \geq c, (A^\top u - c)_i = 0 \ \forall i \notin I(\bar{y}), Cc = \tilde{c}\}$
4. $I^0(\bar{y}) = \bigcap_{I \in \mathcal{I}(\bar{y})} I$.

Using these definitions, problem (10) can be transformed into the following one by replacing the complementarity conditions:

$$\begin{aligned}
f(y) &\longrightarrow \min_{y, b, I} \\
Ay &= b \\
y &\geq 0 \\
y_i &= 0 \quad \forall i \in I \\
Bb &= \tilde{b} \\
I &\in \mathcal{I}(y).
\end{aligned} \tag{11}$$

In this problem, the index set I enters the variables in the sense that problem (11) is to be solved for all fixed sets $I \in \mathcal{I}(y)$ and the best solution over all resulting problems is selected.

The contingent cone to the feasible set of the last problem (or rather its projection onto \mathbb{R}^m) is

$$C(\bar{y}) := \bigcup_{I \in \mathcal{I}(\bar{y})} C_I(\bar{y}),$$

where

$$C_I(\bar{y}) = \{d \mid \exists r : Ad = r, Br = 0, d_i \geq 0 \ \forall i \in I(\bar{y}) \setminus I, d_i = 0 \ \forall i \in I\}$$

for all $I \in \mathcal{I}(\bar{y})$.

Theorem 3 (Optimality condition) *If f is differentiable at \bar{y} , this point is a local optimum of (9) if and only if $\nabla f(\bar{y}) \cdot d \geq 0$ for all $d \in \text{conv } T(\bar{y})$.*

The obvious proof of this theorem is omitted. It should be noted that this theorem is no longer true if the differentiability assumption is dropped.

For an efficient verification of the condition in Theorem 3 a compact formula for the convex hull of the contingent cone of the feasible set is crucial. For that consider the relaxed problem of (11)

$$\begin{aligned}
f(y) &\longrightarrow \min_{y, b} \\
Ay &= b \\
y_i &\geq 0 \quad i = 1, \dots, l \\
y_i &= 0 \quad i = l + 1, \dots, k \\
Bb &= \tilde{b}
\end{aligned} \tag{12}$$

together with the contingent cone to its feasible set (or better: its projection onto \mathbb{R}^m)

$$C_R(\bar{y}) = \{d \mid \exists r : Ad = r, Br = 0, d_i \geq 0, i = 1, \dots, l, d_i = 0, i = l + 1, \dots, k\}$$

at the point \bar{y} . Here, it is assumed that $I(\bar{y}) = \{1, \dots, k\}$ and $I^0(\bar{y}) = \{l + 1, \dots, k\}$.

Remark: We have $j \in I(\bar{y}) \setminus I^0(\bar{y})$ if and only if the system

$$\begin{aligned} (A^\top u - c)_i &= 0 & \forall i \notin I(\bar{y}) \\ (A^\top u - c)_j &= 0 \\ (A^\top u - c)_i &\geq 0 & \forall i \in I(\bar{y}) \setminus \{j\} \\ Cc &= \tilde{c} \end{aligned}$$

has a solution.

In the following theorem we need an assumption: The point \bar{y} is said to satisfy the *full rank condition* (FRC), if

$$\text{span}(\{A_i : i = k + 1, \dots, n\}) = \mathbb{R}^m, \quad (13)$$

where A_i denotes the i th column of the matrix A .

Theorem 4 ([7]) *Let (FRC) be satisfied at the point \bar{y} . Then,*

$$\text{conv } C(\bar{y}) = \text{cone } C(\bar{y}) = C_R(\bar{y}). \quad (14)$$

Proof: The inclusion

$$\text{conv } C(\bar{y}) = \text{cone } C(\bar{y}) \subseteq C_R(\bar{y}) \quad (15)$$

is obviously satisfied. Let \bar{d} be an arbitrary element of $C_R(\bar{y})$, that means there is a \bar{r} with $A\bar{d} = \bar{r}$, $B\bar{r} = 0$, $\bar{d}_i \geq 0 \quad i = 1, \dots, l$, $\bar{d}_i = 0 \quad i = l + 1, \dots, k$. We consider the following linear systems

$$\begin{array}{ll} Ad = \bar{r} & Ad = 0 \\ d_1 = \bar{d}_1 & d_j = \bar{d}_j \\ d_i = 0, \quad i = 2, \dots, k & d_i = 0, \quad i = 1, \dots, k, i \neq j \end{array} \quad (S_1) \text{ and } \quad (S_j)$$

for $j = 2, \dots, l$. These systems are all feasible because of (FRC).

Furthermore let d^1, \dots, d^l be (arbitrary) solutions of the systems (S_j) . We define now the direction $d = \sum_{j=1}^l d^j$ and get $d_i = \bar{d}_i$ for $i = 1, \dots, k$ as well as $Ad = A\bar{d} = \bar{r}$. Because we chose arbitrary vectors d^1, \dots, d^l it is possible that $d \neq \bar{d}$. But we can achieve equality with a translation of the solution d^1 by a specific vector of $\mathcal{N}(A) = \{z : Az = 0\}$. Therefore we define $\hat{d}^1 := d^1 + \bar{d} - d$, and because d^1 is feasible for (S_1) and $d_i = \bar{d}_i$ for $i = 1, \dots, k$ as well as $Ad = A\bar{d} = \bar{r}$ we get $\hat{d}_i^1 = 0$ for all $i = 2, \dots, k$ and $A\hat{d}^1 = A(d^1 + \bar{d} - d) = \bar{r} + \bar{r} - \bar{r} = \bar{r}$.

Hence \hat{d}^1 is also a solution of (S_1) . Thus we have $\hat{d}^1 + \sum_{j=2}^l d^j = \bar{d} - d + \sum_{j=1}^l d^j = \bar{d}$. As a result of the definition of the set $I^0(\bar{y})$ there are index sets $I_j \in \mathcal{I}(\bar{y})$ with $j \notin I_j$ for all $j \in \{1, \dots, l\} = I(\bar{y}) \setminus I^0(\bar{y})$. So \hat{d}^1 is an element of the contingent cone $T_{I_1}(\bar{y})$ and d^j are elements of the contingent cones $T_{I_j}(\bar{y})$ for $j = 2, \dots, l$. Finally \bar{d} is the sum of a finite number of elements of $T(\bar{y})$ and therefore $T_R(\bar{y}) \subseteq \text{cone } T(\bar{y})$. \square

3.3. The nonlinear case

We consider now the inverse optimization problem

$$\min_{x,y,c} \{f(y) : Cc = \tilde{c}, Xx = \tilde{x}, y \in \Psi_N(x, c)\} \quad (16)$$

with a nonlinear lower level problem

$$\Psi_N(x, c) = \underset{y}{\text{Argmin}} \{c^\top y : g(x, y) \leq 0\}, \quad (17)$$

where $g : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^p$. Assume without loss of generality that the matrix X has full row rank. The assumption that the lower level problem has a linear objective function is of no loss of generality since this can be obtained by adding one variable bounding the objective function of a nonlinear programming problem from above and minimizing it.

Using the Karush-Kuhn-Tucker conditions for problem (17) in the upper level problem (16) we get

$$\begin{aligned} f(y) &\rightarrow \min_{x, y, u, c} \\ Cc &= \tilde{c}, Xx = \tilde{x}, \\ c + u^\top \nabla_y g(x, y) &= 0 \\ g(x, y) \leq 0, u &\geq 0, u^\top g(x, y) = 0. \end{aligned} \quad (18)$$

Denote by

$$M = \{(x, y) : \exists c \text{ with } Cc = \tilde{c}, Xx = \tilde{x}, y \in \Psi_N(x, c)\}$$

the feasible set of problem (16), where again, as in subsection (3.2), the lower level objective function coefficients are considered as rather abdicable. This means that we use the following optimality notion:

Definition 4 *A point $(\bar{x}, \bar{y}) \in M$ is a locally optimal solution of the problem (16) if there exists an $\varepsilon > 0$ such that for all $(x, y) \in M$ with $\|(x, y) - (\bar{x}, \bar{y})\| \leq \varepsilon$ we have $f(y) \geq f(\bar{y})$.*

Let

$$I(x, y) = \{i : g_i(x, y) = 0\},$$

$$\mathcal{I}(x, y) = \{I \subseteq I(x, y) : \exists c \text{ with } Cc = \tilde{c}, -c \in \text{cone} \{\nabla_y g_i(x, y) : i \in I\}\}$$

and

$$I^0(x, y) = \bigcap_{I \in \mathcal{I}(x, y)} I.$$

Similarly to subsection (3.2) it is relatively easy to determine the set $I^0(x, y)$. Then, problem (18) has an equivalent formulation with less variables locally around some feasible point (\bar{x}, \bar{y}) :

$$\begin{aligned} f(y) &\rightarrow \min_{x, y, I} \\ Xx &= \tilde{x}, \\ g_i(x, y) &= 0, i \in I \\ g_i(x, y) &\leq 0, i \in I(\bar{x}, \bar{y}) \setminus I \\ I &\in \mathcal{I}(\bar{x}, \bar{y}). \end{aligned} \quad (19)$$

The linearizing cone to the feasible set of (19) is

$$C_L(\bar{x}, \bar{y}) = \bigcup_{I \in \mathcal{I}(\bar{x}, \bar{y})} \left\{ (r, d)^\top : Xr = 0, \nabla g_i(\bar{x}, \bar{y})(r, d)^\top \begin{cases} = 0, & i \in I \\ \leq 0, & i \in I(\bar{x}, \bar{y}) \setminus I \end{cases} \right\}$$

This (in general non-convex) cone equals the contingent cone of M provided that the linear independence constraint qualification

(LICQ) The gradients $\nabla_y g_i(\bar{x}, \bar{y})$ with $i \in I(\bar{x}, \bar{y})$ are linearly independent.

is satisfied. This assumption can be weakened in demanding that the Mangasarian-Fromowitz constraint qualification is valid for each of the feasible sets of the problems in (19) for a fixed set $I \in \mathcal{I}(\bar{x}, \bar{y})$.

Now, consider the relaxed problem

$$\begin{aligned} f(y) &\rightarrow \min_{x,y} \\ Xx &= \tilde{x}, \\ g_i(x, y) &= 0, i \in I^0(\bar{x}, \bar{y}) \\ g_i(x, y) &\leq 0, i \in I(\bar{x}, \bar{y}) \setminus I^0(\bar{x}, \bar{y}). \end{aligned} \tag{20}$$

The linearizing cone to the feasible set of problem (20) is

$$C_N(\bar{x}, \bar{y}) = \left\{ (r, d)^\top : Xr = 0, \nabla g_i(\bar{x}, \bar{y})(r, d)^\top \begin{cases} = 0, & i \in I^0(\bar{x}, \bar{y}) \\ \leq 0 & i \in I(\bar{x}, \bar{y}) \setminus I^0(\bar{x}, \bar{y}) \end{cases} \right\}$$

and we have $C_L(\bar{x}, \bar{y}) \subseteq C_N(\bar{x}, \bar{y})$. The cone $C_N(\bar{x}, \bar{y})$ coincides with the contingent cone if the Mangasarian-Fromowitz constraint qualification is valid.

Theorem 5 *If the linear independence constraint qualification (LICQ) is satisfied for problem (20) then the cones $C_N(\bar{x}, \bar{y})$ and $C_L(\bar{x}, \bar{y})$ coincide.*

Proof: We clearly have by definition $C_L(\bar{x}, \bar{y}) \subseteq C_N(\bar{x}, \bar{y})$. Let $(\bar{r}, \bar{d}) \in C_N(\bar{x}, \bar{y})$. Take a covering of the index set $I(\bar{x}, \bar{y})$, i.e. index sets I_1, I_2, \dots, I_p with

$$I^0(\bar{x}, \bar{y}) \subseteq I_k, \quad I_k \in \mathcal{I}(\bar{x}, \bar{y}), \quad \bigcup_{k=1}^p I_k = I(\bar{x}, \bar{y}).$$

For all $k = 1, \dots, p$ compute a solution d^k of the following system of equations and inequalities:

$$\nabla g_i(\bar{x}, \bar{y})(\bar{r}, d)^\top \begin{cases} = 0, & i \in I^0(\bar{x}, \bar{y}) \\ = \nabla g_i(\bar{x}, \bar{y})(\bar{r}, \bar{d})^\top & i \in (I_k \cup I_1) \setminus I^0(\bar{x}, \bar{y}) \\ \leq 0 & i \in I(\bar{x}, \bar{y}) \setminus (I^0(\bar{x}, \bar{y}) \cup (I_k \cup I_1)) \end{cases}$$

By the linear independence constraint qualification, all these systems have feasible solutions.

By the definition, for all k , the point (\bar{r}, d^k) belongs to one of the convex subcones of $C_L(\bar{x}, \bar{y})$ and, hence, also to $C_L(\bar{x}, \bar{y})$ itself.

Consider the point

$$\hat{d} = \sum_{k=1}^p d^k.$$

Then we have

$$\nabla g_i(\bar{x}, \bar{y})(\bar{r}, \hat{d})^\top = \nabla g_i(\bar{x}, \bar{y})(\bar{r}, \bar{d})^\top, \quad \forall i \in I_1,$$

but $\hat{d} \neq \bar{d}$ is possible. But then,

$$0 = \nabla g_i(\bar{x}, \bar{y})(\bar{r} - \bar{r}, \hat{d} - \bar{d})^\top = \nabla_y g_i(\bar{x}, \bar{y})(\hat{d} - \bar{d})^\top, \quad \forall i \in I_1.$$

Now, set $d^1 := d^1 + \bar{d} - \hat{d}$ and let all the other vectors d^k unchanged. Then, d^1 remains in the respective subcone and $\bar{d} = \sum_k d^k$. This implies $(\bar{r}, \bar{d}) \in T_L(\bar{x}, \bar{y})$. \square

3.4. Optimality conditions using the KKT conditions

Using (5) as a reformulation of the bilevel problem (2) optimality conditions can also be obtained. The results in [5] indicate that each locally optimal solution of (2) is also locally optimal for (5) provided that the lower level problem is a convex, parametric one satisfying a regularity condition. Hence, we are interested in necessary optimality conditions for problem (5). Using Clarke's generalized gradient, optimality conditions having the following structure can be obtained [18]:

$$\begin{aligned}
\nabla_x F(\bar{x}, \bar{y}) + \mu^\top \nabla_{xy}^2 L(\bar{x}, \bar{y}, \bar{u}) + \nu^\top \nabla_x g(\bar{x}, \bar{y}) + \gamma^\top X &= 0 \\
\nabla_y F(\bar{x}, \bar{y}) + \mu^\top \nabla_{yy}^2 L(\bar{x}, \bar{y}, \bar{u}) + \nu^\top \nabla_y g(\bar{x}, \bar{y}) &= 0 \\
\mu^\top \nabla_y g(\bar{x}, \bar{y}) - \xi^\top &= 0 \\
\nu_i &= 0, \quad \text{if } g_i(\bar{x}, \bar{y}) < 0 \\
\xi_i &= 0, \quad \text{if } \bar{u}_i > 0 \\
(\nu_i, \xi_i) &\in \Xi_i, \quad \text{if } g_i(\bar{x}, \bar{y}) = \bar{u}_i = 0.
\end{aligned} \tag{21}$$

Then, the following results can be found in literature. For related results for problem (2), see [5].

1. The point $(\bar{x}, \bar{y}, \bar{u})$ is C-stationary for problem (5) if

$$\Xi_i = \{(a, b) : ab \geq 0\} \text{ for all } i.$$

A locally optimal solution of problem (5) is C-stationary if the Mangasarian-Fromowitz constraint qualification is satisfied for the problem

$$\begin{aligned}
F(x, y) &\rightarrow \min_{x, y, u} \\
\nabla_x L(x, y, u) &= 0 \\
g_i(x, y) &= 0, \text{ if } g_i(\bar{x}, \bar{y}) = 0 \\
u_i &= 0, \text{ if } \bar{u}_i = 0 \\
g_i(x, y) &\leq 0, \text{ if } g_i(\bar{x}, \bar{y}) < 0 \\
u_i &\geq 0, \text{ if } \bar{u}_i > 0 \\
x &\in X,
\end{aligned} \tag{22}$$

see [18].

2. The point $(\bar{x}, \bar{y}, \bar{u})$ is M-stationary for problem (5) if

$$\Xi_i = \{(a, b) : (a > 0 \text{ and } b > 0) \text{ or } ab = 0\} \text{ for all } i.$$

A locally optimal solution of problem (5) is M-stationary if the Abadie constraint qualification is satisfied, i.e., if the contingent cone to the feasible set of problem (5) equals its linearizing one

$$\begin{aligned}
C_M(\bar{x}, \bar{y}, \bar{u}) &= \{(r, d, w)^\top : Xr = 0, \nabla(\nabla_y L(\bar{x}, \bar{y}, \bar{u}))(r, d, w)^\top = 0, \\
&\nabla g_i(\bar{x}, \bar{y})(r, d)^\top = 0 \text{ if } \bar{u}_i > 0, w_i = 0 \text{ if } g_i(\bar{x}, \bar{y}) < 0, \\
&\min\{-\nabla g_i(\bar{x}, \bar{y})(r, d)^\top, w_i\} = 0, \text{ if } \bar{u}_i = g_i(\bar{x}, \bar{y}) = 0\},
\end{aligned} \tag{23}$$

see [9].

3. The point $(\bar{x}, \bar{y}, \bar{u})$ is strong stationary for problem (5) if

$$\Xi_i = \{(a, b) : a \geq 0 \text{ and } b \geq 0\} \text{ for all } i.$$

A locally optimal solution of problem (5) is strong stationary if the so-called strong Mangasarian-Fromowitz constraint qualification is satisfied for problem (22), see [16, 18].

3.5. Conditions using the optimal value function

Necessary optimality conditions being based on the reformulation in (6) can be found e.g. in the papers [1] and [19]. Let

$$f_d^-(x, v) := \liminf_{t \downarrow 0} t^{-1}[f(x + tv) - f(x)]$$

and

$$f_d^+(x, v) := \limsup_{t \downarrow 0} t^{-1}[f(x + tv) - f(x)]$$

denote the lower and upper Dini derivative of a function $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{\infty\}$. Then, the function F is said to have a convexicator $\partial^* f(x)$ at x if $\partial^* f(x) \subset \mathbb{R}^n$ is a closed set such that

$$f_d^-(x, v) \leq \sup_{x^* \in \partial^* f(x)} \langle x^*, v \rangle \text{ and } f_d^+(x, v) \geq \inf_{x^* \in \partial^* f(x)} \langle x^*, v \rangle$$

holds for all $v \in \mathbb{R}^n$. Convexicators are e.g. the Clarke's generalized derivative of locally Lipschitz functions or the coderivative of B. Mordukhovich [14]. Let for simplicity the constraints $Xx = \tilde{x}$ be missing.

Assume that the functions F, g_i, f, φ have upper semicontinuous convexicators. This assumption is e.g. satisfied if the functions f, g_i are locally Lipschitz continuous, a certain growth condition is satisfied, the set of abnormal multipliers

$$\{\lambda : 0 \in \sum_{i=1}^p \lambda_i \partial_y^o g_i(x, y), \lambda \geq 0, \lambda^\top g(x, y) = 0\}$$

reduces to zero [4] and Clarke's generalized derivative is used as a convexificator.

Then, consider the point-to-set mapping

$$H(x, y) := (g(x, y), f(x, y) - \varphi(x)) + \mathbb{R}_+^p \times \mathbb{R}.$$

Using this mapping, problem (6) can equivalently be formulated as

$$\min\{F(x, y) : 0 \in H(x, y)\}.$$

Moreover the distance function

$$(x, y) \mapsto d(0, H(x, y)) := \inf_z \{\|z\| : z \in H(x, y)\}$$

has also a bounded convexicator. This enables us to derive necessary optimality conditions for the bilevel programming problem (2).

Theorem 6 ([1]) *Let (\bar{x}, \bar{y}) be a locally optimal solution of the bilevel programming problem (2). Assume that there exists a neighborhood U of (\bar{x}, \bar{y}) such that the functions F, f, g are continuous on U and that they admit upper semicontinuous convexifiers which are bounded at (\bar{x}, \bar{y}) . Then, there exist scalars $\lambda_1, \lambda_2, \gamma \geq 0$ and a vector $\mu \geq 0$ such that*

$$\begin{aligned} \|(\mu, \gamma)\| = 1, \lambda_1 + \lambda_2 = 1, \mu^\top g(\bar{x}, \bar{y}) = 0, \\ (0, 0) \in \lambda_1 \partial^* F(\bar{x}, \bar{y}) + \lambda_2 \gamma \partial^* f(\bar{x}, \bar{y}) + \lambda_2 \sum_{i=1}^p \mu_i \partial^* g_i(\bar{x}, \bar{y}) - \lambda_2 \gamma (\partial^* \varphi(\bar{x}) \times \{0\}). \end{aligned}$$

The following regularity notion is needed for problem (2): Problem (2) with $X = \mathbb{R}^n$ is called regular at the point (\bar{x}, \bar{y}) if there exist a neighborhood U of (\bar{x}, \bar{y}) and numbers $\delta, \beta > 0$ such that:

$$\begin{aligned} \forall \mu \in \mathbb{R}_+^p, \forall (x, y) \in U, \forall x_g^* \in \partial^* g(x, y), \forall x_f^* \in \partial^* f(x, y), \forall x_v^* \in \partial^* V(x) \times \{0\} \\ \exists \xi, \|\xi\| \leq \delta \end{aligned}$$

such that

$$\mu g(x, y) + \langle (x_g^*, x_f^* - x_v^*), \xi \rangle \geq \beta$$

Theorem 7 ([1]) *Let the assumptions of Theorem 6 be satisfied and assume that problem (2) is regular. Then, one has $\lambda_1 > 0$.*

References

- [1] H. Babahadda and N. Gadhi. Necessary optimality conditions for bilevel optimization problems using convexifiers. Technical report, Department of Mathematics, Dhar El Mehrez, Sidi Mohamed Ben Abdellah University, Fes, Morocco, 2005.
- [2] B. Bank, J. Guddat, D. Klatte, B. Kummer, and K. Tammer. *Non-Linear Parametric Optimization*. Akademie-Verlag, Berlin, 1982.
- [3] J.F. Bard. *Practical Bilevel Optimization: Algorithms and Applications*. Kluwer Academic Publishers, Dordrecht, 1998.
- [4] F.H. Clarke. *Optimization and Nonsmooth Analysis*. John Wiley & Sons, New York, 1983.
- [5] S. Dempe. *Foundations of Bilevel Programming*. Kluwer Academic Publishers, Dordrecht, 2002.
- [6] S. Dempe. Annotated bibliography on bilevel programming and mathematical programs with equilibrium constraints. *Optimization*, 52:333–359, 2003.
- [7] S. Dempe and S. Lohse. Inverse linear programming. Technical report, Technical University Bergakademie Freiberg, Department of Mathematics and Computer Science, 2004.
- [8] X. Deng. Complexity issues in bilevel linear programming. In A. Migdalas, P.M. Pardalos, and P. Värbrand, editors, *Multilevel Optimization: Algorithms and Applications*, pages 149–164. Kluwer Academic Publishers, Dordrecht, 1998.

- [9] M. L. Flegel and C. Kanzow. On M-stationary points for mathematical programs with equilibrium constraints. Technical report, University of Würzburg, Institute of Applied Mathematics and Statistics, Germany, 2005.
- [10] P.T. Harker and J.-S. Pang. Existence of optimal solutions to mathematical programs with equilibrium constraints. *Operations Research Letters*, 7:61–64, 1988.
- [11] M. Kojima. Strongly stable stationary solutions in nonlinear programs. In S.M. Robinson, editor, *Analysis and Computation of Fixed Points*, pages 93–138. Academic Press, New York, 1980.
- [12] Z.-Q. Luo, J.-S. Pang, and D. Ralph. *Mathematical Programs with Equilibrium Constraints*. Cambridge University Press, Cambridge, 1996.
- [13] J. A. Mirrlees. The theory of moral hazard and unobservable behaviour: part I. *Review of Economic Studies*, 66:3–21, 1999.
- [14] B.S. Mordukhovich. Generalized differential calculus for nonsmooth and set-valued mappings. 183:250–288, 1994.
- [15] J. Outrata, M. Kočvara, and J. Zowe. *Nonsmooth Approach to Optimization Problems with Equilibrium Constraints*. Kluwer Academic Publishers, Dordrecht, 1998.
- [16] J.-S. Pang and M. Fukushima. Complementarity constraint qualifications and simplified B-stationarity conditions for mathematical programs with equilibrium constraints. 13:111–136, 1999.
- [17] D. Ralph and S. Dempe. Directional derivatives of the solution of a parametric nonlinear program. *Mathematical Programming*, 70:159–172, 1995.
- [18] H. Scheel and S. Scholtes. Mathematical programs with equilibrium constraints: stationarity, optimality, and sensitivity. *Mathematics of Operations Research*, 25:1–22, 2000.
- [19] J.J. Ye and D.L. Zhu. Optimality conditions for bilevel programming problems. *Optimization*, 33:9–27, 1995. with correction in *Optimization* 39(1997), pp. 361-366.

ЛОКАЛЬНЫЙ ПОИСК В КОМБИНАТОРНОЙ ОПТИМИЗАЦИИ. НУЖНА ЛИ ПРОИЗВОДНАЯ? ¹

Ю.А.Кочетов, А.В.Плясунов

*Институт математики им. С. Л. Соболева СО РАН, Новосибирск
e-mail: jkochet@math.nsc.ru, apljas@math.nsc.ru*

Аннотация. Показано, что локальные оптимумы в задачах комбинаторной оптимизации тесно связаны со стационарными точками в соответствующих задачах с непрерывными переменными. Методы локального поиска неявно используют необходимые условия Каруша–Куна–Таккера, стремясь выбрать среди стационарных точек наилучшую. Такой взгляд позволяет лучше понять причину высокой эффективности этих методов и дает направление дальнейших исследований.

Ключевые слова: NP-трудные задачи, локальный поиск, условия Каруша–Куна–Таккера, волновое уравнение, ландшафты.

Введение

Методы локального поиска являются одним из наиболее наглядных и естественных подходов к решению задач комбинаторной оптимизации. Высокая эффективность при решении NP-трудных задач и гибкость в адаптации к сложным математическим моделям открыли для них широкую дорогу к практическому использованию. Несмотря на различие концепций, эти методы часто используют процедуру поиска локальных оптимумов. Попытка сконцентрировать внимание на локальных оптимумах близка в идейном смысле к выделению стационарных точек и использованию вариаций в непрерывной оптимизации. Тем не менее необходимые условия Каруша–Куна–Таккера (ККТ), подразумевающие взятие производных, в методах комбинаторной оптимизации не используются. Создается впечатление, что классические методы непрерывной оптимизации слишком далеки от комбинаторных. Тем не менее в ряде случаев условие локальной оптимальности дискретного решения эквивалентно условиям ККТ для непрерывной задачи, полученной релаксацией требования целочисленности переменных. Для NP-трудных задач проблема часто состоит не в том, чтобы найти локальные оптимумы, а в том, что таких точек оказывается слишком много. Найти среди них точку глобального оптимума представляется серьезной проблемой, что, по-видимому, и делает исходную задачу NP-трудной. Понятие глобального оптимума не является конструктивным. Даже получив такое решение, доказать его оптимальность очень трудно. Аналогичные проблемы возникают и в непрерывной оптимизации. Там необходимые условия также имеют локальный характер, так как они опираются на понятие вариации. В комбинаторной оптимизации реализуется та же идея, и понятия окрестности, соседних решений — аналог вариации играют здесь центральную роль.

Другим важным аспектом является трудоемкость получения локального оптимума. На практике стандартный алгоритм локального спуска сходится очень быстро, за линейное или квадратичное число итераций [14]. Вместе с тем известен широкий класс задач локального поиска, класс *PLS*, в котором по аналогии с NP-полными задачами удалось

¹Работа выполнена при поддержке РФФИ (проект 03-01-00455)

выделить класс наиболее трудных задач. Они названы *PLS*-полными [17]. Существование полиномиального алгоритма нахождения локального оптимума для любой из них, влечет существование полиномиального алгоритма для всех задач из класса *PLS*. Возвращаясь к условиям ККТ, это означает, что разработка алгоритмов поиска целочисленных стационарных точек является важным направлением в дискретной оптимизации.

Интересным является также вопрос об уклонении локальных оптимумов. Известно, что в худшем случае относительная погрешность локального оптимума может оказаться сколь угодно большой. Однако, если рассматривать среднее значение целевой функции на множестве всех допустимых решений, то во многих задачах любой локальный оптимум оказывается не хуже этого среднего значения. Более того, стандартный алгоритм локального спуска достигает среднего значения за полиномиальное число итераций. Достаточным условием для выполнения этого свойства является справедливость разностного соотношения, форма записи которого для стоимостной функции ψ совпадает с дифференциальным уравнением колебания мембраны ([4], глава 8):

$$\Delta\psi + \frac{K\psi}{n} = 0, \quad K > 0, \quad n - \text{размерность задачи.}$$

Оператор Δ задает среднюю разность между соседними решениями для заданной окрестности. Если каждое допустимое решение имеет m соседей и i -й сосед отличается по целевой функции на δ_i от заданного решения, то $\Delta\psi = (\sum_{i=1}^m \delta_i)/m$. Данное определение приводит к разностному аналогу оператора Лапласа в непрерывном случае, что позволяет использовать тот же символ для дискретных задач. Действительно, рассмотрим оператор Лапласа для одномерного $\Delta = \frac{\partial^2}{\partial x^2}$ и двухмерного $\Delta = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$ случаев. Соответствующие разностные соотношения имеют следующий вид [3]

$$\frac{(f(x+h) - f(x)) + (f(x-h) - f(x))}{h^2}$$

для одномерного случая и

$$\frac{(f(x+h, y) - f(x, y)) + (f(x-h, y) - f(x, y))}{h^2} + \frac{(f(x, y+h) - f(x, y)) + (f(x, y-h) - f(x, y))}{h^2}$$

для двухмерного случая. В обоих случаях Лапласиан пропорционален среднему значению от приращения функции при переходе от данной точки к соседней. Окрестность точки в данном случае совпадает с множеством ближайших точек разностной сетки.

Справедливость разностного аналога волнового уравнения указывает на глубинные связи дискретной оптимизации с другими разделами математики, влечет ряд важных свойств локального поиска, дает толчок для исследования ландшафтов [5] и использования методов спектральной теории графов [4].

1. Локальные оптимумы и условия ККТ

Рассмотрим задачу минимизации псевдобулева полинома

$$P(z) = \sum_{j \in J} a_j \prod_{i \in \alpha_j} z_i$$

от переменных $z_i \in \{0, 1\}$, $i \in I = \{1, \dots, n\}$, $\alpha_j \subset I$, $j \in J = \{1, \dots, m\}$. Для любой точки $z = (z_1, \dots, z_n)$ булева куба B^n под окрестностью $Flip(z)$ будем понимать множество

точек, отличающихся от z только в одной координате. Точка z называется локальным минимумом, если в ее окрестности нет точек с меньшим значением целевой функции. Задача минимизации псевдодобулевых полиномов на множестве $E^n \setminus \{0, \dots, 0\}$ тесно связана с дискретными задачами размещения, задачей о выполнимости, о разбиении графа и др.

Заменим условие булевости переменных на их принадлежность отрезку $[0, 1]$. Тогда функция Лагранжа с множителями $\mu_i, \sigma_i, i \in I$ и непрерывными переменными $0 \leq z_i \leq 1, i \in I$ имеет вид:

$$L(z, \mu, \sigma) = P(z) + \sum_{i \in I} \sigma_i (z_i - 1) + \sum_{i \in I} \mu_i z_i.$$

Условия ККТ записываются следующим образом:

$$\frac{\partial L}{\partial z_i} = 0, \quad z_i \mu_i = 0, \quad (1 - z_i) \sigma_i = 0, \quad i \in I,$$

$$\sigma_i \geq 0, \quad \mu_i \geq 0, \quad 0 \leq z_i \leq 1, \quad i \in I.$$

Теорема 1. *Булев вектор $z = (z_1, \dots, z_n)$ удовлетворяет условиям ККТ если и только если z является локальным минимумом функции $P(z)$ для окрестности $Flip$.*

Аналогичным свойством обладают локальные минимумы в задаче о раскраске графов [15], в задаче о p -медиане и др. Заметим, что если некоторая окрестность включает в себя окрестность $Flip$ в качестве подмножества, то для ее локальных оптимумов тоже будут выполняться условия ККТ. Отсюда следует, что если, например, в генетических алгоритмах [10] применяется процедура локального спуска к каждому элементу популяции, то в процессе эволюции участвуют только стационарные точки. Таким образом, алгоритм неявно использует необходимые условия экстремума, а расширение окрестности приводит к дополнительному отсеву среди стационарных точек.

Заметим, что сведение, например, простейшей задачи размещения [1] к задаче минимизации псевдодобулевого полинома приводит к появлению новых свойств, которых не было в исходной постановке. Легко проверить, что среди оптимальных решений задачи минимизации полинома с непрерывными переменными $0 \leq z_i \leq 1, i \in I$ всегда найдется целочисленное оптимальное решение. Другими словами, разрыв двойственности (целочисленности) всегда равен нулю, что неверно для исходной задачи размещения. Таким образом, полиномиальное сведение одной задачи к другой, эквивалентное с точки зрения поиска оптимального решения, может существенно менять свойства оптимизационной задачи. В частности, в работе [2] приводится сведение квадратичной задачи о назначениях к задаче максимизации выпуклой функции на выпуклом многограннике. В итоге каждая целочисленная допустимая точка исходной задачи оказывается стационарной точкой для непрерывной релаксации.

2. Локальная структура NP-трудных задач

Задача о разбиении графа формулируется следующим образом. Задан взвешенный граф $G = (V, E)$ с четным числом вершин, $|V| = n$. Каждому ребру $e \in E$ приписан вес w_e . Требуется разбить множество V на две равные части $V_1 \cup V_2 = V$ так, чтобы суммарный вес ребер, соединяющих V_1 и V_2 , был бы минимальным.

Для допустимого решения (V_1, V_2) определим окрестность $Swap$ как множество всех разбиений множества V на две равные части, которые получаются из разбиения (V_1, V_2)

заменой одной вершины из V_1 на одну вершину из V_2 . Мощность такой окрестности равна $n^2/4$. Среднее значение целевой функции C_{AV} на всем множестве допустимых решений равно $C_{AV} = \frac{n}{4(n-1)} \sum_{e \in E} w_e$. Наряду с исходной целевой функцией $C(V_1, V_2)$ рассмотрим нормированную целевую функцию $\bar{C}(V_1, V_2) = C(V_1, V_2) - C_{AV}$.

Теорема 2 [6]. *Для нормированной целевой функции $\bar{C}(V_1, V_2)$ задачи о разбиении графа справедливо равенство*

$$\Delta \bar{C} = -\frac{8(n-1)}{n^2} \bar{C}.$$

Так как $\Delta C = \Delta \bar{C}$, то в окрестности любого допустимого решения среднее изменение целевой функции $C(V_1, V_2)$ равно $-8(n-1)\bar{C}/n^2$, то есть зависит только от разбиения (V_1, V_2) и не зависит от элементов окрестности. Наличие такого свойства кажется экзотикой. По сути это дискретный аналог волнового уравнения математической физики и его справедливость для оптимизационных задач на графах кажется невероятным! Тем не менее аналогичные утверждения справедливы и для ряда других NP-трудных задач, в частности, для задачи коммивояжера, раскраски графов и некоторых других [8]. Из уравнения непосредственно вытекает следующие утверждения.

Следствие 1. *Если разбиение (V_1, V_2) является локальным минимумом, то $C(V_1, V_2) \leq C_{AV}$.*

Следствие 2. *Алгоритм локального спуска достигает значение C_{AV} с произвольного разбиения (V_1, V_2) за $O(nr)$ итераций, если $\max_{(V_1, V_2)} C(V_1, V_2) \leq 2^r C_{AV}$.*

Другими словами, если величина r полиномиально ограничена, то алгоритм локального спуска достигает среднего значения за полиномиальное число шагов и при достижении локального оптимума нельзя получить слишком плохое значение функционала. Конечно, отсюда еще не следует, что и локальный оптимум может быть получен за полиномиальное число шагов, хотя в среднем, как и для симплекс-метода, который по сути также является алгоритмом локального спуска, удается получать такие оценки при необременительных ограничениях на исходные данные [14].

Задача о коммивояжере является одной из наиболее изученных задач комбинаторной оптимизации [10]. Дано n городов и матрица расстояний (r_{ij}) между ними. Требуется найти замкнутый путь минимальной длины, проходящий через каждый город ровно один раз. Если такой путь представить в виде последовательности городов, то окрестность $2 - exchange$ определяется как множество всех последовательностей, получающихся из данной перестановкой двух городов. Такая окрестность содержит $n(n-1)/2$ элементов. Пусть матрица (r_{ij}) является симметричной. Тогда среднее значение R_{AB} целевой функции R на всем множестве допустимых решений определяется равенством $R_{AB} = \sum_{i \neq j} r_{ij} / (n-1)$.

Теорема 3 [6]. *Для нормированной целевой функции $\bar{R} = R - R_{AB}$ задачи коммивояжера с симметричной матрицей расстояний справедливо равенство*

$$\Delta \bar{R} = -\frac{4}{n} \bar{R}.$$

Как и прежде, оператор Лапласа задает среднее значение разности между допустимым решением и его соседями. Это значение зависит от самого решения и размерности задачи и не зависит от соседних решений.

Приведем еще один пример оптимизационной задачи, для которой выполняется аналогичное соотношение. Она называется задачей о разбиении. Дано n предметов. У каждого предмета свой вес $w_i, i = 1, \dots, n$. Можно ли разбить множество предметов на два подмножества так, чтобы суммарные веса подмножеств совпадали?

Введем переменные $x_i \in \{-1, 1\}, i = 1, \dots, n$, знак которых указывает на принадлежность предмета к одному из подмножеств. Тогда величина $\sum_{i=1}^n w_i x_i$ определяет разность весов двух подмножеств. Если она равна нулю, то требуемое разбиение найдено. Для произвольного разбиения окрестностью будем называть все разбиения, получающиеся из данного переносом одного предмета в другое подмножество. Для переменных $x_i, i = 1, \dots, n$ это означает изменение знака ровно в одной координате. Рассмотрим целевую функцию $U = (\sum_{i=1}^n w_i x_i)^2$. Ее среднее значение U_{AB} на множестве всех разбиений определяется равенством $U_{AB} = \sum_{i=1}^n w_i^2$.

Теорема 4 [6]. *Для нормированной целевой функции $\bar{U} = U - U_{AB}$ справедливо равенство*

$$\Delta \bar{U} = -\frac{4}{n} \bar{U}.$$

Следствия 1 и 2 могут быть переформулированы для задач коммивояжера и разбиения. Они вытекают из полученных соотношений, которые оказываются тесно связанными с понятием ландшафта.

3. Ландшафты и их свойства

Рассмотрим задачу минимизации целевой функции C определенной на конечном множестве допустимых решений \mathfrak{S} . Для любого $x \in \mathfrak{S}$ через $N(x) \subseteq \mathfrak{S}$ обозначим окрестность решения x в \mathfrak{S} . Графом соседства для пары (\mathfrak{S}, N) назовем ориентированный граф $G = (V, E)$ с множеством вершин $V = \mathfrak{S}$ и множеством дуг вида (x, y) , где $y \in N(x)$. Если окрестность является симметричной, т. е. $y \in N(x) \Leftrightarrow x \in N(y)$, то вместо дуг используются ребра и граф G из ориентированного становится неориентированным. В этом разделе будут рассматриваться только симметричные окрестности, мощность которых не зависит от x .

Граф G , каждой вершине x которого приписан вес $C(x)$, будем называть ландшафтом для задачи локального поиска (\mathfrak{S}, C, N) . Понятие ландшафта удобно для описания работы алгоритмов локального поиска. Каждому шагу соответствует ребро в графе. Если граф связный, то из любого решения можно добраться до оптимального, двигаясь только в соседние решения. Это свойство является важным для методов локального поиска. Другим важным свойством ландшафтов является его *пересеченность* или *изрезанность* (ruggedness). Сильно изрезанные ландшафты имеют много локальных оптимумов и представляют серьезную трудность для методов локального поиска. Для измерения изрезанности ландшафтов используют функцию автокорреляции $\rho(d)$, где d - расстояние по числу ребер в графе G между вершинами. По определению [5]

$$\rho(d) = 1 - \frac{\langle (C(x) - C(y))^2 \rangle_{d(x,y)=d}}{\langle (C(x) - C(y))^2 \rangle},$$

где $\langle (C(x) - C(y))^2 \rangle$ обозначает среднее значение величины $(C(x) - C(y))^2$ по всем парам (x, y) из \mathfrak{X} , а $\langle (C(x) - C(y))^2 \rangle_{d(x,y)=d}$ есть среднее значение для пар (x, y) , находящихся в G на расстоянии d . Величина $\rho(d)$ показывает уровень корреляции между решениями, находящимися на расстоянии d друг от друга. Наиболее важным случаем является $d = 1$, так как он соответствует корреляции для соседних решений. Если $\rho(1)$ близко к 1, то соседние решения мало отличаются друг от друга по целевой функции. Это мало изрезанные ландшафты, удобные для локального поиска. Если же $\rho(1)$ близко к 0, то корреляции между соседними решениями почти нет, ландшафт сильно изрезан и, эффективность методов локального поиска скорее всего будет низкой.

Для вычисления $\rho(1)$ иногда удобно использовать другую функцию $r(s)$, $s \geq 1$, целое. Рассмотрим последовательность значений $C(x_i)$ для случайного блуждания $\{x_i\}$ по вершинам графа G . На шаге i с равной вероятностью переходим из вершины x_i в одну из смежных вершин x_{i+1} . По определению

$$r(s) = 1 - \frac{\langle (C(x_i) - C(x_{i+s}))^2 \rangle}{2(\langle C^2 \rangle - \langle C \rangle^2)}, \quad \langle C \rangle = C_{AV}.$$

При $s = 1$ получаем $r(1) = \rho(1)$. Для вычисления $r(s)$ получена формула, использующая разложение функции $C(x)$ через собственные вектора $\phi_i, i = 0, 1, \dots, |V| - 1$ матрицы Лапласа $L = D - A$, где D - диагональная матрица размера $|V| \times |V|$ со степенями вершин по диагонали, A - матрица смежности.

Теорема 5 [12]. Пусть $C(x) = \sum_{i=0}^{|V|-1} a_i \phi_i(x)$ - разложение функции $C(x)$, λ_i - собственные числа для $\phi_i(x)$ и $|N|$ - мощность окрестности. Тогда

$$r(s) = \sum_{i>0} \frac{a_i^2}{\sum_{j>0} a_j^2} \left(1 - \frac{\lambda_i}{|N|}\right)^s.$$

Ландшафт называют элементарным [12], если $\lambda_i = \lambda$ для всех $i > 0$. Если для некоторой задачи локального поиска справедливо волновое уравнение, то соответствующий ландшафт является элементарным [12]. В таблице 1 приводятся характеристики ландшафтов для следующих задач: задача коммивояжера (TSP), квадратичная задача о назначениях (QAP), задача о вершинном покрытии (NC), задача о разбиении графа (GBP), задача о максимальном разрезе (MC), задача о раскраске вершин графа в s цветов, задача о независимом множестве (WIS) со штрафной функцией $C(x) = \sum_i x_i w_i - (\alpha/2) \sum_{ij} x_i x_j a_{ij}$, $A = \{a_{ij}\}$ - матрица смежности. Для задачи коммивояжера с симметричной матрицей расстояний окрестность $2 - opt$ содержит $n(n-3)/2$ туров. Каждый из них получается удалением из данного тура двух несмежных ребер и добавлением двух других так, чтобы снова получился тур. Для определения окрестности $2 - exchange$ тур представляется в виде последовательности вершин графа. Соседние решения порождаются перестановкой двух вершин в этой последовательности. Аналогично определяется окрестность $3 - exchange$. Из таблицы 1 следует, что коэффициент автокорреляции $\xi = n/2$ для окрестности $2 - opt$ и $\xi = n/4$ для окрестности $2 - exchange$, т.е. окрестность $2 - opt$ порождает менее изрезанный ландшафт и должна приводить к лучшим результатам, что подтверждается экспериментальными исследованиями [9].

Для задачи о разбиении графа окрестность $Swap$ сохраняет равенство $|V_1| = |V_2|$. Поиск ведется только в допустимой области, $\xi = n/8$. Окрестность $\alpha - Flip$ разрешает нарушать баланс, но за нарушение вводится штраф в целевую функцию:

Задача	N	$ N $	$ \mathfrak{S} $	$\xi = 1/(1 - \rho(1))$	Элементарный
TSP	2-opt	$n(n-3)/2$	$(n-1)!/2$	$n/2$	да
QAP	2-exchange	$n(n-1)/2$	$n!$	$\geq n/4$	нет
TSP	2-exchange	$n(n-1)/2$	$(n-1)!/2$	$n/4$	да
TSP	2,3-exchange	$n^3 - n$	$(n-1)!/2$	$(n+1)/6$	да
GC(c)	c-Flip	$n(c-1)$	c^n	$(c-1)n/2c$	да
WIS	α -Flip	n	2^n	$\geq n/4$	нет
NC	α -Flip	n	2^n	$\geq n/4$	нет
GBP	α -Flip	n	$C_n^{n/2}$	$n/4$	да
MC	Flip	n	2^n	$n/4$	да
GBP	Swap	$n^2/4$	$C_n^{n/2}$	$n/8$	да

Таблица 1: Характеристики ландшафтов [5]

$C(x) = \sum_{ij} w_{ij} x_i (1 - x_j) + \alpha (\sum_i 2x_i - n)^2$, $w_{ij} \in R$ — вес ребра (ij) . Область поиска расширяется, коэффициент автокорреляции возрастает, $\xi = n/4$. Расширение области приводит к более пологим ландшафтам, что должно положительно сказываться на результатах локального поиска. Экспериментальные исследования подтверждают это предположение [11].

4. Теория элементарных ландшафтов

Основываясь на алгебраических инвариантах графа соседства, удается получать характеристики локальных экстремумов оптимизационной задачи и поведения алгоритмов локального улучшения. В этом разделе предлагается новое определение Лапласиана графа [6], которое позволяет обобщить понятие элементарных ландшафтов на несимметричные и нерегулярные графы соседства. Обозначим через \tilde{L} нормированную матрицу Лапласа

$$\tilde{L} = D^{-1}L = D^{-1}(D - A) = I - D^{-1}A.$$

Для любой целевой функции C имеет место равенство

$$D^{-1}AC(x) = \frac{\sum_{y \in N(x)} C(y)}{|N(x)|},$$

т.е. величина $\tilde{C}(x) = D^{-1}AC(x)$ является средним значением функции C на множестве соседних решений $N(x)$.

Ландшафт назовем элементарным, если для подходящей константы α функция $C_\alpha = C - \alpha$ является собственным вектором матрицы \tilde{L} , т.е. $\tilde{L}C_\alpha = \lambda C_\alpha$ для некоторого собственного числа λ . Здесь и далее будем отождествлять целевую функцию с вектором ее значений.

Из определения нормированной матрицы Лапласа и разностного оператора Δ следует, что

$$\tilde{L}C_\alpha(x) = -\frac{\sum_{y \in N(x)} (C_\alpha(y) - C_\alpha(x))}{|N(x)|} = -\Delta C_\alpha(x),$$

т.е. $\tilde{L} = -\Delta$. Таким образом ландшафт является элементарным тогда и только тогда,

когда для некоторого α функция C_α удовлетворяет следующему соотношению

$$\Delta C_\alpha + \lambda C_\alpha = 0.$$

Из уравнения получаем $\tilde{C}_\alpha(x) = (1 - \lambda)C_\alpha(x)$. Если собственное значение λ является положительным и x^* — локальный минимум, то $C_\alpha(x^*) \leq \tilde{C}_\alpha(x^*) = (1 - \lambda)C_\alpha(x^*)$ и, следовательно, $C(x^*) \leq \alpha$. Аналогично для локальных максимумов получаем $C(x^*) \geq \alpha$.

Для ландшафтов с регулярными и симметричными графами соседства величина сдвига α совпадает со средним значением C_{AV} целевой функции на множестве всех решений \mathfrak{S} [8]. Это утверждение можно расширить на более широкий класс ландшафтов.

Теорема 6 [7] *Если ландшафт является элементарным и матрица $D^{-1}A$ является дважды стохастической, то величина сдвига α совпадает со значением C_{AV} .*

Для многих задач локального поиска граф соседства совпадает с гиперкубом, например, для окрестности *Flip* в задачах размещения, всех вариантах задачи о выполнимости, задачах поиска оптимального разреза и других [17]. Поэтому для выявления элементарных ландшафтов необходима информация о спектре гиперкуба.

Обозначим через B^n множество всех булевых векторов длины n . Весом булева вектора x называется число его ненулевых координат. Под гиперкубом будем понимать граф с множеством вершин B^n . Две вершины x, y соединены ребром тогда и только тогда, когда расстояние Хэмминга между ними равно единице. Из определения следует, что гиперкуб является связным двудольным n -регулярным графом диаметра n и порядка 2^n .

Каждому $z \in B^n$ сопоставим функцию Уолша $W_z : B^n \rightarrow \{-1, 1\}$, которая вычисляется по формуле

$$W_z(x) = (-1)^{\sum_{i=1}^n x_i z_i}, x \in B^n.$$

Под весом функции W_z будем понимать вес вектора z . Нормированные функции Уолша $\{\frac{1}{2^{n/2}}W_z : z \in B^n\}$ образуют ортонормированный базис пространства R^{2^n} [13]. Пусть L — матрица Лапласа гиперкуба.

Теорема 7 [16]. *Спектр матрицы L состоит из $n + 1$ числа $\lambda_k = 2k, k = 0, \dots, n$. Кратность собственного числа λ_k равна числу сочетаний из n по k . Функции Уолша веса k образуют ортонормированный базис собственного пространства для числа λ_k .*

Учитывая, что собственные числа λ_L и $\lambda_{\tilde{L}}$ матриц L и \tilde{L} для гиперкуба связаны соотношением $\lambda_{\tilde{L}} = \lambda_L/n$ в дальнейшем можно ограничиться исследованием спектра матрицы Лапласа L . Так как гиперкуб — регулярный и симметричный граф, то для любой целевой функции C , для которой ландшафт является элементарным, параметр нормализации α совпадает со средним значением C_{AV} целевой функции на множестве всех решений [8].

В задаче о разбиении, сформулированной в разделе 2, граф соседства совпадает с гиперкубом. Из теорем 4 и 7 следует, что нормированная целевая функция задачи принадлежит собственному пространству натянутому на функции Уолша веса 2. Аналогичный результат имеет место и для следующего варианта задачи о 3-выполнимости (NAE-3SAT) [8]. Задано n булевых переменных. Литералом назовем переменную, либо ее отрицание. Вход задачи задается конечным набором дизъюнкций, каждая из которых состоит ровно из трех литералов. Дизъюнкция считается истинной, если в ней найдется по крайней мере один истинный и один ложный литерал. Значение целевой функции для любого

набора истинностных значений переменных равно числу выполненных дизъюнкций. Окрестность произвольного решения состоит из всех булевых векторов, находящихся от него на расстоянии 1.

Теорема 8 *Нормированная целевая функция задачи NAE-3SAT принадлежит собственному пространству натянутому на функции Уолша веса 2.*

5. Заключение

Основным критерием при отборе представленных результатов был вопрос насколько они полезны при анализе возможностей локального поиска в задачах комбинаторной оптимизации. Мы опустили упоминание о таких хорошо известных методах как метод эллипсоидов и метод внутренних точек, которые доказали свою высокую эффективность, но не относятся к методам локального поиска. В той же мере это касается полуопределенного программирования и некоторых других подходов, которые связывают непрерывную и комбинаторную оптимизацию. Конечно, с этой точки зрения полученные связи с условиями ККТ недостаточно конструктивны. Они пока не дали каких-то зримых прорывов в анализе возможностей локального поиска. Эти условия больше говорят о глубокой внутренней связи дискретной оптимизации с непрерывной оптимизацией, когда-то отталкнувшейся от идеи вариации и приведшей к созданию мощного арсенала для анализа экстремальных задач. Что же касается спектрального подхода, то в последние годы идет активное развитие спектральной теории ландшафтов комбинаторных задач. Во всяком случае уже трудно себе представить изучение методов локального поиска без этих исследований [5, 6, 7, 12, 13, 16], начатых пионерской работой Гровера [8].

Исследование ландшафтов является важным направлением дальнейших исследований. Было бы интересно получить значение коэффициента автокорреляции для дискретных задач размещения, задачи о покрытии, задачи о p -медиане с различными по мощности окрестностями и др. Исследование спектральных свойств графов соседства поможет лучше понять возможности локальных методов и предсказать их поведение при решении практических задач.

Список литературы

- [1] Береснев В.Л., Гимади Э.Х., Дементьев В.Т. Экстремальные задачи стандартизации. – Новосибирск: Наука, 1978.
- [2] Васильев И.Л. *Об опыте решения задачи о назначениях.* - Оптимизация, Управление, Интеллект, 1999. N 3, с. 133-151.
- [3] Годунов С.К., Рябенский В.С. *Разностные схемы.* - Москва: Наука, 1977.
- [4] Цветкович Д., Дуб М., Захс Х. *Спектры графов. Теория и применение.* - Киев: Наукова думка, 1984.
- [5] Angel E., Zissimopoulos V. *On the classification of NP-complete problems in terms of their correlation coefficient.* - Discrete Applied Mathematics, 2000, v. 99, p. 261–277.

- [6] Barnes J.W., Dimova B., Dokov S.P., Solomon A. *The theory of elementary landscapes*. - Applied Mathematics Letters, 2003, v. 16, p. 261–277.
- [7] Barnes J.W., Dokov S.P., Solomon A. *Extending elementary landscape characterizations for COPs to arbitrary neighborhood definitions*. Air Force Office of Scientific Research Principal Investigators Meeting on Optimization and Discrete Mathematics at the Minnowbrook Conference Center, Blue Mountain Lake, New York, November 11-13, 2001.
- [8] Grover L. K. *Local search and the local structure of NP-complete problems*. - Operations Research Letters, 1992, v.12, N4, p. 235–244.
- [9] Johnson D. S. *Local optimization and the traveling salesman problem*. - ICALP 90 Automata, Languages and Programming, 1990, p. 446–461.
- [10] Johnson D. S., McGeoch L. A. *The traveling salesman problem: a case study*. in: E. Aarts and J. K. Lenstra (eds.) *Local Search in Combinatorial Optimization*, Chichester: John Wiley & Sons, 1997, p. 215–310.
- [11] Johnson D. S., Aragon C. R., McGeoch L. A., Schevon C. *Optimization by simulated annealing: an experimental evaluation. Part 1, Graph Partitioning*. - Operations Research, 1989, v.37, N6, p. 865–892.
- [12] Stadler P. F. *Landscapes and their correlation functions*. Technical report 95-07-067, Santa Fe Institute, Santa Fe, NM, 1995.
- [13] Stadler P. F. *Fitness landscapes*. in: Lassig M. and Valleriani A. (eds.) *Biological Evolution and Statistical Physics*. Berlin: Springer-Verlag, 2002, p. 187–207.
- [14] Tovey C. *Local improvement on discrete structures*. in: E. Aarts and J. K. Lenstra (eds.) *Local Search in Combinatorial Optimization*, Chichester: John Wiley & Sons, 1997, p. 57–90.
- [15] Vredeveld T., Lenstra J. K. *On local search for the generalized graph coloring problem*. Operations Research Letters, 2003, v. 31, p. 28–34.
- [16] Weinberger E.D. *Fourier and Taylor series on fitness landscapes*. Biological cybernetics, 1991, v. 65, p. 321–330.
- [17] Yannakakis M. *Computational Complexity*. in: E. Aarts and J. K. Lenstra (eds.) *Local Search in Combinatorial Optimization*. Chichester: John Wiley & Sons, 1997, p. 19–56.

LOCAL SEARCH IN COMBINATORIAL OPTIMIZATION. WHAT ABOUT THE DERIVATIVE?

Yu. Kochetov, A. Pljasunov

Sobolev Institute of Mathematics, Novosibirsk
e-mail: jkochet@math.nsc.ru, apljas@math.nsc.ru

Abstract. For combinatorial optimization problems it is shown that local optima are closely related with the stationary solutions for continued optimization problems. Local search methods implicitly use the Karush-Kuhn-Tucker conditions and try to select the best stationary solution. This point of view allows us to understand why the local search methods are so powerful and shows directions for further research.

Key words: NP-hard problems, local search, Karush-Kuhn-Tucker conditions, wave equation, fitness landscapes.

СЕКЦИОННЫЕ ДОКЛАДЫ

ЛИНЕЙНОЕ ПРОГРАММИРОВАНИЕ

О СЛОЖНОСТИ ЗАДАЧИ ПАРАМЕТРИЧЕСКОГО ЛИНЕЙНОГО ПРОГРАММИРОВАНИЯ

В.А. Березнев

Вычислительный центр им. А.А.Дородницына РАН, Москва
e-mail: berezn@ccas.ru

Аннотация. Приводится оценка числа изломов кусочно-линейной функции оптимальных значений задачи линейного программирования с параметром в правой части ограничений.

Ключевые слова: линейное программирование, параметр, сложность.

Рассмотрим задачу

$$F(\lambda) = \min_{x \in X_\lambda} \langle c, x \rangle, \quad X_\lambda = \{x \in \mathbb{R}_n \mid Ax = p - \lambda q, \quad x \geq 0\}, \quad (1)$$

где векторы $c, x \in \mathbb{R}_n$, матрица A порядка $m \times n$ ($m < n$) ранга m , векторы q и p из \mathbb{R}_m фиксированы, а $\lambda \geq 0$. Как известно [1], функция $F(\lambda)$ оптимальных значений в задаче (1) представляет собой непрерывную выпуклую кусочно-линейную функцию скалярного параметра λ . Обозначим через Λ область определения функции $F(\lambda)$. Множество Λ (если оно не пусто) представляет собой либо конечный отрезок $[0, \lambda^*]$, либо луч $[0, \infty)$, разбитые на конечное число интервалов линейности функции $F(\lambda)$ так называемыми *точками излома*. Исследуем свойства ломаной $F(\lambda)$ и точки ее излома.

Пусть при некотором $\lambda' \geq 0$ точка $x(\lambda')$ является невырожденной оптимальной угловой точкой множества $X_{\lambda'}$, т.е. $F(\lambda') = \langle c, x(\lambda') \rangle = \min_{x \in X_{\lambda'}} \langle c, x \rangle$. Поскольку ранг матрицы A равен m , то это означает, что существуют такие m линейно независимых вектор-столбцов матрицы A , образующих квадратную невырожденную матрицу B , что $x^{(1)}(\lambda') = B^{-1}(p - \lambda'q) > 0$, $x^{(2)}(\lambda') = 0$, где $x^{(1)}(\lambda') \in \mathbb{R}_m$ составлен из базисных компонент точки $x(\lambda')$, а $x^{(2)}(\lambda') \in \mathbb{R}_{n-m}$ — из небазисных. Не нарушая общности, можно считать, что $B = [A_1, \dots, A_m]$.

Пусть теперь $\lambda = \lambda' + \theta$, $\theta > 0$. Так как вектор базисных компонент $x^{(1)}(\lambda') > 0$, то для достаточно малых $\theta > 0$ имеем $x^{(1)}(\lambda) > 0$. Таким образом, точка $x(\lambda) = (x^{(1)}(\lambda), x^{(2)}(\lambda))^T$, где $x^{(2)}(\lambda) = 0$, будет оставаться невырожденной угловой точкой множества X_λ с тем же базисом B до тех пор, пока какая-либо из компонент вектора $x^{(1)}(\lambda)$ не обратится в нуль.

Компоненты $x_j^{(1)}(\lambda)$, $j \in J_0 = \{1, \dots, n - m\}$ обращаются в нуль при

$$\theta_j = \theta_j(\lambda') = \begin{cases} \frac{x_j^{(1)}(\lambda')}{(B^{-1}q)_j}, & \text{если } (B^{-1}q)_j > 0, \\ +\infty, & \text{если } (B^{-1}q)_j \leq 0. \end{cases}$$

Следовательно, точка $x(\lambda)$ будет оставаться невырожденной оптимальной (поскольку оценки векторов условий A_j не зависят от вектора ограничений) угловой точкой множества X_λ с тем же базисом B для всех λ таких, что $\lambda' \leq \lambda < \lambda''$, где

$$\lambda'' = \lambda' + \min_{j \in J_0^+} \theta_j, \quad J_0^+ = \{j \in J_0 \mid (B^{-1}q)_j > 0\}.$$

Если $J_0^+ = \emptyset$, то $\lambda'' = \infty$, а точка $x(\lambda)$ остается оптимальной невырожденной угловой точкой множества X_λ с тем же базисом B при любом $\lambda \geq \lambda'$. Пусть $J_0^+ \neq \emptyset$. Обозначим $\tilde{J}_0^+ = \{j \in J_0^+ \mid \lambda' + \theta_j = \lambda''\}$.

Вместе с задачей (1) рассмотрим задачу с возмущенным вектором ограничений

$$F_\delta(\lambda) = \min_{x \in X_\lambda(\delta)} \langle c, x \rangle, \quad (2)$$

где $X_\lambda(\delta) = \{x \in \mathbb{R}_n \mid Ax = p + \delta p' - \lambda q, \quad x \geq 0\}$.

Лемма 1 Пусть $J_0^+ \neq \emptyset$ и при некотором фиксированном $\lambda' \in \Lambda$ множество решений $X_{\lambda'}^*$ задачи (1) содержит невырожденную угловую точку $\tilde{x}(\lambda')$. Тогда существует такой вектор p' , что при том же λ' и при любом $\delta > 0$ множество $\tilde{J}_0^+(\delta)$ задачи (2) состоит из единственного индекса, причем $\lambda''(\delta) = \lambda''$.

Действительно, пусть B — базис невырожденной угловой точки $\tilde{x}(\lambda')$. Предположим, что \tilde{J}_0^+ содержит более одного индекса. Выберем любой из них и обозначим его через k . Легко проверить, что утверждение леммы 1 справедливо при выборе $p' = Be^+$, где $e_j^+ = 1$, если $j \in J_0$ и $j \neq k$, и $e_j^+ = 0$, если $j = k$.

При $\lambda = \lambda''$ точка $\tilde{x}(\lambda'', \delta)$ становится вырожденной угловой точкой, причем среди ее базисных компонент лишь одна компонента $\tilde{x}_k(\lambda'', \delta)$ равна нулю. В связи с этим естественно дать следующее определение.

Определение. Угловую точку $x \in \mathbb{R}_n$ многогранного множества X назовем точкой ρ -го порядка вырожденности ($\rho \geq 0$), если среди ее базисных компонент ρ компонент равны нулю.

Таким образом, точка $\tilde{x}(\lambda'', \delta)$ является угловой точкой множества $X_{\lambda''}(\delta)$ первого порядка вырожденности.

Будем считать, что для задачи (1) $\lambda'' < \infty$ и $\min_{j \in J_0^+} \theta_j$ достигается на единственном индексе k . Исследуем ситуацию при $\lambda > \lambda''$. Положим $J_1 = J \setminus J_0$, где J_0 — множество индексов базисных переменных при $\lambda = \lambda''$ и пусть B — базис соответствующего решения $\tilde{x}(\lambda'')$. При $\lambda > \lambda''$ точка $\tilde{x}(\lambda) = (\tilde{x}^{(1)}(\lambda), \tilde{x}^{(2)}(\lambda))$, где $\tilde{x}^{(1)}(\lambda) = B^{-1}(p - \lambda q)$, $\tilde{x}^{(2)}(\lambda) = 0$, являясь решением уравнения $Ax = p - \lambda q$, перестает быть допустимой точкой множества X_λ , так как $\tilde{x}_k(\lambda) < 0$, однако, остальные базисные компоненты этой точки остаются положительными при $\lambda = \lambda'' + \varepsilon$ и достаточно малом $\varepsilon > 0$. Это означает, что вектор $p - \lambda q$ при $\lambda > \lambda''$ выходит за пределы конуса, натянутого на векторы A_j , $j \in J_0$, однако он может принадлежать конусу, полученному его расширением путем добавления к образующим A_j , $j \in J_0$ какого-либо из небазисных векторов A_r . Разложим произвольный небазисный вектор-столбец A_r , $r \in J_1$ по базису B , т.е. представим его в виде $A_r = B\alpha_r$, $\alpha_r = (\alpha_{1r}, \dots, \alpha_{mr})^T \in \mathbb{R}_m$. Легко показать справедливость следующих утверждений.

Лемма 2 Пусть $\tilde{x}(\lambda'')$ — угловая точка множества $X_{\lambda''}$ первого порядка вырожденности, причем $\tilde{x}_k(\lambda'') = 0$ и $\tilde{x}_k(\lambda) < 0$ при $\lambda > \lambda''$. Пусть B — базис этой угловой точки. Тогда если $\alpha_{kr} \geq 0$ для всех индексов $r \in J_1$, то система уравнений $Ax = p - \lambda q$ не имеет неотрицательных решений при $\lambda > \lambda''$, т.е. $X_\lambda = \emptyset$. Если же $\alpha_{kr} < 0$ для некоторого $r \in J_1$, то точка $\tilde{x}(\lambda)$, у которой $\tilde{x}^{(1)}(\lambda) = B^{-1}(p - \lambda q)$, $\tilde{x}^{(2)}(\lambda) = 0$, является невырожденной угловой точкой множества X_λ при $\lambda = \lambda'' + \varepsilon$ и достаточно малом $\varepsilon > 0$.

Из леммы 2 следует, что при анализе любой точки излома функции $F(\lambda)$, когда $\tilde{x}(\lambda'')$ является угловой точкой множества $X_{\lambda''}$ первого порядка вырожденности, у которой базисная компонента $\tilde{x}_k^{(1)}(\lambda'') = 0$, возможны две ситуации.

а) Коэффициенты $\alpha_{kr} \geq 0$ для всех $r \in J_1$. Тогда при $\lambda > \lambda''$ множество $X_\lambda = \emptyset$, а функция $F(\lambda)$ определена на конечном отрезке $\Lambda = [0, \lambda^*]$, где $\lambda^* = \lambda''$.

б) Существует хотя бы один индекс $r \in J_1$, для которого $\alpha_{kr} < 0$. Тогда заменой в базисе B вектора A_k на вектор A_r можно перейти к новому базису \bar{B} . При этом любой новый базис, полученный такой заменой, оказывается базисом невырожденной угловой точки $\bar{x}(\lambda)$ множества X_λ при некотором $\lambda > \lambda''$.

Лемма 3 Пусть выполнены условия леммы 2, т.е. $X_\lambda \neq \emptyset$ при $\lambda = \lambda'' + \varepsilon$ и достаточно малом $\varepsilon > 0$. Тогда точка $\bar{x}(\lambda) = \bar{x}(\lambda, r)$, для которой $\bar{x}^{(1)}(\lambda) = \bar{B}^{-1}(p - \lambda q)$, а $\bar{x}^{(2)}(\lambda) = 0$, где матрица \bar{B} получена заменой в базисе B вектора A_k на вектор A_r , а индекс r выбирается из условия

$$-\frac{\Delta_r}{\alpha_{kr}} = \min_{\alpha_{kj} < 0, j \in J_1} \left(-\frac{\Delta_j}{\alpha_{kj}} \right),$$

удовлетворяет достаточному условию оптимальности, т.е. $\Delta_j = \langle c^{(1)}, \bar{B}A_j \rangle - c_j \leq 0$ для всех $j \in J = \{1, \dots, n\}$, где вектор $c^{(1)} \in \mathbb{R}_m$ составлен из базисных компонент c .

Переобозначим вычисленную точку излома λ'' ломаной $F(\lambda)$ через λ_1 , считая $\lambda_0 = 0$, полученную невырожденную угловую точку — через $x^1(\lambda)$ и продолжим процесс перехода от одной точки излома к другой, т.е. определим λ_2 и т.д. Заметим, что в случае, когда каждой точке излома λ_ν соответствует угловая точка $x^\nu(\lambda_\nu)$ первого порядка вырожденности, в каждом интервале $(\lambda_\nu, \lambda_{\nu+1})$ решение $x^\nu(\lambda)$ не вырождено. Следовательно, в этой точке вновь реализуется одна из указанных выше двух ситуаций а) или б). В результате N таких итераций будут получены точки $\lambda_0 = 0, \lambda_1, \dots, \lambda_N$, базисы B^0, B^1, \dots, B^N и соответствующие им невырожденные угловые точки $x^\nu(\lambda)$, $\nu = 0, N-1$, каждая из которых является оптимальным решением с базисом B^ν задачи (1) в интервале значений параметра $(\lambda_\nu, \lambda_{\nu+1})$. Важно отметить, что если переход от базиса $B^{\nu-1}$ к базису B^ν был связан с заменой k -го столбца в $B^{\nu-1}$, то переход от B^ν к $B^{\nu+1}$ связан с заменой в B^ν столбца с индексом l , причем $l \neq k$.

Обозначим через K^0 выпуклый конус, натянутый на векторы A_j , $j \in J = \{1, \dots, n\}$. Пусть для некоторого $\nu \geq 1$ базисом угловой точки $x^\nu(\lambda)$ является матрица B^ν . Не нарушая общности, будем считать, что $B^\nu = [A_1 \dots A_m]$. Предположим, что при переходе в точке излома $\lambda_{\nu+1}$ от базиса B^ν к базису $B^{\nu+1}$ согласно лемме 3 на место вектора A_k введен вектор A_r , для которого $\alpha_{kr} < 0$. Тогда матрица $C^\nu = [A_1 \dots A_{k-1} A_{k+1} \dots A_m]$ размерности $m \times (m-1)$ определяет в пространстве \mathbb{R}_m подпространство H_ν размерности $m-1$ и выпуклый конус $K^\nu = K^0 \cap H_\nu$, натянутые на вектор-столбцы матрицы C^ν . При этом любой вектор $\xi^\nu \neq 0$, такой что $(C^\nu)^T \xi^\nu = 0$, принадлежит ортогональному дополнению H_ν^\perp . Поскольку $A_k \notin H_\nu$, то $\langle A_k, \xi^\nu \rangle \neq 0$. Выберем вектор ξ^ν так, чтобы выполнялось неравенство $\langle A_k, \xi^\nu \rangle > 0$. Новый базис $B^{\nu+1} = [A_1 \dots A_{k-1} A_r A_{k+1} \dots A_m]$. Учитывая, что $\alpha_{kr} < 0$ и $\langle A_k, \xi^\nu \rangle > 0$, имеем

$$\langle A_r, \xi^\nu \rangle = \left\langle \sum_{i=1}^m \alpha_{ir} A_i, \xi^\nu \right\rangle = \sum_{i=1, i \neq k}^m \alpha_{ir} \langle A_i, \xi^\nu \rangle + \alpha_{kr} \langle A_k, \xi^\nu \rangle = \alpha_{kr} \langle A_k, \xi^\nu \rangle < 0.$$

Таким образом, если $\xi^\nu \in H_\nu^\perp$ выбирается из условия $\langle A_k, \xi^\nu \rangle > 0$, то вводимый в базис вектор должен удовлетворять неравенству $\langle A_r, \xi^\nu \rangle < 0$. В связи с этим для любого

$\nu \geq 1$ определим подмножество $U_\nu = \{A_i \mid \langle A_i, \xi^\nu \rangle \geq 0\}$ вектор-столбцов матрицы A , не подлежащих вводу в базис на ν -й итерации. Таким образом, $A_k \in U_\nu$ и $A_r \notin U_\nu$. Ключевую роль в оценке числа изломов функции $F(\lambda)$ играет следующее утверждение.

Теорема 1 Пусть выполнены N итераций описанного выше процесса построения ломаной $F(\lambda)$, причем для каждого ν угловая точка $x^\nu(\lambda_\nu)$ является вершиной множества X_λ первого порядка вырожденности. Тогда, если на некоторой ν -й итерации при переходе от точки первого порядка вырожденности $x^\nu(\lambda_\nu)$ к невырожденной точке $x^{\nu+1}(\lambda)$ из базиса B^ν для получения базиса $B^{\nu+1}$ исключен вектор A_k , то $A_k \notin B^{\nu+i}$ для любого $i = 1, \dots, N - \nu$.

Доказательство. Пусть для некоторого $\rho > \nu$ базисом точки $x^\rho(\lambda_\rho)$ является матрица $B^\rho = [A_{i_1} \dots A_{i_m}] = [B_1^\rho \dots B_m^\rho]$. Предположим, что при переходе к базису $B^{\rho+1}$ из B^ρ выводится вектор A_{i_s} и вводится вектор A_q , для которого $\alpha_{sq} < 0$. Тогда $C^\rho = [B_1^\rho \dots B_{s-1}^\rho B_{s+1}^\rho \dots B_m^\rho]$, система $(C^\rho)^T \xi^\rho = 0$ определяет вектор $\xi^\rho \in H_\rho^\perp$, а новый базис имеет вид $B^{\rho+1} = [B_1^\rho \dots B_{s-1}^\rho A_q B_{s+1}^\rho \dots B_m^\rho]$. Выбрав ξ^ρ так, чтобы выполнялось неравенство $\langle A_{i_s}, \xi^\rho \rangle > 0$, построим соответствующее множество $U_\rho = \{A_i \mid \langle A_i, \xi^\rho \rangle \geq 0\}$. Утверждение теоремы будет доказано, если покажем, что $A_k \in U_{\nu+i}$ для любого $i = 1, \dots, N - \nu$. Для доказательства воспользуемся индукцией по номеру i , т.е. предположим, что $A_k \in U_\rho$, где $\rho \geq \nu$, и покажем, что тогда $A_k \in U_{\rho+1}$. Принадлежность вектора A_k множеству U_ν показана выше.

Так как по предположению индукции $A_k \in U_\rho$, то для него справедливо неравенство $\langle A_k, \xi^\rho \rangle \geq 0$, а для A_q — неравенство $\langle A_q, \xi^\rho \rangle < 0$. Тогда, во-первых, для любого $\mu \in [0, 1]$ вектор $u(\mu) = \mu A_k + (1 - \mu) A_q$ принадлежит конусу K^0 , а, во-вторых, существует такое число $\mu^* \in [0, 1]$, что $\langle u(\mu^*), \xi^\rho \rangle = 0$, т.е. вектор $u(\mu^*) \in K^0 \cap H_\rho = K^\rho$. Из того, что $u(\mu^*) \in H_\rho$, следует, что

$$u(\mu^*) = \sum_{j=1, j \neq s}^n \beta_j B_j^\rho,$$

а в силу принадлежности $u(\mu^*)$ конусу K^0 коэффициенты $\beta_j \geq 0$ для всех $j = \overline{1, n}$, $j \neq s$. Заметим, что при $\mu^* = 0$ вектор $u(0) = A_q$, т.е. $\langle u(0), \xi^\rho \rangle = \langle A_q, \xi^\rho \rangle < 0$. Следовательно, $\mu^* > 0$. Тогда справедливо равенство $\mu^* A_k = u(\mu^*) - (1 - \mu^*) A_q$ или

$$A_k = \frac{1}{\mu^*} u(\mu^*) - \frac{(1 - \mu^*)}{\mu^*} A_q = \sum_{j=1, j \neq s}^n \frac{\beta_j}{\mu^*} B_j^\rho - \frac{(1 - \mu^*)}{\mu^*} A_q.$$

Пусть на следующей $(\rho + 1)$ -й итерации из базиса $B^{\rho+1}$ выводится вектор $B_l^{\rho+1} = B_l^\rho$ и вводится вектор A_t , для которого $\alpha_{lt} < 0$. Заметим, что $B_l^\rho \neq A_q$, т.е. $l \neq s$.

Вновь выберем вектор $\xi^{\rho+1}$ так, чтобы выполнялось неравенство $\langle B_l^{\rho+1}, \xi^{\rho+1} \rangle = \langle B_l^\rho, \xi^{\rho+1} \rangle > 0$, а для всех $j \neq l$ равенства $\langle B_j^{\rho+1}, \xi^{\rho+1} \rangle = 0$, т.е. $\langle B_j^\rho, \xi^{\rho+1} \rangle = 0$ для $j = \overline{1, m}$, $j \neq s, l$ и $\langle A_q, \xi^{\rho+1} \rangle = 0$. Тогда для вектора A_k имеем

$$\langle A_k, \xi^{\rho+1} \rangle = \sum_{j=1, j \neq s}^n \frac{\beta_j}{\mu^*} \langle B_j^\rho, \xi^{\rho+1} \rangle - \frac{(1 - \mu^*)}{\mu^*} \langle A_q, \xi^{\rho+1} \rangle.$$

Учитывая, что $\langle B_j^\rho, \xi^{\rho+1} \rangle = 0$ для всех $j \neq l, s$ и $\langle A_q, \xi^{\rho+1} \rangle = 0$, имеем

$$\langle A_k, \xi^{\rho+1} \rangle = \frac{\beta_l}{\mu^*} \langle B_l^{\rho+1}, \xi^{\rho+1} \rangle = \frac{\beta_l}{\mu^*} \langle B_l^\rho, \xi^{\rho+1} \rangle \geq 0,$$

так как $\beta_l \geq 0$, $\mu^* > 0$ и $\langle B_l^p, \xi^{\rho+1} \rangle > 0$. Таким образом, вектор A_k принадлежит $U_{\rho+1}$ — множеству векторов, не подлежащих вводу в базис на очередной итерации. ■

Как следует из теоремы 1, число итераций процесса (т.е. число точек излома функции $F(\lambda)$) не превосходит $n - m$, так как на каждой из них множество U_ν увеличивается по крайней мере на один вектор, исключаемый из базиса на этой итерации. Учитывая это, приходим к утверждению, что на вычисление значений функции $F(\lambda)$ в ее точках излома затрачивается порядка $O(n^2 m^2)$ элементарных операций.

Действительно, легко видеть, что вычисление вектора α_r коэффициентов разложения произвольного небазисного вектора A_r по базису B^ν связано с затратами $O(m^2)$ элементарных операций. Число небазисных вектор-столбцов матрицы A равно $n - m$, следовательно, на одну итерацию процесса затрачивается порядка $O(nm^2 - m^3)$ элементарных операций. Учитывая, что число итераций (точек излома функции $F(\lambda)$) по теореме 1 не превосходит $n - m$, получаем указанную оценку.

Таким образом, вычислительная сложность алгоритма, связанного с вычислением функции $F(\lambda)$ в точках ее излома, полиномиально зависит от размерности задачи, если при $\lambda = 0$ значению $F(0)$ отвечает угловая точка \bar{x} , удовлетворяющая достаточному условию оптимальности, и каждой точке излома функции $F(\lambda)$ отвечает угловая точка множества X_λ первого порядка вырожденности. Если эти условия не выполнены, то требуется возмущать вектор ограничений $p - \lambda q$ и заменять его вектором $p + \delta p' - \lambda q$, в результате чего вместо $F(\lambda)$ будет построена другая ломаная $F_\delta(\lambda)$, которая будет удовлетворять всем условиям теоремы 1. Справедлива следующая

Теорема 2 Пусть $x(\lambda', \delta)$ — решение задачи (2) при $\lambda = \lambda'$, где λ' — некоторая точка излома функции $F_\delta(\lambda)$, являющаяся оптимальной угловой точкой множества $X_{\lambda'}(\delta)$ первого порядка вырожденности с базисом $B = [B_{i_1}, \dots, B_{i_m}]$, у которой базисная компонента $x_{i_k}(\lambda', \delta) = 0$. Пусть $\hat{B} = [B_{i_1}, \dots, B_{i_{k-1}}, q, B_{i_{k+1}}, \dots, B_{i_m}]$. Тогда вектор $\tilde{x} = (\tilde{x}^{(1)}, \tilde{x}^{(2)}) \in \mathbb{R}_n$, где $\tilde{x}^{(2)} = 0$, $\tilde{x}^{(2)} \in \mathbb{R}_{n-m}$, $\tilde{x}^{(1)} = \hat{B}^{-1}p \in \mathbb{R}_m$, является угловой точкой множества $X_{\lambda'}$ (возможно, вырожденной), удовлетворяющей достаточному условию оптимальности.

Список литературы

- [1] Гольштейн Е.Г., Юдин Д.Б. Новые направления в линейном программировании. М.: Советское радио, 1966.

COMPLEXITY OF PARAMETRIC LINEAR PROGRAMMING PROBLEM

V.A. Bereznev

Dorodnicyn Computing Center of RAS, Moscow
e-mail: berezn@ccas.ru

Abstract. Evaluation of number of breaks of optimal values function of linear programming problem with parameter in right side of limitations is given.

Key words: linear programming, parameter, complexity

ИСПОЛЬЗОВАНИЕ ТОЧНЫХ И ПРИБЛИЖЕННЫХ РЕШЕНИЙ В ПАРАЛЛЕЛЬНОМ МЕТОДЕ ПРЯМО-ДВОЙСТВЕННЫХ ОТСЕЧЕНИЙ ДЛЯ БЛОЧНЫХ ЗАДАЧ ЛИНЕЙНОГО ПРОГРАММИРОВАНИЯ

А.С. Величко

Институт автоматизации и процессов управления, Владивосток
e-mail: vandre@dvo.ru

Аннотация. В статье показан способ построения плотных и ε -отсечений, которые используются в параллельном декомпозиционном алгоритме прямо-двойственных отсечений, применяемом для двублочных задач линейного программирования. Задача построения таких отсечений основана на вычислении точных или ε -субградиентов кусочно-линейных функций отклика и сводится к точному или приближенному решению подзадач алгоритма. Доказана сходимость алгоритма с использованием ε -отсечений.

Ключевые слова: декомпозиция, отсечение, блочный, линейный, программирование, приближенный, ε -субградиент.

Введение

В работе [2] был предложен прямо-двойственный алгоритм, используемый для решения двублочных оптимизационных задач, и доказана его сходимость при достаточно общих предположениях. В случае возникновения недопустимых и неограниченных подзадач алгоритма необходима его модификация [1]. Вычислительные эксперименты проводились для двублочных задач декомпозиции метода конечных элементов, стохастического программирования и экономической задачи о репликации портфеля рыночных активов. В этих различных по структуре задачах алгоритм продемонстрировал полиномиальную вычислительную сложность $O(n^k)$ с показателем степени k порядка 4.

Выигрыш от использования параллелизма составляет в большинстве тестов 20-25%. Вычислительные эксперименты проводились как на отдельных вычислительных узлах в локальной сети, так и на многопроцессорных вычислительных комплексах МВС-1000/16, МВС-1000/17 в составе Центра коллективного пользования "Дальневосточный вычислительный ресурс" (ЦКП ДВВР) [3].

Перспективным направлением развития данного алгоритма является использование ε -отсечений, основанных на приближенном решении подзадач алгоритма. Для этого необходимо уметь вычислять ε -субградиенты кусочно-линейных функций отклика. Важным для приложений является вопрос использования минимальной доступной прямо-двойственной информации, получаемой в процессе приближенного итеративного решения подзадач.

1. Теоретическая постановка структурированной линейной оптимизационной задачи

В качестве основной модели структурированной оптимизационной задачи рассмотрим

двублочную проблему следующего вида:

$$\min_{z_A, z_B, x} c_A z_A + c_B z_B, \quad (1)$$

$$A_A z_A + B_A x \leq d_A, \quad (2)$$

$$A_B z_B + B_B x \leq d_B, \quad (3)$$

$$z_A \geq 0, z_B \geq 0, \quad (4)$$

где z_A, z_B – векторы переменных задачи, x – вектор связывающих переменных, A_A, A_B, B_A, B_B – матрицы соответствующих размерностей.

При фиксированном x эта задача распадается на два независимых блока, что используется для развития декомпозиционного алгоритма двойственных отсечений.

Введя функции отклика

$$f_A(x) = \min_{z_A \geq 0} c_A z_A, \quad f_B(x) = \min_{z_B \geq 0} c_B z_B, \quad (5)$$

$$A_A z_A \leq d_A - B_A x \quad A_B z_B \leq d_B - B_B x$$

получим эквивалентную для (1)-(4) задачу:

$$\min_x \{f_A(x) + f_B(x)\}. \quad (6)$$

Используя сопряженные функции $h_A(p)$ и $h_B(p)$:

$$h_A(p) = f_A^*(x) = \max_x \{px - f_A(x)\}, \quad h_B(p) = f_B^*(x) = \max_x \{px - f_B(x)\}$$

задачу (6) легко переписать в терминах сопряженных функций:

$$\min_p \{h_A(-p) + h_B(p)\}. \quad (7)$$

Такая эквивалентность задач (6) и (7) позволяет организовать эффективный процесс обмена координирующей информацией между прямыми и двойственными задачами линейного программирования в декомпозиционном подходе к решению задачи (1)-(4).

2. Построение плотных отсечений в параллельном алгоритме прямо-двойственных отсечений

k -ая итерация параллельного алгоритма прямо-двойственных отсечений для задачи (6) показана на рисунке 3.

Кусочно-линейные аппроксимации $f_B^k(x) = \max\{f_B^{k-1}(x), \bar{p}^k x - h_B(\bar{p}^k), p^k x - h_B(p^k)\}$ и $h_A^k(p) = \max\{h_A^{k-1}(p), p\bar{x}^k - f_A(\bar{x}^k), px^{k+1} - f_A(x^{k+1})\}$ используют информацию о всех отсечениях, уже построенных до данного шага. После выполнения вычислений в каждом из блоков, осуществляется обмен значениями x^{k+1}, p^{k+1} между процессами для дальнейших вычислений на следующей итерации алгоритма.

Важными вычислительными задачами алгоритма являются вычисление субградиентов функций $f_B(\cdot), h_A(\cdot)$.

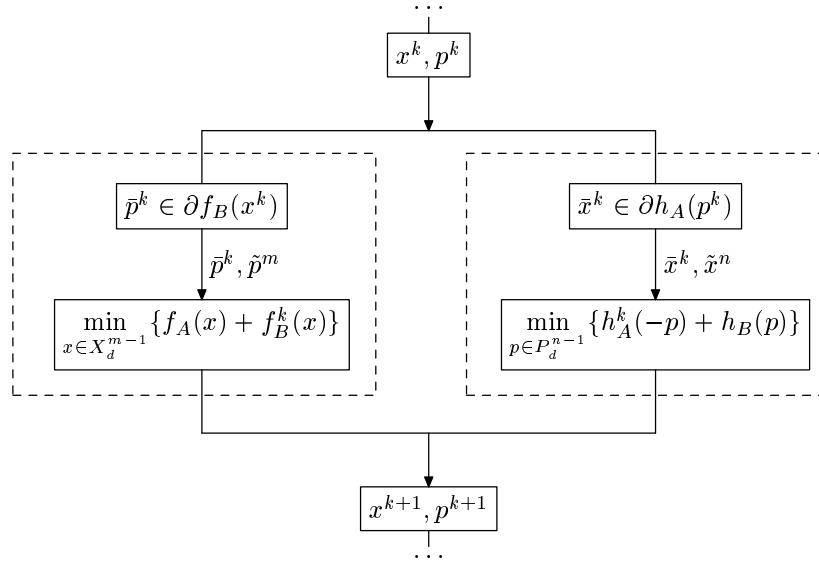
Пусть $f_B(x)$ определена соотношением (5). Тогда для некоторого фиксированного вектора \bar{x} имеем

$$f_B(\bar{x}) = \min_{z_B, x} c_B z_B, \quad (8)$$

$$A_B z_B + B_B x \leq d_B,$$

$$x = \bar{x}, z_B \geq 0.$$

Рис. 3. k -ый шаг параллельного алгоритма прямо-двойственных отсечений.



Из стандартной теории двойственности линейного программирования и теории выпуклого анализа легко получить следующее утверждение.

Теорема 1 Пусть задача (8) разрешима для заданного \bar{x} , p^* является вектором оптимальных двойственных переменных к ограничению $x = \bar{x}$ в задаче (8). Тогда $p^* \in \partial f_B(\bar{x})$, и обратно, любой вектор $p^* \in \partial f_B(\bar{x})$ является вектором оптимальных двойственных переменных, соответствующих ограничению $x = \bar{x}$ в задаче (8).

В процессе выполнения алгоритма могут возникать некорректные задачи, в работе [1] показано как построить дополнительные отсечения и модифицировать алгоритм в случаях недопустимости или неограниченности возникающих подзадач.

Решить проблему неразрешимости подзадач алгоритма двойственных отсечений можно, добавляя дополнительные отсечения. При вычислении $f_B(x^k)$ и $\bar{p}^k \in \partial f_B(x^k)$ находится отсечение вида $p_d^m x \leq p_d^m x^k - \phi(x^k)$, где $\phi(x^k)$ определяется в результате решения задачи

$$\phi(x^k) = \min_{z_B, x, s} \|s\|_1 \quad (9)$$

$$A_B z_B + B_B x \leq d_B$$

$$x + s = x^k$$

$$z_B \geq 0.$$

По теореме 1 p_d^m находится как двойственный вектор к ограничению $x + s = x^k$ в задаче (9) поиска допустимого базиса для последующего решения задачи (8).

Такие отсечения отделяют область недопустимых аргументов x^k от эффективного множества функции $f_B(x)$, которое описывается условием $f_B(x) < +\infty$, $f_B(x) \neq -\infty$ и которое обозначается как $\text{dom } f_B$. Отсечения аппроксимируют $\text{dom } f_B$ и формируют множество допустимости $X_d^m = X_d^{m-1} \cup \{x : p_d^m x \leq h_d^m = p_d^m x^k - \phi(x^k)\}$, что препятствует возникновению недопустимости на следующих итерациях алгоритма. Отсечение, определяемое после решения подзадачи (9), показано на рисунке 1. Возможны и другие стратегии построения отсечений. Используя евклидову метрику для задачи минимизации нормы вектора невязок в задаче (9), получим вектор \tilde{q}^k , показанный на рисунке 2.

Сложность решения задачи (8) на практике заключается в том, что невозможно использовать стратегию досчета, часто применяемую для класса методов отсечений. Ее использование возможно, когда оптимальное решение некоторой подзадачи можно использовать в качестве допустимого для решения этой задачи на следующей итерации алгоритма.

С другой стороны, такой досчет можно осуществить в двойственной постановке, однако размерность базиса в последней задаче существенно выше, чем в исходной. Поэтому возникает вопрос об эффективном использовании информации в процессе приближенного решения задачи (8) симплекс-методом для построения ε -отсечений.

Рис. 1.

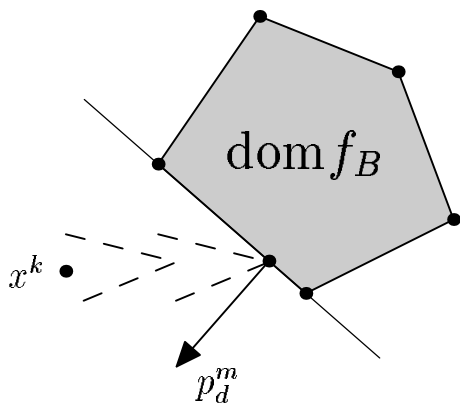
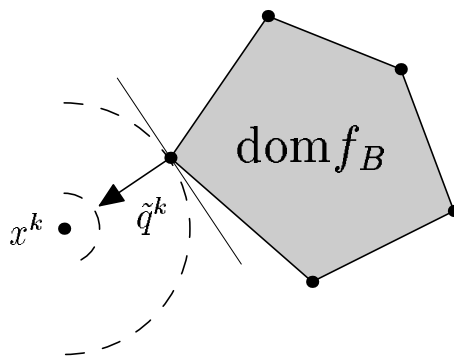


Рис. 2.



3. Построение ε -отсечений и использование приближенных решений подзадач алгоритма

В приведенных выше формулировках f_B , h_A являются выпуклыми кусочно-линейными функциями, в этих условиях удастся доказать сходимость алгоритма с использованием приближенных решений подзадач.

Теорема 2 Пусть f_B , h_A , определенные в (5), являются конечными, выпуклыми и кусочно-линейными, задачи поиска субградиентов функций f_B , h_A решаются приближенно с некоторой точностью ε_k и $\lim_{k \rightarrow \infty} \varepsilon_k = 0$. Последовательности $\{x^k\}$, $\{p^k\}$, генерируемые алгоритмом, ограничены. Тогда все предельные точки последовательностей $\{x^k\}$, $\{p^k\}$ являются решениями задач (6), (7) соответственно.

Доказательство:

В силу теорем двойственности для $x^k \in \text{dom } f_B$ задача вычисления $\bar{p}^k \in \partial f_B(x^k)$ может быть представлена в виде

$$\begin{aligned} f_B(x^k) = \max_{u, p} & du + x^k p, \\ & u' A_B \leq c, \\ & u' B_B + p_B \leq 0, \\ & u \leq 0, \end{aligned} \quad (10)$$

и пусть $(u_\varepsilon^k, p_\varepsilon^k)$ – приближенное решение этой задачи с точностью $\varepsilon \geq 0$, то есть $f_B(x^k) = \tilde{f}_B^k + \varepsilon = du_\varepsilon^k + x^k p_\varepsilon^k + \varepsilon$.

Заметим, что $f_B(x) \geq du_\varepsilon^k + x p_\varepsilon^k$ для $x \in \text{dom } f_B$, и, следовательно,

$$f_B(x) - f_B(x^k) \geq du_\varepsilon^k + x p_\varepsilon^k - (du_\varepsilon^k + x^k p_\varepsilon^k + \varepsilon) = (x - x^k) p_\varepsilon^k - \varepsilon,$$

то есть $p_\varepsilon^k \in \partial_\varepsilon f_B(x^k)$.

По построению последовательность функций $\{f_B^k(x)\}$ удовлетворяет неравенствам:

$$f_B^k(x) \leq f_B^{k+1}(x) \leq f_B(x) < +\infty. \quad (11)$$

Неравенство $f_B^{k+1}(x) \leq f_B(x)$ покажем по индукции.

$$f_B^{k+1}(x) = \max\{f_B^k(x), \tilde{f}_B^k + p_{\varepsilon_k}^k(x - x^k)\}, \text{ где } p_{\varepsilon_k}^k \in \partial_{\varepsilon_k} f_B(x^k).$$

$f_B^0(x) = \max\{-\infty, \tilde{f}_B^0 + p_{\varepsilon_0}^0(x - x^0)\} = f_B(x^0) - \varepsilon_0 + p_{\varepsilon_0}^0(x - x^0)$, где $p_{\varepsilon_0}^0 \in \partial_{\varepsilon_0} f_B(x^0)$. По определению ε -субградиента $f_B(x^0) - \varepsilon_0 + p_{\varepsilon_0}^0(x - x^0) \leq f_B(x)$, и значит $f_B^0(x) \leq f_B(x)$.

Предполагая, что неравенство $f_B^k(x) \leq f_B(x)$ выполняется, имеем

$$f_B^{k+1}(x) = \max\{f_B^k(x), \tilde{f}_B^k + p_{\varepsilon_k}^k(x - x^k)\} \leq f_B(x),$$

так как $p_{\varepsilon_k}^k \in \partial_{\varepsilon_k} f_B(x^k)$ и по определению ε -субградиента $\tilde{f}_B^k + p_{\varepsilon_k}^k(x - x^k) \leq f_B(x)$. Таким образом показано, что $f_B^{k+1}(x) \leq f_B(x)$. Кусочно-линейная аппроксимация $f_B^{k+1}(x)$ использует только ε -отсечения, основанные на использовании ε -субградиентов p_{ε}^k .

Неравенство (11) обеспечивает сходимость последовательности функций $f_B^k(x)$ к некоторой конечной выпуклой функции $f_B(x) = \lim_{k \rightarrow \infty} f_B^k(x)$.

С другой стороны, для x^k , являющегося решением задачи $\min_x f_A(x) + f_B^{k+1}(x)$, имеем

$$f_B^{k+1}(x^k) = \max\{f_B^k(x^k), \tilde{f}_B^k\} \geq f_B(x^k) - \varepsilon_k.$$

Из последнего неравенства и (11) следует, что $f_B(x^k) \geq f_B^{k+1}(x^k) \geq f_B(x^k) - \varepsilon_k$. Тогда для любой предельной точки x последовательности $\{x^k\}$ $f_B(x) = f_B(x)$, поскольку $\lim_{k \rightarrow \infty} \varepsilon_k = 0$.

Оптимальность x следует из цепочки неравенств $f_A(x) + f_B(x) = f_A(x) + f_B(x) = \lim_{k \rightarrow \infty} \{f_A(x^k) + f_B^{k+1}(x^k)\} \leq f_A(x) + f_B^{k+1}(x) \leq f_A(x) + f_B(x)$, откуда следует, что x решает задачу $\min_x \{f_A(x) + f_B(x)\}$.

Аналогично показывается оптимальность произвольной предельной точки p последовательности $\{p^k\}$. ■

Недостатком теоремы 2 является необходимость решения двойственной задачи (10). Можно воспользоваться прямо-двойственным симплекс-методом. В этом случае приближенные решения двойственной задачи (10) являются допустимыми для прямой задачи (8), и тогда, являются ε -субградиентами, что следует из доказательства теоремы 2.

Список литературы

- [1] Величко, А.С., Нурминский, Е.А. *Опыт декомпозиции метода конечных элементов с использованием теории структурированных оптимизационных задач*. - Электронный журнал "ИССЛЕДОВАНО В РОССИИ", 2002, с. 1237-1256. <http://zhurnal.ape.relarn.ru/articles/2002/113.pdf>.
- [2] Нурминский, Е.А. *Численные методы выпуклой оптимизации*. М.: Наука, 1991.
- [3] Центр коллективного пользования "Дальневосточный вычислительный ресурс" ИА-ПУ ДВО РАН. <http://www.dvo.ru/bbc>.

USING EXACT AND APPROXIMATE SOLUTIONS IN PARALLEL PRIMAL-DUAL CUTS METHOD FOR BLOCK-ANGULAR PROBLEMS OF LINEAR PROGRAMMING

A.S. Velichko

*Institute of Automatics and Control Processes, Vladivostok
e-mail: vandre@dvo.ru*

Abstract. In the article the method of dense and ε -cuts construction is exhibited. These cuts are used in parallel decomposition algorithm with primal-dual cuts which is applied for block-angular problems of linear programming. The problem of construction of such cuts is based on exact or ε -subgradients calculation of piecewise linear response functions and is reduced to exact or approximate solving of algorithm subtasks. The convergence of algorithm with ε -cuts is proved.

Key words: decomposition, cut, block-angular, linear, programming, approximate, ε -subgradient

АЛГОРИТМ ОПРЕДЕЛЕНИЯ СОВМЕСТИМОСТИ СИСТЕМ ЛИНЕЙНЫХ ДВУЧЛЕННЫХ НЕРАВЕНСТВ

А.С. Герасимов, Н.К. Косовский

Санкт-Петербургский государственный университет, математико-механический факультет, Санкт-Петербург
e-mail: gas-teach@yandex.ru, kosov@nk1022.spb.edu

Аннотация. Уточнен и реализован алгоритм Давыдка, определяющий совместность систем линейных двучленных неравенств. Приводится уточненное описание алгоритма, соответствующее его реализации на языке программирования Java. Получены оценки временной и емкостной сложности реализованного алгоритма, а также оценка длины всех чисел, используемых в работе алгоритма.

Ключевые слова: линейное программирование, двучленные линейные неравенства, системы, совместность, истинно полиномиальный алгоритм, сложность, Java.

Введение

Рассмотрим задачу определения совместности системы из m линейных неравенств с целочисленными коэффициентами относительно n рациональнозначных переменных.

Для решения этой задачи можно применить полиномиальные алгоритмы Л.Г. Хачияна ([1]) или Н. Кармаркара ([2]). Число выполняемых этими алгоритмами арифметических операций зависит от длины коэффициентов системы.

Алгоритм решения рассматриваемой задачи называется истинно полиномиальным ([3], с. 304), если он совершает полиномиальное по m и n число арифметических операций (сложение, умножение, деление, сравнение) над рациональными числами, длина которых полиномиально ограничена длиной входа.

До сих пор остается открытым вопрос о существовании истинно полиномиального алгоритма для определения совместности систем общего вида. Однако известны истинно полиномиальные алгоритмы для решения систем линейных неравенств с не более чем двумя переменными в каждом неравенстве, причем допускаются только нестрогие неравенства (см., например, [4]).

В [5] предложен эффективный алгоритм определения совместности систем, с одной стороны, более узкого класса — рассматриваются системы линейных двучленных неравенств, а с другой стороны, более широкого — допускаются строгие и нестрогие неравенства. Также в [5] показано, что этот алгоритм является истинно полиномиальным и требует $O(n^3 + m)$ арифметических операций над числами длины $O(I)$, где I — длина входа.

Ниже даны постановка задачи и основные определения, приводятся уточненное описание алгоритма, соответствующее его реализации на языке программирования Java [6, 7], и оценки сложности реализованного алгоритма.

1. Постановка задачи и основные определения

Для удобства изложения наряду с основными переменными, входящими в систему

неравенств, введем дополнительные переменные $\mathbf{0}$ и $\mathbf{1}$ для чисел 0 и 1.

Под ограничением будем понимать линейное неравенство вида $ax + by \prec 0$, где \prec — знак $<$ или \leq , a и b — целые числа — коэффициенты, x и y — переменные (основные или дополнительные).

Системой назовем конъюнкцию конечного числа ограничений. Требуется определить, существует ли вектор рациональных значений основных переменных, удовлетворяющий каждое ограничение системы, то есть требуется определить, совместна ли система.

Ограничение $ax + by \prec 0$ назовем приведенным, если (1) обе переменные x и y — нули, или переменные не совпадают; и (2) если ровно одна из этих переменных — нуль, то абсолютная величина коэффициента при другой переменной равна 1; и (3) если переменная — нуль, то коэффициент при этой переменной — тоже нуль. Приведенное ограничение, равносильное ограничению C , назовем приведением C и обозначим $red(C)$.

Ограничение $ax + by \prec 0$ назовем правильным, если оно приведенное и переменные x и y не являются нулями одновременно.

Обозначим $S^{\{x,y\}}$ множество ограничений системы S , в которые одновременно входят переменные x и y .

Зафиксируем константу $r_0 = 6$ (ее выбор объясняется нижеследующим описанием алгоритма). Будем называть систему S правильной, если все ограничения в ней правильны, и для любых двух переменных x, y выполняется $|S^{\{x,y\}}| \leq r_0$.

Пусть правильные ограничения $C_1: a_1x + b_1y \prec_1 0$ и $C_2: a_2y + b_2z \prec_2 0$ таковы, что $b_1a_2 < 0$. Определим композицию этих ограничений относительно переменной y как $C_1 \circ_y C_2 : a_1|a_2|x + b_2|b_1|z \prec 0$, где \prec — знак \leq , если оба \prec_1 и \prec_2 — нестрогие, и $<$ иначе. Если $b_1a_2 \geq 0$, то композиция $C_1 \circ_y C_2$ не определена.

2. Описание алгоритма

Алгоритм заключается в последовательном исключении переменных ([3], с. 239–242) и одновременном с исключением переменной удалении избыточных неравенств так, что количество неравенств, содержащих любые две переменные, ограничено константой r_0 .

При таком подходе часто выполняется операция выбора всех неравенств, содержащих две данные переменные. Для эффективной реализации этой операции введен строгий линейный порядок на множестве неупорядоченных пар переменных, и используется сбалансированное дерево поиска для хранения неравенств.

Определим строгий линейный порядок на множестве неупорядоченных пар переменных. Неупорядоченная пара переменных $\{u_1, u_2\}$ предшествует паре $\{w_1, w_2\}$, если эти пары не равны, и соответствующая упорядоченная пара (u_{i_1}, u_{i_2}) , где $u_{i_1} \prec_V u_{i_2}$, лексикографически предшествует соответствующей упорядоченной паре (w_{j_1}, w_{j_2}) , где $w_{j_1} \prec_V w_{j_2}$. При этом порядок \prec_V на множестве переменных можно считать лексикографическим строковым порядком, либо обычным порядком на натуральных числах, если вместо имен переменных использовать их номера.

Для представления системы S в памяти компьютера используется сбалансированное дерево поиска, ключом которого является пара переменных $\{x, y\}$, а значением, соответствующим ключу, — множество $S^{\{x,y\}}$. Такое представление гарантирует логарифмическое по числу узлов дерева время поиска ключа.

Опишем **вспомогательный алгоритм** $AddConstraint(R, C)$, который служит для добавления ограничения C в правильную систему R так, чтобы полученная система была правильной и равносильной системе $R \& C$.

1. Если $red(C)$ содержит две нулевые переменные, то, если это ограничение строгое, сообщается, что система после добавления стала несовместной, если же ограничение нестрогое, система не изменяется.

2. Если $red(C)$ содержит ровно одну ненулевую переменную x , то $red(C)$ добавляется в множество $R^{\{0,x\}}$. Очевидно, мощность любого такого множества не превзойдет r_0 в силу свойства (2) определения приведенного ограничения.

3. Пусть $red(C)$ содержит две ненулевые переменные x и y . Добавим $red(C)$ в множество $R^{\{x,y\}}$. Выберем меньшую (по отношению порядка \prec_V на множестве переменных) из переменных, для определенности x . Также для определенности будем считать, что коэффициент при x в $red(C)$ положительный. Выделим из $R^{\{x,y\}}$ подмножество ограничений с положительными коэффициентами при x , обозначим это подмножество $R_{+x}^{\{x,y\}}$.

Разберем случай, когда $R_{+x}^{\{x,y\}}$ состоит из трех ограничений. Упорядочим эти ограничения $C_i : a_i x + b_i y \prec_i 0$ ($i = 1, 2, 3$) так, что $b_1 a_2 \leq a_1 b_2$ и $b_2 a_3 \leq a_2 b_3$ (то есть $b_1/a_1 \leq b_2/a_2 \leq b_3/a_3$, но деление в реализации алгоритма не используется).

Если знак \prec_2 — нестрогий, или все \prec_i ($i = 1, 2, 3$) — строгие, то нетрудно видеть, что $C_1 \& C_3 \Rightarrow C_2$, и поэтому C_2 можно удалить и получить систему, равносильную системе $R \& C$.

Если \prec_1 есть знак \leq , $\prec_2 = <$, $\prec_3 = <$, то при $b_1 a_2 = a_1 b_2$ удаляем C_1 , а при $b_1 a_2 < a_1 b_2$ удаляем C_2 . Если знаки неравенств расположены в таком порядке: $<<<$, то действуем аналогично, удаляя, соответственно, C_3 или C_2 .

Если же порядок знаков неравенств таков: $\leq \leq \leq$, то оставляем в $R_{+x}^{\{x,y\}}$ все C_i ($i = 1, 2, 3$). Однако при добавлении четвертого ограничения в $R_{+x}^{\{x,y\}}$ обязательно возникнет по крайней мере один из разобранных выше случаев, который позволит удалить одно или два ограничения.

Если коэффициент при x в $red(C)$ отрицательный, то действуем аналогично.

Из построения алгоритма следует, что в полученной системе количество неравенств, содержащих любые две переменные, не превзойдет r_0 . Описание алгоритма *AddConstraint* завершено.

Теперь опишем **основной алгоритм** *Feasibility(S)*, определяющий, совместна ли система S .

1. Добавим в изначально пустую систему с помощью алгоритма *AddConstraint* все m ограничений системы S . В результате либо получим правильную систему S_0 , равносильную S , либо сделаем вывод о несовместности системы, если при добавлении некоторого ограничения алгоритм *AddConstraint* сообщил о несовместности.

2. На каждом из n следующих шагов будем исключать одну из основных переменных, получая правильные системы S_i ($i = 1, 2, \dots, n$), которые совместны или несовместны одновременно с исходной системой S .

На шаге i выбираем любую не исключенную ранее основную переменную y . В систему S_i переносим все ограничения системы S_{i-1} , в которые не входит переменная y . Для всех пар ограничений C_1 и C_2 системы S_{i-1} , содержащих переменную y , если определена композиция $C_1 \circ_y C_2$, добавляем эту композицию в S_i с помощью вызова *AddConstraint(S_i, C_1 \circ_y C_2)*.

Если на каком-либо шаге *AddConstraint* сообщил о несовместности, то алгоритм *Feasibility* выдает такой же ответ. Если некоторая система S_i оказалась пуста, то алгоритм *Feasibility* выдает ответ, что система совместна.

3. Наконец, если получена система S_n (она содержит лишь дополнительные переменные

ные), проверим, выполнена ли она при замене $\mathbf{0}$ на 0 и $\mathbf{1}$ на 1. Если да, то *Feasibility* выдает ответ, что система совместна, иначе — несовместна.

3. Оценки сложности алгоритма

Отметим, что в данной реализации используются только следующие операции над коэффициентами систем ("длинными" целыми числами): сложение, умножение, сравнение и нахождение абсолютной величины. Нахождение абсолютной величины, сложение и сравнение чисел длины l выполняется за время $O(l)$, а умножение — за $O(l^2)$.

Оценка длины чисел. Приведенные в [5] леммы 9 и 10 о верхней оценке длины чисел, используемых в алгоритме *Feasibility*, доказывают то, что длина любого ограничения системы S_i ($i = 1, 2, \dots, n$) не превосходит удвоенной суммы длин ограничений системы S_0 . Поэтому длина любого коэффициента системы S_i не превосходит $2L$, где L — сумма длин всех коэффициентов исходной системы. Поскольку результат умножения двух таких коэффициентов используется в сравнении (см. *AddConstraint*, п. 3), то в целом реализация алгоритма выполняет арифметические операции над числами длины не более $4L$.

Если под длиной чисел понимать их длину в двоичной кодировке, и для представления каждого числа в исходной системе используется не более k двоичных разрядов, то $L \leq 2km$, и длина чисел, используемых в работе алгоритма, не превосходит $8km$.

Оценка временной сложности. Для системы S_j ($i = 0, 1, \dots, n$) с не более чем n основными переменными алгоритм *AddConstraint* находит приведение добавляемого ограничения за время $O(L)$; затем выполняет поиск пары переменных в сбалансированном дереве с $O(n^2)$ узлами, на что затрачивает время $O(\log n^2) = O(\log n)$. Наконец, *AddConstraint* осуществляет удаление одного или двух избыточных ограничений из множества ограничений, мощность которого не превосходит r_0 ; на это уходит время $O(L^2)$. Таким образом, *AddConstraint* работает за время $O(\log n + L^2)$.

Алгоритм *Feasibility* при построении правильной системы S_0 (см. *Feasibility*, п. 1) вызывает *AddConstraint* m раз, затрачивая на это время $O(m(\log n + L^2))$.

Далее, на каждом из n шагов по исключению основной переменной из системы S_i (см. *Feasibility*, п. 2) строятся композиции пар ограничений, в которых фиксирована одна (исключаемая) переменная. В силу правильности системы S_i таких ограничений $O(n - i) \cdot r_0 = O(n)$, а таких пар ограничений $O(n^2)$. Композиция находится за время $O(L^2)$. Для каждой из $O(n^2)$ композиций ограничений вызывается *AddConstraint*. Значит, исключение всех основных переменных требует времени $O(n^3(\log n + L^2))$.

На последнем этапе (см. *Feasibility*, п. 3) проверяется выполнение правильной системы S_n с одними лишь дополнительными переменными, в такой системе $O(1)$ ограничений. На эту проверку уходит время $O(L)$.

Итак, реализация алгоритма *Feasibility* на Java имеет временную сложность $O(m \log n + mL^2 + n^3 \log n + n^3 L^2)$.

Оценка емкостной сложности. При работе алгоритма *Feasibility* требуется память только для хранения исходной системы S одновременно с S_0 , либо S_{i-1} одновременно с S_i ($i = 1, 2, \dots, n$). Длина системы S есть длина входа I . Система S_j ($j = 0, 1, \dots, n$) содержит $O((n-j)^2)$ ограничений длиной $O(L)$, и потому ее длина не превосходит $O(Ln^2)$. Таким образом, емкостная сложность алгоритма *Feasibility* составляет $O(\max\{I, Ln^2\})$.

Список литературы

- [1] Л.Г. Хачиян *Полиномиальные алгоритмы в линейном программировании*. - Журн. вычислит. математики и мат. физики, 1980, N1, с. 51-68.
- [2] N. Karmarkar *A new polynomial-time algorithm for linear programming*. - In: Proceedings of the Sixteenth Annual ACM Symposium on Theory of Computing. N.Y.: The Association for Computing Machinery, p. 302-311.
- [3] А. Схрейвер *Теория линейного и целочисленного программирования*. Т. 1. М.: Мир, 1991, 360 с.
- [4] N. Megiddo *Towards a genuinely polynomial algorithm for linear programming*. - SIAM J. Comput., 1983, N12, p. 347-353.
- [5] Д.В. Давыдок *О совместности систем двучленных линейных неравенств*. - В кн.: Н.К. Косовский, А.В. Тишков *Логика конечнозначных предикатов на основе неравенств*. СПб: Издательство С.-Петербург. университета, 2000, с. 246-268.
- [6] К. Арнольд, Дж. Гослинг, Д. Холмс *Язык программирования Java*. М.: Издательский дом "Вильямс", 2001, 624 с.
- [7] *The Java technology site*. <http://java.sun.com>

AN ALGORITHM FOR DETERMINING FEASIBILITY OF SYSTEMS OF TWO-TERM LINEAR INEQUALITIES

A.S. Gerasimov, N.K. Kossovskii

Saint-Petersburg State University, The Department of Mathematics and Mechanics, Saint-Petersburg

e-mail: gas-teach@yandex.ru, kosov@nk1022.spb.edu

Abstract. Davydok's algorithm for determining feasibility of systems of two-term linear inequalities is refined and implemented. A refined description of the algorithm is given; the description corresponds to its implementation in the Java programming language. Bounds on time and space complexity of the implementation as well as a bound on the length of all used numbers are obtained.

Key words: linear programming, two-term linear inequalities, systems, feasibility, genuinely polynomial algorithm, complexity, Java.

КОНТРПРИМЕРЫ В ТЕОРИИ СХОДИМОСТИ МЕТОДА ВНУТРЕННИХ ТОЧЕК

И.И. Дикин, О.М. Попова

Институт систем энергетики им. Л.А. Мелентьева СО РАН, Иркутск
e-mail: idikin@isem.sei.irk.ru, pom@isem.sei.irk.ru

Аннотация. Исследуются примеры однородных задач линейного программирования, целевые функции которых не ограничены снизу на допустимом множестве. Представлен способ определения стартовой точки. При этом метод внутренних точек вырабатывает последовательность векторов, сходящуюся к нулю.

Ключевые слова: линейное программирование, метод внутренних точек, сходимость, стартовая точка.

В работе приведены модификации примеров, исследуемых в статьях [1, 2].

Рассмотрим следующую задачу линейного программирования:

найти минимум

$$\sum_{j=1}^n c_j x_j \quad (1)$$

при условиях

$$\sum_{j=1}^n a_j x_j = b, \quad x_j \geq 0, \quad j = 1, \dots, n. \quad (2)$$

Для решения экстремальной проблемы (1), (2) применим итеративный алгоритм

$$x_j^{k+1} = x_j^k (1 + \lambda_k x_j^k \delta_j^k), \quad \delta_j^k = \delta_j(x^k) = a_j u_k - c_j, \quad j = 1, \dots, n,$$

$$u_k = u(x^k) = \sum_{j=1}^n c_j a_j (x_j^k)^2 / \sum_{j=1}^n a_j^2 (x_j^k)^2,$$

$$\lambda_k = \lambda(x^k) = \rho / \varphi_k, \quad 0 < \rho < 1,$$

$$\varphi_k = \max_{1 \leq j \leq n} (-x_j^k \delta_j^k).$$

Заметим, что вектор $x^0 > 0$ является допустимым.

Теперь будем исследовать частный случай задачи (1), (2). Примем

$$n = 4, \quad a = (1, -1, 1, -1), \quad b = 0,$$

$$c = (C, C, -1, -1), \quad C > 0.$$

Очевидно, что здесь существует такая допустимая точка x^* , что $(c, x^*) < 0$.

Имеем

$$x_1 - x_2 + x_3 - x_4 = 0, \quad x_1^2 \delta_1(x) - x_2^2 \delta_2(x) + x_3^2 \delta_3(x) - x_4^2 \delta_4(x) = 0. \quad (3)$$

Предположим, что

$$(x_1^1, x_2^1, x_3^1, x_4^1) = \mu (x_2^0, x_1^0, x_4^0, x_3^0). \quad (4)$$

Тогда $u_1 = -u_0$, $\lambda_1 = \lambda_0/\mu$,

$$x^2 = \mu^2 x^0. \quad (5)$$

На основании (5) получаем, что

$$0 < \mu < 1, \quad x^k \rightarrow 0, .$$

В статьях [1, 2] рассматриваются однородные задачи линейного программирования, ограничения которых имеют вид $Ax \geq 0$, а целевые функции не ограничены снизу на допустимом множестве. При изучении приведенных в этих статьях примеров доказано существование такого вектора x^0 , что справедливо равенство (5). При этом в работе [1] с большой степенью точности представлена стартовая точка x^0 , но не предлагается способ ее определения.

Пусть

$$C = 20, \quad \rho = 0.995.$$

Определим такое начальное приближение x^0 , что имеет место равенство (4). Полагаем

$$x_4 = 1, \quad x_3 = \frac{\mu}{1-\rho}, \quad x_2 = x_1 + x_3 - 1, \quad \lambda(x) = \frac{\rho}{x_3(-u(x) - 1)}$$

и введем функции

$$f(x_1, \mu) = x_1 + \lambda(x) x_1^2 (u(x) - C) - \mu x_2,$$

$$g(x_1, \mu) = x_2 + \lambda(x) x_2^2 (-u(x) - C) - \mu x_1,$$

$$h(x_1, \mu) = 1 + \lambda(x) (-u(x) + 1) - \mu x_3.$$

Используя компьютерную систему Maple для решения системы нелинейных уравнений

$$f(x_1, \mu) = 0, \quad h(x_1, \mu) = 0, \quad (6)$$

находим числа

$$x_1^0 = 2.0395413790632040669698007549870,$$

$$\mu' = 0.073532324247437522402946355005742.$$

Из равенств (3), (6) следует, что $g(x_1^0, \mu') \approx 0$.

Таким образом, при $\rho = 0.995$ можно сделать так, что соотношение (4) выполняется с требуемой точностью. Когда $C = 50$, $\rho = 0.9945$, получаем аналогичный результат.

Геометрический способ решения исследуемой системы уравнений представлен на рис. 1.

Теперь приведем несколько искусственный пример. К ограничениям представленной выше задачи добавим условия

$$y_j = x_1, \quad z_j = x_2, \quad j = 1, \dots, L - 1.$$

Причем, $L = 10^{14}$, $C = (2 + d)/d$, $d = 10^{-12}$. Новая экстремальная проблема является модификацией контрпримера, исследуемого в статье [2].

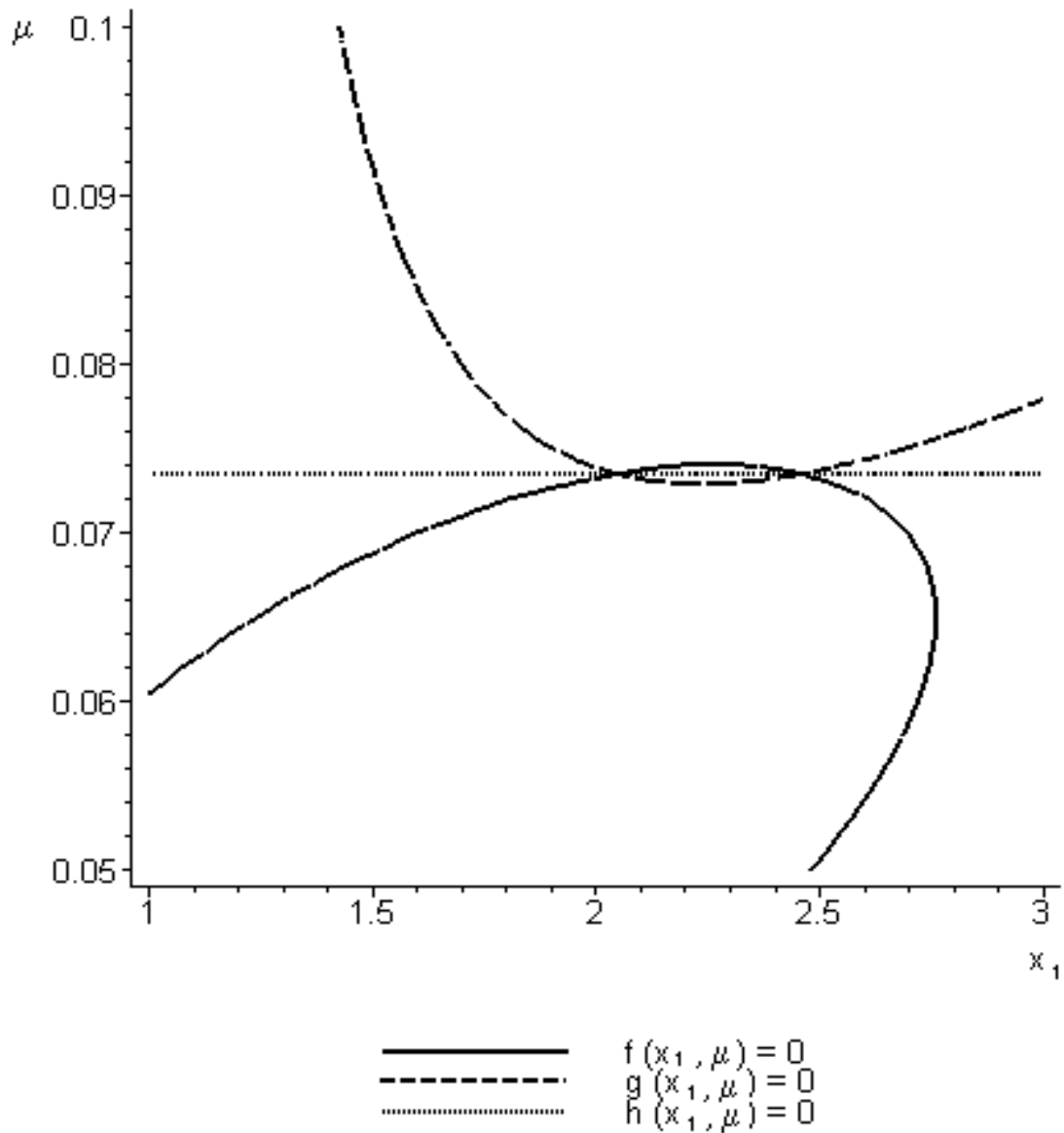


Рис. 1. Графики уравнений.

Получаем соотношения

$$x_1^2(v - C) - x_2^2(v + C) + Lx_3^2(v + 1) + Lx_4^2(v - 1) = 0,$$

$$v(x) = \frac{Cx_1^2 - Cx_2^2 - Lx_3^2 + Lx_4^2}{x_1^2 + x_2^2 + Lx_3^2 + Lx_4^2}.$$

Как и раньше, примем

$$x_4 = 1, \quad x_3 = \frac{\mu}{1 - \rho}, \quad x_2 = x_1 + x_3 - 1, \quad \lambda(x) = \frac{\rho}{x_3(-v(x) - 1)}$$

и рассмотрим функции

$$\bar{f}(x_1, \mu) = x_1 + \lambda(x) x_1^2 (v(x) - C)/L - \mu x_2,$$

$$\begin{aligned}\bar{g}(x_1, \mu) &= x_2 + \lambda(x) x_2^2 (-v(x) - C)/L - \mu x_1, \\ \bar{h}(x_1, \mu) &= 1 + \lambda(x) (-v(x) + 1) - \mu x_3.\end{aligned}$$

Решив систему

$$\bar{f}(x_1, \mu) = 0, \quad \bar{h}(x_1, \mu) = 0$$

при $\rho = 0.92$, будем иметь

$$\begin{aligned}x_1^0 &= 395.18043396193417836199091003601, \\ \mu'' &= 0.33332625586548303561660382241031.\end{aligned}$$

Отметим, что при $0 < \rho \leq 2/3$ опубликованы интересные результаты по исследованию глобальной сходимости последовательностей векторов двойственных и прямых переменных при решении общей задачи линейного программирования [3 – 7].

По мнению авторов, представленные в данной работе результаты имеют только теоретическое значение. При решении прикладных задач можно спокойно принимать $\rho > 2/3$, например, $\rho = 0.9$.

Список литературы

- [1] Mascarenhas W.F. *The affine scaling algorithm fails for stepsize 0.999*. — Siam Journal on Optimization — 1997. — Vol. 7, No. 1. — P. 34–46.
- [2] Terlaky T., Tsuchiya T. *A note on Mascarenhas' counterexample about convergence of the affine scaling algorithm fails*. — Applied Mathematics and Optimization — 1999. — Vol. 40. — P. 287–314.
- [3] Дикин И.И. *Определение внутренней точки одной системы линейных неравенств*. — Кибернетика и системный анализ. — 1992. — N 1. — С. 67–74.
- [4] Saigal R. *Linear programming: a modern integrated analysis*. — Dordrecht: Kluwer Academic Publishers, 1995. — 342 p.
- [5] Tsuchiya T. *Affine scaling algorithm // Interior point methods of mathematical programming / Ed. by T. Terlaky*. — Dordrecht: Kluwer Academic Publishers, 1996. — P. 35–82.
- [6] Дикин И.И., Попова О.М. *Исследование и ускорение сходимости алгоритмов метода внутренних точек: Решение оптимизационных задач термодинамики*. — Новосибирск: Наука. Сиб. предприятие РАН, 1997. — 70 с.
- [7] Дикин И.И. *Определение допустимых и оптимальных решений методом внутренних точек*. — Новосибирск: Наука. Сиб. предприятие РАН, 1998. — 110 с.

THE COUNTEREXAMPLES IN THE THEORY OF AFFINE SCALING METHOD CONVERGENCE

I.I.Dikin, O.M.Popova

Energy Systems Institute SB RAS, Irkutsk
e-mail: idikin@isem.sei.irk.ru, pom@isem.sei.irk.ru

Abstract. The examples of the homogeneous problems of linear programming are investigated. The objective function of the problem is not bounded below on a feasible set. The technique of determining a starting point is presented. In doing so the affine scaling method generates a sequence of vectors that converge to zero.

Key words: linear programming, affine scaling method, convergence, starting point.

МАТРИЧНАЯ КОРРЕКЦИЯ НЕСОБСТВЕННЫХ ЗАДАЧ ЛИНЕЙНОГО ПРОГРАММИРОВАНИЯ ПО МИНИМУМУ ЕВКЛИДОВОЙ НОРМЫ С ПРОИЗВОЛЬНЫМИ ВЕСАМИ И ФИКСИРОВАННЫМИ ЭЛЕМЕНТАМИ¹

В.И. Ерохин

Борисоглебский государственный педагогический институт
e-mail: erohin_v_i@mail.ru

Аннотация. Задача многопараметрической коррекции несобственной задачи линейного программирования в общей форме с произвольными фиксированными коэффициентами по минимуму взвешенной с произвольными положительными весами евклидовой нормы сводится к задаче минимизации дифференцируемой функции при условии неотрицательности части ее переменных. Приводятся формулы для частных производных указанной функции, что делает возможным применение для ее минимизации градиентных методов.

Ключевые слова: несобственные задачи линейного программирования, многопараметрическая коррекция

Введение

Пусть

$$L : \begin{cases} c_1^T x_1 + c_2^T x_2 \rightarrow \inf, \\ A_{11} x_1 + A_{12} x_2 \leq b_1, \\ A_{21} x_1 + A_{22} x_2 = b_2, \\ x_1 \geq 0 \end{cases} \quad \text{and} \quad L^* : \begin{cases} b_1^T u_1 + b_2^T u_2 \rightarrow \sup, \\ u_1^T A_{11} + u_2^T A_{21} \geq c_1, \\ u_1^T A_{12} + u_2^T A_{22} = c_2, \\ u_1 \geq 0, \end{cases}$$

где $A_{11} \in \mathbb{R}^{m_1 \times n_1}$, $A_{12} \in \mathbb{R}^{m_1 \times n_2}$, $A_{21} \in \mathbb{R}^{m_2 \times n_1}$, $A_{22} \in \mathbb{R}^{m_2 \times n_2}$, $x_1, c_1 \in \mathbb{R}^{n_1}$, $x_2, c_2 \in \mathbb{R}^{n_2}$, $b_1 \in \mathbb{R}^{m_1}$, $b_2 \in \mathbb{R}^{m_2}$, $u_1 \in \mathbb{R}^{m_1}$, $u_2 \in \mathbb{R}^{m_2}$ - пара взаимодвойственных задач линейного программирования (ЛП), записанных в общей форме, причем $m_1 + m_2 = m$, $n_1 + n_2 = n$,

$$\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} = A = (a_{ij}), \quad \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} = b = (b_i), \quad \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} = c = (c_j), \quad \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = x = (x_j), \quad \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = u = (u_i).$$

Обозначим допустимое множество задачи L как $X_L(A, b) = \{x\}$, а допустимое множество задачи L^* как $U_{L^*}(A, c) = \{u\}$. Задачи L и L^* предполагаются несобственными, поэтому первоначально хотя бы одно из множеств $X_L(A, b)$ или $U_{L^*}(A, c)$ является пустым [1].

Пусть

$$H = \begin{bmatrix} H_{11} & H_{12} & -h_1 \\ H_{21} & H_{22} & -h_2 \end{bmatrix} = [H \quad -h] = (h_{ij}) \in \mathbb{R}^{m \times (n+1)} \quad -$$

некоторая матрица коррекции (отдельные элементы которой возможно являются нулевыми), т.е., такая матрица, что

$$X_L(A + H, b + h) \neq \emptyset, U_{L^*}(A + H, c) \neq \emptyset. \quad (1)$$

¹Работа выполнена при поддержке Федерального агентства по образованию (заказ N 1.1.05)

Рассмотрим также матрицу

$$W = (w_{ij}) \in \mathbb{R}^{m \times (n+1)},$$

состоящую из произвольных положительных элементов. Пусть $\|\cdot\|_E$ - евклидова матричная или (в зависимости от контекста) векторная норма, а знак "o" означает поэлементное перемножение матриц (произведение Адамара). Тогда основная задача, являющаяся предметом нашего исследования, может быть сформулирована как

$$Z_1 : \|W \circ H\|_E \rightarrow \inf_{\substack{X_L(A+H, b+h) \neq \emptyset, \\ U_{L^*}(A+H, c) \neq \emptyset}} (= \zeta_1).$$

Указанная задача является дальнейшим обобщением задач матричной коррекции несовместных систем линейных алгебраических уравнений, неравенств и несобственных задач ЛП по минимуму евклидовой нормы, исследованных в свое время А.А. Ватолиным [1] (§§12-13), [2] и позже В.А. Гореликом [3].

Будем предполагать, что в задаче Z_1 нет фиксированных элементов, т.е., в матрице A и векторе b нет элементов, коррекция которых запрещена. Данное предположение не ограничивает общности рассуждений, но на начальном этапе позволяет упростить некоторые обозначения и выкладки.

В дополнение к задаче Z_1 рассмотрим и задачу Z'_1 , которая будет отличаться от Z_1 наличием произвольных фиксированных элементов в матрице A и некоторых (но не всех) произвольных фиксированных элементов в векторе b .

1. Анализ задачи Z_1

Пусть

$$r_1(x_1, x_2, y) = \begin{bmatrix} b_1 - y - A_{11}x_1 - A_{12}x_2 \\ b_2 - A_{21}x_1 - A_{22}x_2 \end{bmatrix}, r_2(u_1, u_2, z) = \begin{bmatrix} c_1 + z - u_1^T A_{11} - u_2^T A_{21} \\ c_2 - u_1^T A_{12} - u_2^T A_{22} \end{bmatrix}, \quad (2)$$

где $y \in \mathbb{R}^{m_1}$, $z \in \mathbb{R}^{n_1}$, $y, z \geq 0$. Тогда условие (1) можно записать в виде

$$Hx - h = r_1(x_1, x_2, y), u^T H = r_2(u_1, u_2, z). \quad (3)$$

Рассмотрим векторы

$$\omega = [w_{11}, \dots, w_{1,n+1}, w_{21}, \dots, w_{2,n+1}, \dots, w_{m1}, \dots, w_{m,n+1}]^T \in \mathbb{R}^{m(n+1)},$$

$$\hbar = [h_{11}, \dots, h_{1,n+1}, h_{21}, \dots, h_{2,n+1}, \dots, h_{m1}, \dots, h_{m,n+1}]^T \in \mathbb{R}^{m(n+1)}, \quad (4)$$

$$g = \begin{bmatrix} x \\ u \end{bmatrix}, \quad (5)$$

$$r(g, y, z) = \begin{bmatrix} r_1(x_1, x_2, y) \\ r_2(u_1, u_2, z) \end{bmatrix}, \quad (6)$$

и матрицы (I_n - единичная матрица порядка n)

$$X(x) = \begin{bmatrix} [x^T \ 1] & 0 & \dots & 0 \\ 0 & [x^T \ 1] & \dots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & [x^T \ 1] \end{bmatrix} \in \mathbb{R}^{m \times (m(n+1))},$$

$$\begin{aligned}\tilde{I}_{n+1} &= \begin{bmatrix} I_n & 0 \\ 0 & 0 \end{bmatrix} \in \mathbb{R}^{(n+1) \times (n+1)}, \\ \mathbf{U}(u) &= [u_1 \cdot \tilde{I}_{n+1} \quad \cdots \quad u_m \cdot \tilde{I}_{n+1}] \in \mathbb{R}^{n \times (m(n+1))} \\ \tilde{X}(x) &= \mathbf{X}(x) \cdot (\text{diag}(\omega))^{-1}, \tilde{U}(u) = \mathbf{U}(u) \cdot (\text{diag}(\omega))^{-1}, \\ G(g) &= \begin{bmatrix} \tilde{X}(x) \\ \tilde{U}(u) \end{bmatrix}.\end{aligned}$$

Поскольку вектор \tilde{h} состоит из тех же самых элементов, что и матрица \mathbf{H} , евклидовы нормы указанных объектов совпадают. А в силу того, что вектор ω составлен из тех же самых элементов, что и матрица \mathbf{W} , справедливо условие

$$\|\text{diag}(\omega) \tilde{h}\|_E = \|\mathbf{W} \circ \mathbf{H}\|_E. \quad (7)$$

Определенные выше векторы и матрицы позволяют записать условие (3) в виде

$$G(g) \cdot (\text{diag}(\omega) \cdot \tilde{h}) = G(g) \cdot \tilde{h} = r(g, y, z). \quad (8)$$

Заметим, что при произвольном x и произвольном $u \neq 0$ система линейных алгебраических уравнений (8) совместна. Действительно, с одной стороны, число строк матрицы $G(g)$ $n + m$ меньше числа ее столбцов $n + nm + n^2$, т.е. система (8) является недоопределенной. В то же время, строки матрицы $\tilde{X}(x)$ линейно независимы, строки матрицы $\tilde{U}(u)$ также линейно независимы и ни одна из строк одной матрицы не может быть представлена в виде линейной комбинации строк другой, что следует из их структуры. Таким образом, матрица $G(g)$ является матрицей полного строчного ранга, откуда и следует совместность системы (8).

Найдем нормальное (т.е., обладающего минимальной евклидовой нормой) решение (8) относительно вектора \tilde{h} . Как известно (см., например, [4]), для любой совместной системы оно существует, единственно и может быть выражено формулой

$$\hat{\tilde{h}}(g, y, z) = G^+(g) \cdot r(g, y, z), \quad (9)$$

где "+" означает псевдообращение. Пусть

$$\hat{h}(g, y, z) = (\text{diag}(\omega))^{-1} \hat{\tilde{h}}(g, y, z). \quad (10)$$

Тогда, в силу (7),

$$\left\| \hat{h}(g, y, z) \right\|_E = \left\| \text{diag}(\omega) \hat{h}(g, y, z) \right\|_E = \left\| \mathbf{W} \circ \hat{\mathbf{H}} \right\|_E, \quad (11)$$

где матрица $\hat{\mathbf{H}}$ составлена из элементов вектора \hat{h} по правилу, задаваемому формулой (4).

Таким образом, при фиксированных x, u, y, z , матрица $\hat{\mathbf{H}}$, корректирующая несобственную задачу L и имеющая минимальную взвешенную евклидову норму, определяется формулами (9), (10) и (4).

В соответствии с (11) и (9) введем в рассмотрение функцию

$$f(g, y, z) = \left\| \mathbf{W} \circ \hat{\mathbf{H}} \right\|_E^2 = r^T(g, y, z) \cdot G^{+T}(g) G^+(g) \cdot r(g, y, z)$$

и задачу ее минимизации

$$Z_2 : f(g, y, z) \rightarrow \inf_{g, y, z | x_1, u_1, y, z \geq 0} (= \zeta_2).$$

Тогда, в силу приведенных выше рассуждений, оказывается справедливой следующая теорема, сводящая задачу Z_1 к задаче Z_2 :

Теорема 1. Пусть для пары несобственных задач линейного программирования L, L^* рассматривается задача взвешенной матричной коррекции Z_1 . Тогда

1) $\zeta_1^2 = \zeta_2$.

2) Нижние грани целевых функций Z_1 и Z_2 достигаются или не достигаются одновременно.

3) Если $g^* = \begin{bmatrix} x^* \\ u^* \end{bmatrix}, y^*, z^*$ - оптимальное решение задачи Z_2 , то матрица $H^* = \begin{bmatrix} H^* & -h^* \end{bmatrix}$, построенная по вектору $\hat{h}(g^*, y^*, z^*)$ с учетом формулы (4), является оптимальным решением задачи Z_1 , причем $x^* \in X_L(A + H^*, b + h^*)$ и $u^* \in U_{L^*}(A + H^*)$.

2. Методы решения задачи Z_1

Рассмотренная в предыдущем параграфе теорема 1 сводит решение задачи Z_1 к задаче условной $n+m+n_1+m_1$ -мерной минимизации Z_2 . Теперь мы рассмотрим возможность использования для решения задачи Z_2 градиентных методов, а точнее, предложим формулы для вычисления градиента функции $f(g, y, z)$. Указанные формулы открывают возможность использования таких методов, как метод спроектированного градиента, метод возможных направлений или метод отсекающих плоскостей. В то же время очевидно, что сами формулы еще не являются законченным результатом в том смысле, что они не определяют всех деталей возможных вычислительных алгоритмов, проработка которых вполне может служить темой отдельного исследования.

Пусть

$$D(g) = G^{+T}(g) G^+(g). \quad (12)$$

С учетом (12)

$$f(g, y, z) = r^T(g, y, z) \cdot D(g) \cdot r(g, y, z). \quad (13)$$

С учетом (13)

$$\text{grad} f(g, y, z) = \begin{bmatrix} \frac{\partial f(g, y, z)}{\partial g} \in \mathbb{R}^{m+n} \\ \frac{\partial f(g, y, z)}{\partial y} \in \mathbb{R}^{n_1} \\ \frac{\partial f(g, y, z)}{\partial z} \in \mathbb{R}^{m_1} \end{bmatrix},$$

где

$$\begin{aligned} \frac{\partial f(g, y, z)}{\partial g} &= \frac{\partial}{\partial g} r^T(g, y, z) \cdot D(g) \cdot r(g, y, z) + r^T(g, y, z) \cdot \frac{\partial}{\partial g} D(g) \cdot r(g, y, z) + \\ &\quad + r^T(g, y, z) \cdot D(g) \cdot \frac{\partial}{\partial g} r(g, y, z), \\ \frac{\partial f(g, y, z)}{\partial y} &= \frac{\partial}{\partial y} r^T(g, y, z) \cdot D(g) \cdot r(g, y, z) + r^T(g, y, z) \cdot D(g) \cdot \frac{\partial}{\partial y} r(g, y, z), \\ \frac{\partial f(g, y, z)}{\partial z} &= \frac{\partial}{\partial z} r^T(g, y, z) \cdot D(g) \cdot r(g, y, z) + r^T(g, y, z) \cdot D(g) \cdot \frac{\partial}{\partial z} r(g, y, z). \end{aligned} \quad (14)$$

Дифференцирование вектора $r(g, y, z)$ не вызывает затруднений. В соответствии с (2), (5) и (6),

$$\frac{\partial}{\partial x_j} r(g, y, z) = \begin{bmatrix} -a_{1j} \\ \vdots \\ -a_{mj} \\ 0 \in \mathbb{R}^n \end{bmatrix}, j = 1, 2, \dots, n, \frac{\partial}{\partial u_i} r(g, y, z) = \begin{bmatrix} 0 \in \mathbb{R}^m \\ -a_{i1} \\ \vdots \\ -a_{in} \end{bmatrix}, j = 1, 2, \dots, n,$$

$$\frac{\partial}{\partial y_k} r(g, y, z) = \begin{bmatrix} -e_k(m_1) \\ 0 \in \mathbb{R}^{m_2+n} \end{bmatrix}, k=1, 2, \dots, m_1, \frac{\partial}{\partial z_\ell} r(g, y, z) = \begin{bmatrix} 0 \in \mathbb{R}^m \\ e_\ell(n_1) \\ 0 \in \mathbb{R}^{n_2} \end{bmatrix}, \ell=1, 2, \dots, n_1,$$

где $e_k(m_1)$ - столбец единичной матрицы порядка m_1 с номером k , $e_\ell(n_1)$ - столбец единичной матрицы порядка n_1 с номером ℓ .

Дифференцирование матрицы $D(g)$, представляющей произведение псевдообратных матриц, в общем случае можно выполнить, следуя работе [5]. Но в нашем случае это не обязательно, поскольку

$$D(g) = (G(g) \cdot G^T(g))^{-1} = Q^{-1}(g). \quad (15)$$

Заметим, что формула (15) справедлива только для матриц полного строчного ранга (см., например, [4]), но это и есть рассматриваемый нами случай. С учетом (15) имеем

$$\frac{\partial}{\partial g_j} D(g) = -D(g) \cdot \frac{\partial}{\partial g_j} Q(g) \cdot D(g) = -D(g) \cdot \left(\frac{\partial}{\partial g_j} G(g) \cdot G^T(g) + G(g) \cdot \left(\frac{\partial}{\partial g_j} G(g) \right)^T \right) \cdot D(g). \quad (16)$$

Дифференцирование матрицы $G(g)$ несложно и мы опустим соответствующие формулы из-за ограниченного объема статьи.

3. Метод решения задачи Z'_1

Пусть необходимо запретить коррекцию коэффициента a_{pq} . Обнулیم столбец с номером $(p-1) \cdot (n+1) + q$ матрицы $G(g)$ и для получившейся матрицы $\bar{G}(g)$ повторим приведенные выше рассуждения. Если необходимо зафиксировать коэффициент b_i , следует обнулить столбец с номером $(n+1) \cdot i$, однако при этом матрица $\bar{G}(g)$ может выйти из класса матриц полного строчного ранга, что, во-первых, сделает невозможным использование формулы (15), а во-вторых, что более серьезно, может привести к несовместности системы (9). Таким образом, этот случай требует дополнительного исследования.

4. Возможные редукции задач Z_1 и Z'_1

Несложно заметить, что рассмотренный в работе подход к коррекции несобственных задач ЛП, заданных в общей форме, применим (как для частных случаев) к коррекции задач ЛП, заданных в других формах - основной, стандартной и канонической. Также, в качестве частных случаев, могут быть рассмотрены задачи матричной коррекции несовместных систем линейных ограничений, содержащих уравнения и неравенства, и просто несовместные системы линейных алгебраических уравнений. При этом расчетные формулы значительно упрощаются. Например, при $g \equiv x$, матрица $D(g)$ оказывается диагональной, что позволяет из (14) и (16) аналитически получить явные выражения для частных производных $f(g, y, z)$ как первого, так и второго порядка [6], открывающие путь к использованию метода Ньютона.

Список литературы

- [1] И.И. Еремин, В.Д. Мазуров, Н.Н. Астафьев *Несобственные задачи линейного и выпуклого программирования*. М.: Наука, 1983, 336 с.
- [2] А.А. Ватолин *Аппроксимация несобственных задач линейного программирования по критерию евклидовой нормы*. ЖВМ и МФ, 1984, Т. 24, N 12, с. 1907–1908.
- [3] В.А. Горелик *Матричная коррекция задачи линейного программирования с несовместной системой ограничений*. ЖВМ и МФ, 2001, Т. 41, N 11, с. 1697–1705.
- [4] Ф.Р. Гантмахер *Теория матриц*. М.: Наука, 1988, 552 с.
- [5] G.H. Golub, V. Pereyra *The differentiation of pseudo-inverses and nonlinear least squares problems whose variables separate*. SIAM J. Numer. Anal., 1973, Vol. 10, No. 2, p. 413–432.
- [6] В.А. Горелик, В.И. Ерохин *Оптимальная матричная коррекция несовместных систем линейных алгебраических уравнений по минимуму евклидовой нормы*. М.: ВЦ РАН, 2004, 192 с.

MATRIX CORRECTION OF THE IMPROPER LINEAR PROGRAMMING PROBLEMS ON THE MINIMUM OF EUCLIDEAN NORM WITH THE ARBITRARY WEIGHTS AND THE FIXED ELEMENTS

V.I. Erokhin

State Teachers Training Institute, Borisoglebsk

e-mail: erohin_v_i@mail.ru

Abstract. The problem multiparameter correction of a problem of a linear programming in the common shape with the arbitrary fixed coefficients on a minimum the Euclidean norm weighed with the arbitrary positive weights it is reduced to a problem of minimization of a differentiable function under condition of a nonnegativity of a part of its variables. Formulas for partial derivatives of indicated function are reduced, that makes possible to execute gradient methods for its minimization.

Key words: improper linear programming problems, multiparameter correction

МЕТОД ВНУТРЕННЕЙ ТОЧКИ С НАИСКОРЕЙШИМ СПУСКОМ ДЛЯ ЛИНЕЙНОЙ ЗАДАЧИ ДОПОЛНИТЕЛЬНОСТИ¹

В.Г.Жадан, М.В. Втюрина

Вычислительный центр им. А.А.Дородницына РАН, Москва

e-mail: zhadan@ccas.ru

Московский физико-технический институт (государственный университет)

e-mail: vturina@scph.mipt.ru

Аннотация. Рассматривается линейная задача дополнителности. Для ее решения предлагается метод внутренней точки, в котором для выбора шага используется наискорейший спуск. Данный метод является обобщением барьерно-проективного метода, разработанного ранее для задач линейного программирования. Доказывается, что метод обладает конечной локальной и нелокальной сходимостью.

Ключевые слова: линейная задача дополнителности, метод внутренней точки, барьерно-проективный метод, наискорейший спуск.

Введение

Линейная задача дополнителности (ЛЗД) является достаточно общей постановкой, к решению которой сводятся многие оптимизационные и равновесные задачи [1], [2]. Имеется несколько подходов к решению ЛЗД, среди которых одним из основных является подход, опирающийся на идеологию методов внутренней точки. Методами данного типа являются, например, [3], [4]. В настоящем сообщении рассматривается барьерно-проективный метод, также относящийся к классу методов внутренней точки. Непрерывный и дискретный варианты этого метода были предложены первоначально в [5]. Метод строился путем применения барьерно-проективного метода [6] к решению задачи квадратичного программирования, порождаемой ЛЗД. Предлагаемый здесь вариант метода отличается от предыдущих использованием наискорейшего спуска для выбора шага.

В барьерно-проективном методе с наискорейшим спуском все точки в ходе итеративного процесса являются допустимыми. Это позволяет выбирать шаги из условия минимизации билинейной целевой функции при условии сохранения неотрицательности всех переменных. Однако в этом случае переменные могут принадлежать границам неотрицательных ортантов. Основные формулы барьерно-проективного метода из-за наличия эффекта "прилипания" становятся непригодными для расчетов, так как появляются дополнительные стационарные точки. Показывается, что при выполнении специального условия невырожденности этими стационарными точками могут быть лишь вершины допустимого множества. Чтобы устранить дополнительные стационарные точки предлагается модифицировать правые части рекуррентных соотношений, описывающих метод. А именно, предлагается правые части вычислять не непосредственно в угловых точках, а в других точках, сдвинутых по части компонент. Конкретный выбор изменяемых компонент зависит от соответствующих компонент градиентов функции Лагранжа.

¹Работа выполнена при содействии Российского фонда фундаментальных исследований (проект 03-01-00464) и гранта ведущих научных школ (НШ-1737.2003.1)

1. Постановка задачи

ЛЗД состоит в нахождении двух векторов x и y , удовлетворяющих следующей системе равенств и неравенств

$$\begin{aligned} x &\geq 0_n, & y &\geq 0_n, \\ y &= Mx + q, \\ 0 &= y^T x, \end{aligned} \quad (1)$$

где 0_n – нулевой n -мерный вектор. Предполагается, что M – положительно определенная (не обязательно симметричная) матрица порядка n и что задача является неоднородной, т.е. $q \neq 0_n$. Пусть $[x_*, y_*]$ – решение задачи (1). В силу положительной определенности матрицы M для любого $q \in R^n$ решение задачи (1) существует и единственно [1].

Пара $[x, y] \in R^{2n}$ называется внутренней (строго внутренней), если $x \geq 0_n, y \geq 0_n$ ($x > 0_n, y > 0_n$). Внутренняя пара $[x, y] \in R^{2n}$ называется допустимой, если $Mx - y + q = 0_n$. Ниже множество допустимых пар обозначается через \mathcal{Z}_+ .

Пусть J – множество индексов от 1 до n . Для внутренней пары $[x, y]$ введем четыре подмножества множества J (некоторые из них могут оказаться пустыми):

$$\begin{aligned} J_P(x, y) &= \{i \in J : x^i > 0, y^i > 0\}, \\ J_B(x, y) &= \{i \in J : x^i > 0, y^i = 0\}, \\ J_N(x, y) &= \{i \in J : x^i = 0, y^i > 0\}, \\ J_Z(x, y) &= \{i \in J : x^i = 0, y^i = 0\}. \end{aligned} \quad (2)$$

Для определенности везде ниже считаем, что в рассматриваемой паре $[x, y]$ они идут последовательно одно за другим в порядке, указанном в (2). Матрица M в соответствии с (2) делится на блоки:

$$M = \begin{bmatrix} M_{PP} & M_{PB} & M_{PN} & M_{PZ} \\ M_{BP} & M_{BB} & M_{BN} & M_{BZ} \\ M_{NP} & M_{NB} & M_{NN} & M_{NZ} \\ M_{ZP} & M_{ZB} & M_{ZN} & M_{ZZ} \end{bmatrix}.$$

Аналогичное обозначение будем использовать для разбиения любых n -мерных векторов на подвекторы, например, x^P – часть вектора x , состоящая из компонент, индексы которых принадлежат множеству $J_P(x, y)$.

Определение 1. Внутренняя пара $[x, y] \in R_+^{2n}$ называется регулярной, если $J_Z(x, y) = \emptyset$. В противном случае пара $[x, y]$ называется нерегулярной.

Введем в рассмотрение подматрицу $P(x, y)$ матрицы M и вектор $p(x, y)$, положив

$$P(x, y) = \begin{bmatrix} M_{BP} & M_{BB} \\ M_{ZP} & M_{ZB} \end{bmatrix}, \quad p(x, y) = \begin{bmatrix} y^P + M_{PP}^T x^P \\ M_{PB}^T x^P \end{bmatrix}.$$

Через $\mathcal{L}(x, y)$ обозначим пространство строк матрицы $P(x, y)$.

Определение 2. Допустимая пара $[x, y] \in \mathcal{Z}_+$ называется слабо невырожденной, если матрица $P(x, y)$ имеет полный ранг, равный $|J_B(x, y)| + |J_Z(x, y)|$. Слабо невырожденная пара называется невырожденной, если вектор $p(x, y)$ принадлежит \mathcal{L} в том и только том случае, когда $|J_Z(x, y)| = |J_P(x, y)|$.

Пусть $D(z)$ — диагональная матрица с вектором z на диагонали. Обозначим через $G(x, y)$ матрицу $G(x, y) = MD(x)M^T + D(y)$. Для слабо невырожденных пар матрица является неособой.

Если допустимая пара регулярна, то она обязательно является слабо невырожденной парой. Задачу (1) назовем *слабо невырожденной*, если все допустимые пары слабо невырожденны. Слабо невырожденную задачу, у которой все пары невырожденны, назовем *невырожденной*. У такой задачи решение $[x_*, y_*]$ является регулярной парой, в которой выполняется неравенство $x_* + y_* > 0_n$.

2. Барьерно-проективный метод

Пусть угловые скобки обозначают евклидово скалярное произведение. Введем функцию Лагранжа $L(x, y, u) = \langle x, y \rangle + \langle u, Mx - y + q \rangle$ и определим функцию $u(x, y)$ из условия

$$MD(x)L_x(x, y, u) = D(y)L_y(x, y, u), \quad (3)$$

где через $L_x(x, y, u) = y + M^T u$ и $L_y(x, y, u) = x - u$ обозначены производные функции $L(x, y, u)$ по x и y соответственно. Для слабо невырожденной задачи для любой допустимой пары получаем из (3)

$$u(x, y) = G^{-1}(x, y)(I_n - M)D(x)y.$$

Положим далее

$$F^{(1)}(x, y) = D(x)[y + M^T u(x, y)], \quad F^{(2)}(x, y) = D(y)[x - u(x, y)]. \quad (4)$$

Те допустимые пары $[x, y]$, в которых $F^{(1)}(x, y) = F^{(2)}(x, y) = 0_n$, называются *особыми*. В противном случае они называются *неособыми*.

Барьерно-проективный метод описывается следующими рекуррентными соотношениями:

$$x_{k+1} = x_k - \alpha_k \Delta x_k, \quad y_{k+1} = y_k - \alpha_k \Delta y_k, \quad (5)$$

где

$$\Delta x_k = F_0^{(1)}(x_k, y_k), \quad \Delta y_k = F_0^{(2)}(x_k, y_k). \quad (6)$$

Если начальная пара $[x_0, y_0]$ допустима, то при выборе шага α_k на каждой k -й итерации из условия, чтобы новые точки x_{k+1}, y_{k+1} не покидали ортанта R_+^n , получаем, что все последующие пары также оказываются допустимыми. Стационарными точками процесса (5) являются лишь особые пары. Имеет место следующее утверждение.

Лемма 1. Пусть задача (2) невырожденна. Тогда для того, чтобы пара $[x, y]$ была особой необходимо и достаточно, чтобы она была угловой точкой множества \mathcal{Z}_+ .

Введем функцию $V(x, y) = \langle x, y \rangle$ и рассмотрим ее поведение вдоль обратных направлений Δx_k и Δy_k , выпущенных из произвольной допустимой пары $[x_k, y_k]$. Взяв $\alpha \geq 0$, получаем

$$\begin{aligned} \phi(\alpha) &= V(x_k - \alpha \Delta x_k, y_k - \alpha \Delta y_k) = \\ &= V(x_k, y_k) - c_1(x_k, y_k)\alpha + c_2(x_k, y_k)\alpha^2, \end{aligned} \quad (7)$$

где $c_1(x, y) = \langle \Delta x, y \rangle + \langle \Delta y, x \rangle$, $c_2(x, y) = \langle \Delta x, \Delta y \rangle$.

Как было отмечено, для невырожденной ЛЗД $\Delta x_k = 0, \Delta y_k = 0$ в том и только том случае, когда пара $[x_k, y_k]$ является угловой точкой множества \mathcal{Z}_+ . Поэтому, если пара $[x_k, y_k]$ совпадает с угловой точкой множества \mathcal{Z}_+ , то $c_1(x_k, y_k) = c_2(x_k, y_k) = 0$. Для неособых пар $[x_k, y_k]$ коэффициенты $c_1(x_k, y_k)$ и $c_2(x_k, y_k)$ отличны от нуля.

Лемма 2. Пусть задача (1) является невырожденной. Пусть, кроме того, $[x_k, y_k]$ — неособая пара. Тогда $c_1(x_k, y_k) > 0$, $c_2(x_k, y_k) > 0$.

Если пара $[x_k, y_k]$ неособая, то, как следует из (7) и утверждения леммы 2, движение вдоль направлений, противоположных Δx_k , Δy_k , приводит к уменьшению значения функции $V(x, y)$, по крайней мере, при достаточно малых α . Чтобы получить наибольшее убывание функции $V(x, y)$ целесообразно взять шаг α_k равным

$$\alpha_k = \min \left\{ \alpha_k^{(1)}, \alpha_k^{(2)} \right\}, \quad (8)$$

где

$$\alpha_k^{(1)} = \frac{c_1(x_k, y_k)}{2c_2(x_k, y_k)}, \quad \alpha_k^{(2)} = \operatorname{argmax} \{ \alpha \geq 0 : x_k - \alpha \Delta x_k \geq 0_n, y_k - \alpha \Delta y_k \geq 0_n \}.$$

Итерационный процесс (5) с выбором шага согласно (8) назовем основным вариантом барьерно-проективного метода с наискорейшим спуском. Он позволяет находить решение невырожденной ЛЗД лишь в том случае, когда в ходе итеративного процесса все пары $[x_k, y_k]$ не являются угловыми точками множества \mathcal{Z}_+ за исключением, быть может, пары $[x_*, y_*]$.

Для произвольного $\delta > 0$ обозначим $\mathcal{W}_\delta(x_*, y_*) = \{ [x, y] \in \mathcal{Z}_+ : V(x, y) \leq \delta \}$.

Теорема 1. Пусть задача (1) является невырожденной. Тогда найдется такая окрестность $\mathcal{W}_\delta(x_*, y_*)$ решения $[x_*, y_*]$, что для любой начальной пары $[x_0, y_0]$ из этой окрестности основной вариант метода наискорейшего спуска находит $[x_*, y_*]$ не более чем за n итераций.

3. Модифицированный барьерно-проективный метод

Опишем теперь алгоритм решения задачи (1), основанный на измененных рекуррентных соотношениях (5), но по-прежнему использующий наискорейший спуск. Характерной особенностью этого алгоритма является его монотонность по отношению к множеству индексов $J_P(x, y)$ в том смысле, что если какой-либо индекс i покидает данное множество, то в дальнейшем он обратно в него не возвращается. При построении данной модификации метода существенным образом используется следующее свойство функции Лагранжа в особых парах.

Лемма 3. Пусть $[x, y]$ — особая пара, не совпадающая с решением задачи (1) и пусть $u = u(x, y)$. Тогда найдется по крайней мере один индекс $i \in J_Z(x, y)$, для которого $L_x^i(x, y, u) < 0$ или $L_y^i(x, y, u) < 0$.

Пусть задана начальная допустимая пара $[x_0, y_0]$. Предположим, что на некоторой k -й итерации получена допустимая пара $[x_k, y_k]$. Если данная пара есть решение задачи, то процесс останавливается. Иначе определяем новую пару $[x_{k+1}, y_{k+1}]$. При этом возможны два случая:

1) пара $[x_k, y_k]$ не является угловой точкой множества \mathcal{Z}_+ . Тогда новую пару вычисляем по формулам (5), где направления Δx_k и Δy_k вычисляются согласно (6).

2) пара $[x_k, y_k]$ есть угловая точка множества \mathcal{Z}_+ и $J_P(x_k, y_k) \neq \emptyset$. Для невырожденной задачи (1) при этом обязательно $J_Z(x_k, y_k) \neq \emptyset$ и $|J_Z(x_k, y_k)| = |J_P(x_k, y_k)|$.

Обозначим

$$\begin{aligned} J_Z^-(x, y) &= \left\{ i \in J_Z(x, y) : (M^T u(x, y))^{(i)} < 0 \right\}, \\ J_Z^+(x, y) &= \left\{ i \in J_Z(x, y) : u^{(i)}(x, y) > 0 \right\}. \end{aligned}$$

Выберем произвольный индекс j , принадлежащий либо непустому множеству $J_Z^-(x_k, y_k)$, либо непустому множеству $J_Z^+(x_k, y_k)$. Согласно лемме 3, по крайней мере, одно из данных множеств не пусто. Пусть, для определенности, это $J_Z^-(x_k, y_k)$. Зададимся $\epsilon > 0$ и положим

$$\bar{x}_k = x_k + \epsilon e_j, \quad \bar{y}_k = y_k, \quad (9)$$

где e_j — j -й единичный орт. Если $j \in J_Z^+(x_k, y_k)$, то вместо (9) возьмем

$$\bar{x}_k = x_k, \quad \bar{y}_k = y_k + \epsilon e_j. \quad (10)$$

Правую часть (6) вместо пары $[x_k, y_k]$ теперь вычисляем в паре $[\bar{x}_k, \bar{y}_k]$.

При данном изменении правых частей сохраняется утверждение леммы 2. Поэтому верхняя оценка для шага α_k , обеспечивающая наибольшее убывание функции $V(x, y)$, остается прежней, равной $\alpha_k^{(1)} = c_1/2c_2$.

Теорема 2. Пусть задача (1) является невырожденной. Пусть, кроме того, функция $V(x, y)$ принимает разные значения в разных угловых точках множества Z_+ . Тогда для любой начальной допустимой пары $[x_0, y_0]$ метод наискорейшего спуска находит решение ЛЗД за конечное число итераций.

Возможен также вариант метода, в котором одновременно изменяются несколько компонент вектора x_k^i в (9) или несколько компонент вектора y_k^j в (10). Важно только, чтобы индексы i и j входили в соответствующие множества $J_Z^-(x_k, y_k)$, $J_Z^+(x_k, y_k)$ и не совпадали между собой.

Список литературы

- [1] Cottle R.W., Pang J.-S., Stone R.E. *The linear complementarity problem*. Boston: Academic press, Inc., 1992, 398 p.
- [2] Ferris M.C., Pang J.-S. *Engineering and economic applications of complementarity problems*. - SIAM Review. 1997, V. 39, p. 669-713.
- [3] Kojima M., Mizuno Sh., Yoshize A. *A polynomial-time algorithm for a class of linear complementarity problems*. - Mathematical programming, 1989, v. 44, p. 1-26.
- [4] Monteiro R.D.C., Wright S.J. *Superlinear primal-dual affine scaling algorithms for LCP*. - Mathematical Programming, 1995, v. 69, p. 311-333.
- [5] Втюрина М.В., Жадан В.Г. *Барьерно-градиентные методы для линейных задач допустимости*. Сообщения по прикладной математике. М.: ВЦ РАН, 2003, 32 с.
- [6] Евтушенко Ю.Г., Жадан В.Г. *Барьерно-проективные методы для решения задач нелинейного программирования*. - Журнал вычисл. матем. и матем. физ., 1994, т. 34, № 5, с. 669-684.

INTERIOR POINT METHOD WITH STEEPEST DESCENT FOR LINEAR COMPLEMENTARITY PROBLEM

V.G. Zhadan, M.V. Vtyurina

Dorodnicyn Computing Centre of RAS, Moscow

e-mail: zhadan@ccas.ru

Moscow Institute of Physics and Technology (State University)

e-mail: vtyurina@scph.mipt.ru

Abstract. The linear complementarity problem is considered. For its solution the interior-point method is suggested. In this method the steepest descent approach is used for choosing the step length. The method is generalization of the barrier-projective method which had been proposed early for the linear programming problem. The finite local and nonlocal convergence of the method is proved.

Key words: linear complementarity problem, interior point method, barrier-projective method, steepest descent

КОМБИНИРОВАННЫЕ АЛГОРИТМЫ ВНУТРЕННИХ ТОЧЕК¹

В.И.Зоркальцев

Институт систем энергетики им. Мелентьева СО РАН, Иркутск
E-mail: : zork@isem.sei.irk.ru

Аннотация. Излагаются алгоритмы внутренних точек для решения задач линейного программирования, сочетающие в едином вычислительном процессе ввод в область допустимых решений с оптимизацией. Исходные варианты таких алгоритмов разработаны автором в 70-х годах и нашли широкое практическое применение. Приводится теорема, содержащая теоретическое обоснование алгоритмов. Рассматриваются возможности повышения вычислительной эффективности алгоритмов за счёт введения новых правил выбора направлений корректировки решения на каждой итерации - в виде оптимальной линейной комбинации нескольких векторов улучшения решения.

Ключевые слова: метод внутренних точек, линейное программирование.

Рассматриваются взаимно - двойственные задачи линейного программирования:

$$(c, x) \rightarrow \min, Ax = b, x \geq 0; \quad (1)$$

$$(b, u) \rightarrow \max, g(u) \geq 0, \text{ где} \quad (2)$$
$$g(u) = c - A^T u.$$

Заданными являются матрица A размерности $m \times n$, векторы $b \in R^m, c \in R^n$. Обозначим X, \bar{X} множества допустимых по ограничениям и оптимальных значений векторов переменных $x \in R^n$ задачи (1). Пусть U, \bar{U} - множества допустимых и оптимальных значений векторов переменных $u \in R^m$ задачи (2).

Алгоритм, сочетающий ввод в области допустимых решений с оптимизацией. Пусть как - то выбран вектор $x^0 \in R^n$ с положительными всеми компонентами, $x^0 > 0$. Этот вектор служит стартовой точкой для итеративного процесса:

$$x^{k+1} = x^k + \lambda_k s^k, k = 0, 1, 2, \dots \quad (3)$$

где k - номер итерации, s^k - вектор R^n направления корректировки решения, λ_k - шаг корректировки. Отметим, что за счёт специального правила выбора шага векторы x^k на всех итерациях удовлетворяют ограничениям - неравенствам задачи (1) в строгой форме, т.е.

$$x^k > 0, k = 0, 1, 2, \dots \quad (4)$$

Направление корректировки решения s^k является решением следующей вспомогательной задачи k -ой итерации относительно вектора переменных $s \in R^n$:

$$(c, s) + \frac{1}{2} s^T D_k s \rightarrow \min, As = r^k, \text{ где} \quad (5)$$

¹Работа выполнена при поддержке гранта РФФИ N 05-01-00587

$$r^k = b - Ax^k,$$

$$D^k = \text{diag } d^k,$$

вектор весовых коэффициентов $d^k \in R^n$ имеет компоненты:

$$d_j^k = (x_j^k)^p, j = 1, \dots, n \quad (6)$$

Здесь p заданный параметр алгоритма - вещественное число из диапазона $[1, 3]$,

$$1 \leq p \leq 3. \quad (7)$$

Вспомогательная задача (5) не имеет решения в том и только том случае, если несовместна система уравнений $Ax=b$, что выявляется сразу на нулевой итерации.

При решении задачи (5) методом неопределённых множителей Лагранжа сначала находим вектор $u^k \in R^n$ множителей ограничений этой задачи как решение системы уравнений относительно $u \in R^n$:

$$(AD_k A^T)u = r^k + D_k A^T c \quad (8)$$

Затем полагаем

$$s^k = D_k(A^T u^k - c) \quad (9)$$

Шаг корректировки решения вычисляется по правилу:

$$\lambda_k = \min\{1, B_k\}, \text{ если } r^k \neq 0 \quad (10)$$

$$\lambda_k = B_k, \text{ если } r^k = 0, \text{ где} \quad (11)$$

$$B_k = \gamma \rho_k, \quad (12)$$

$$\rho_k = \max\{\lambda : x^{k+1} + \lambda s^k \geq 0\}, \quad (13)$$

γ - заданный параметр, удовлетворяющий условию:

$$0 < \gamma < 2/(p+1). \quad (14)$$

По правилу (13) нельзя вычислить ρ_k , если $s^k \geq 0$. В этом случае при $r^k \neq 0$ полагаем $s^k = 1$. При $r=0$ возможны два варианта:

если $s^k = 0$, то полученное решение является оптимальным;

если $s^k \neq 0$, то задача (1) не имеет оптимальных решений, по направлению s^k можно неограниченно уменьшать значение целевой функции, не выходя из области допустимых решений.

Отметим, что из сформулированных правил выбора шага следует неравенство

$$x^{k+1} \geq (1 - \gamma)x^k \quad (15)$$

Из чего в силу (14) и условия $x^0 > 0$ следует выполнение неравенства (4) на всех итерациях.

Для изложенного алгоритма на всех итерациях выполняется равенство

$$r^{k+1} = (1 - \lambda_n)r^k \quad (16)$$

Можно выделить два этапа вычислений по изложенному алгоритму. На первом этапе осуществляется ввод в область допустимых решений: при $x^k \notin X$ согласно (10) происходит монотонное сокращение абсолютных значений всех компонент вектора невязок r^k .

На втором этапе осуществляется процесс оптимизации в области допустимых решений.

Модификацией изложенного является вычислительный процесс, в котором на этапе ввода в область допустимых решений не учитывается целевая функция, т.е. при $r^k \neq 0$ полагается $s=0$. Первоначально изложенный здесь алгоритм и его модификации были предложены для вариантов с $p=2$. Подробное описание такого варианта алгоритма и его модификация, опыт их практического использования, представлены в книге [1]. Исходной основой в разработке послужил алгоритм оптимизации в области допустимых решений И.Дикина [2]. Следует отметить, что в 70-х годах разработки и теоретические исследования аналогичных алгоритмов внутренних точек осуществлялись Ю.Г. Евтушенко и В.Г. Жаданом [3]. Более широкий класс алгоритмов внутренних точек, обзор результатов по их обоснованию представлен в статье [4]. Первоначально теоретические результаты по обоснованию алгоритмов были получены при предположении о невырожденности задачи. Принципиальное значение имеет обоснование алгоритмов без этого предположения. Обоснование без использования предположения о невырожденности были получены сначала [5] для алгоритма с $p=1$ и затем [6], [7] для алгоритма с $p \in (1, 3]$. В частности исследования в [5-7] содержат доказательство следующего утверждения:

Теорема. Пусть $\bar{X} \neq \emptyset, X \neq \bar{X}$ и имеется вектор $x \in X$ такой что $x > 0$. Тогда для изложенных алгоритмов

1) через конечное число итераций будет получено решение $x^k \in riX$;

2) существуют векторы $\bar{x} \in ri\bar{X}, \bar{u} \in ri\bar{U}$ такие, что $x^k \rightarrow \bar{x}, u^k \rightarrow \bar{u}$ при $k \rightarrow \infty$, причем величины $\|x^k - \bar{x}\|, \|u^k - \bar{u}\|$ сходятся к нулю со скоростью геометрической прогрессии.

Алгоритм с оптимальной комбинацией направлений ввода в область допустимых решений и оптимизации. Излагаемый ниже новый алгоритм базируется на следующей аппроксимации задачи (1):

$$\Phi(x) \rightarrow \min, x \geq 0, \quad (17)$$

где

$$\Phi(x) = M\|Ax - b\| + (c, x) \quad (18)$$

при достаточно большом $M > 0$.

Приведём только требуемые изменения правил вычисления. Направление улучшения решения определим в виде линейной комбинации двух направлений. Пусть

$$s^k(\alpha) = z^{k,1} + \alpha z^{k,2},$$

где α - любой, искомый в дальнейшем параметр,

$$z^{k,1} = D_k A^T u^{k,1}, \quad (19)$$

$$z^{k,2} = D_k (A^T u^{k,2} - c). \quad (20)$$

Вектор $u^{k,1}$ является решением системы линейных уравнений

$$(AD_k A^T)u = r^k \quad (21)$$

Вектор $u^{k,2}$ определяется как решение следующей системы уравнений

$$(AD_k A^T)u = D_k A^T c \quad (22)$$

Заметим, что у систем (21), (22) одна и та же матрица позволяет осуществлять одновременный поиск решений обеих этих систем.

Пусть

$$\rho_k(\alpha) = \min\{\lambda : x^k + \lambda s^k(\alpha) \geq 0\}, \quad (23)$$

$$\lambda_k(\alpha) = \arg \min\{\Phi(x^k + \lambda s^k(\alpha)) : 0 \leq \lambda \leq \gamma \rho_k(\alpha)\}, \quad (24)$$

$$\alpha_k = \min_{\alpha} \Phi(x^k + \lambda_k(\alpha) s^k(\alpha)). \quad (25)$$

В итеративном переходе (3) положим

$$s^k = s^k(\alpha^k), \lambda_k = \lambda_k(\alpha^k).$$

При практической реализации данных алгоритмов могут использоваться различные процедуры приближенного решения задачи однопараметрической оптимизации (25).

Конечно, при реализации данного алгоритма целесообразно осуществлять только приближённое решение задачи (25). Заметим, что если не искать оптимальное значение α_k по правилу (25), а положить $\alpha_k = 1$, то получим исходный алгоритм (3) - (16). В вычислительном отношении задача решения двух систем уравнений (21), (22) не сложнее, чем решение одной системы, как в исходном случае (8), поскольку у систем (21), (22) одинаковые матрицы. Возможны также алгоритмы с направлениями спуска в виде оптимальной комбинации 3-х векторов, вычисляемых параллельно на данной итерации.

Список литературы

- [1] И.И. Дикин, В.И. Зоркальцев *Итеративное улучшение задачи математического программирования (алгоритмы метода внутренних точек)*. - Новосибирск: Наука 1980.
- [2] И.И. Дикин *Итеративное улучшение задачи линейного программирования*. - ДАН, СССР, 1967, №4.
- [3] Ю.Г. Евтушенко, В.Г. Жадан *Релаксационный метод решения задач нелинейного программирования*. - ЖВМ и МФ, 1977, №4.
- [4] В.И. Зоркальцев *Алгоритмы внутренних точек в линейном программировании. Оптимизация, управление, интеллект*. - 1998, №1.
- [5] В.И. Зоркальцев *Относительно внутренние точки множества оптимальных решений*. - Препринт Коми ФАН СССР, 1984.
- [6] В.И. Зоркальцев *Обоснование семейства проективных алгоритмов (процесс, совмещающий ввод в областидопустимых решений с оптимизацией)*. - Препринт СЭИ СО РАН, 1995.
- [7] В.И. Зоркальцев *Обоснование алгоритма внутренних точек*. - ЖВМ и МФ, 1999, №2.

COMBINATIVE INTERIOR POINT ALGORITHMS

V.I. Zorkaltsev

E-mail: zork@isem.sei.irk.ru

Summary. Interior point algorithms for solving linear programming problems that combine in a single calculation process input into feasible set and optimization are stated. The initial variants of such algorithms were developed by the author in 70th and have found wide practical application. A theorem containing the theoretical base of algorithms is proposed. Possibility of increase of calculation efficiency of algorithms is investigated. The new rules of finding direction at each iteration (as an optimal linear combination of some vectors) are proposed.

КОМБИНИРОВАННЫЕ АЛГОРИТМЫ ВНУТРЕННИХ ТОЧЕК ДЛЯ ЗАДАЧ ЛИНЕЙНОГО ПРОГРАММИРОВАНИЯ ¹

В.И. Зоркальцев, А.Ю. Филатов

Институт систем энергетики им. Л.А. Мелентьева, Иркутск
e-mail: fial@isem.sei.irk.ru

Аннотация. В работе предлагаются комбинированные алгоритмы внутренних точек для решения задач линейного программирования. Их особенностью является то, что направление корректировки вычисляется как линейная комбинация аффинно-масштабирующего и центрирующего направлений. Последнее находится как решение задачи минимизации квадратичной аппроксимации логарифмической барьерной штрафной функции. Приведены результаты экспериментального исследования предложенных алгоритмов, демонстрирующие их высокую эффективность.

Ключевые слова: линейное программирование, методы внутренних точек, аффинно-масштабирующие алгоритмы, алгоритмы центрального пути

Рассматривается пара взаимно-двойственных задач линейного программирования

$$c^T x \rightarrow \min_{x \in X}, \quad X = \{x \in R^n : Ax = b, x \geq 0\}, \quad (1)$$

$$b^T u \rightarrow \max_{u \in U}, \quad U = \{u \in R^m : g(u) \equiv c - A^T u \geq 0\}, \quad (2)$$

где $c \in R^n$, $b \in R^m$, A - матрица размерности $m \times n$, $\text{rank } A = m$.

Итеративный процесс комбинированного алгоритма начинается с произвольной стартовой точки $x^1 > 0$. На каждой итерации $k = 1, 2, \dots$ ищется вектор невязок балансовых ограничений-равенств

$$r^k = b - Ax^k.$$

Итеративный переход осуществляется по правилу

$$x^{k+1} = x^k + \lambda^k \Delta x^k,$$

где Δx^k - направление корректировки, а λ^k - шаг корректировки, задаваемый из соображений невыхода переменных прямой задачи за пределы внутренности допустимой области.

Для нахождения направления корректировки Δx^k решается следующая параметрическая задача минимизации:

$$c^T \Delta x - \mu \sum_{j=1}^n \frac{\Delta x_j}{x_j^k} + \frac{1}{2} \sum_{j=1}^n \frac{\Delta x_j^2}{(x_j^k)^2} \rightarrow \min_{\Delta x \in R^n}, \quad A \Delta x = r^k. \quad (3)$$

Заметим, что если значение параметра μ равно нулю, то формулы задачи (3) полностью совпадают с формулами аффинно-масштабирующего алгоритма [1]. При $\mu = 1$ задача (3) дает такое же направление, как и задача, решаемая в алгоритме центрального пути [2],

¹Работа выполнена при финансовой поддержке интеграционного проекта СО РАН №6, выполняемого в содружестве с учеными УрО РАН, и Фонда содействия отечественной науке

за исключением того, что теперь не требуется иметь труднополучаемого [3] стартового приближения.

Задача (3) решается методом множителей Лагранжа. Двойственные оценки u^{k+1} ограничений-равенств $A\Delta x = r^k$ находятся по формуле

$$u^{k+1}(\mu) = (AX_k^2 A^T)^{-1}(AX_k^2 c + r^k - \mu b).$$

Направление корректировки вычисляется следующим образом:

$$\Delta x^k(\mu) = \mu x^k - X_k^2 g(u^{k+1}(\mu)).$$

Легко обнаружить, что функции $u^{k+1}(\mu)$ и $\Delta x^k(\mu)$ являются линейными относительно μ . Таким образом, можно отыскать 2 вектора

$$u^{k+1}(0) = (AX_k^2 A^T)^{-1}(AX_k^2 c + r^k)$$

и

$$u^{k+1}(1) = (AX_k^2 A^T)^{-1}(AX_k^2 c + r^k - b),$$

что принципиально не сложнее исходной задачи, поскольку здесь по сути решается одна и та же система линейных уравнений с 2 правыми частями, и искомая функция $u^{k+1}(\mu)$ будет иметь вид

$$u^{k+1}(\mu) = (1 - \mu)u^{k+1}(0) + \mu u^{k+1}(1).$$

Аналогично, итоговое направление корректировки $\Delta x^k(\mu)$ будет линейной комбинацией

$$\Delta x^k(\mu) = (1 - \mu)\Delta x^k(0) + \mu\Delta x^k(1)$$

аффинно-масштабирующего

$$\Delta x^k(0) = -X_k^2 g(u^{k+1}(0))$$

и центрирующего

$$\Delta x^k(1) = x^k - X_k^2 g(u^{k+1}(1))$$

направлений. При этом для каждого направления легко найти максимальный шаг, с которым можно производить корректировку, оставаясь внутри допустимой области по ограничениям-неравенствам:

$$\lambda^k(\mu) = \gamma \min_{j: \Delta x_j^k(\mu) < 0} \frac{-x_j^k}{\Delta x_j^k(\mu)}, \quad \gamma \in (0; 1).$$

Таким образом, путем небольшого усложнения алгоритма мы получили целое множество направлений корректировки $\Delta X = \{\Delta x^k(\mu) : \mu \in [0; 1]\}$. Из этого множества возможных направлений нужно выбрать наиболее предпочтительное в целях скорейшего решения задачи.

Работа алгоритма разделяется на 2 этапа. На первом - необходимо максимально быстро найти допустимую точку, удовлетворяющую равенствам $Ax = b$. Поскольку уменьшение невязки ограничений-равенств осуществляется в соответствии с формулой

$$r^{k+1} = (1 - \lambda^k)r^k,$$

то на первом этапе в качестве критерия можно использовать задачу

$$\lambda^k(\mu) \rightarrow \max_{\mu \in [0;1]}, \quad \lambda^k(\mu) \leq 1. \quad (4)$$

На втором этапе необходимо максимально быстро сокращать невязку двойственности (разницу значений целевых функций прямой и двойственной задачи), равную

$$c^T x - b^T u = \sum_{j=1}^n x_j g_j(u).$$

Таким образом, критерием второго этапа будет следующая задача:

$$\sum_{j=1}^n (x_j^k + \lambda^k(\mu) \Delta x_j^k(\mu)) g_j(u^{k+1}) \rightarrow \min_{\mu \in [0;1]}. \quad (5)$$

И на первом, и на втором этапе необязательно решать задачи (4) и (5) точно. Достаточно вычислить критерии для нескольких точек на интервале $\mu \in [0; 1]$, найти наилучшее значение μ^k , отыскать направление $\Delta x^k(\mu^k)$ и шаг $\lambda^k(\mu^k)$ корректировки и осуществить итеративный переход

$$x^{k+1} = x^k + \lambda^k(\mu^k) \Delta x^k.$$

Нельзя гарантировать, что локальный выигрыш на одной итерации ведет и к ускорению всего вычислительного процесса относительно фиксированного случая $\mu = 0$, соответствующего формулам аффинно-масштабирующего алгоритма. Действительно, вычислительные процессы для разных значений μ приводят к различным траекториям. Поэтому одна из целей экспериментального исследования состоит в проверке: имеет ли место ускорение на тестовых задачах.

Сопоставим объем вычислений, потребовавшийся для получения решения тестовых задач аффинно-масштабирующим и комбинированным алгоритмом. Итеративный процесс начинался с единичного вектора соответствующей размерности ($x^1 = e$). Критерием останова было достижение невязкой двойственности $\sum x_j g_j(u)$ заданного критического значения $\varepsilon = 0,000005$.

Как уже отмечалось, в пределах одной итерации параметризация в комбинированных алгоритмах не влияет существенно на увеличение объема вычислений. Таким образом, алгоритмы имеют приблизительно одинаковый объем вычислений на каждой итерации - наибольшую сложность представляет обращение симметричной положительно определенной матрицы размерности $m \times m$. Поэтому данные о числе итераций могут быть использованы в качестве сравнительной характеристики объема вычислений.

Для тестирования были сгенерированы наборы случайных задач различной размерности (по 10 - каждой). В табл. 1 внесено количество итераций, потребовавшееся алгоритмам для решения каждой из них (значения указаны через дробь). Задачи (4) и (5) решались приближенно. Значения критериев вычислялись в 10 точках, при $\mu_i^k = 1/2^{i-1}$, $i = 1, 2, \dots, 9$; $\mu_{10}^k = 0$.

Рассмотрим еще одну небольшую модификацию комбинированного алгоритма - когда значения μ^k могут превышать 1, что означает усиление центрирующего направления. Пусть к 10 точкам μ^k , для которых вычисляются значения критериев (4) и (5), добавится еще $\mu^k = 2$. Значения числа итераций для этого варианта алгоритма также запишем в табл. 1.

Среднее значение it и среднеквадратическое отклонение σ числа итераций запишем в сводную табл. 2.

Таблица 1. Число итераций, необходимое для решения задач линейного программирования различными алгоритмами

алг./размерн.	20 × 40	40 × 80	100 × 200	200 × 500
аффинно-масштабирующий	36/32/22/37	31/34/29/40	26/63/25/22	26/31/27/32
	31/29/36	33/28/25	31/24/22	30/25/26
	20/27/41	39/29/22	28/25/24	26/30/33
комбинированный	28/22/18/23	21/19/23/34	20/47/21/20	22/23/22/22
	33/26/25	26/21/27	20/20/20	23/22/23
	19/29/28	31/20/19	21/22/20	22/24/23
комбинированный расширенный $\mu \in [0; 2]$	21/19/18/20	22/19/19/25	21/49/21/21	23/22/21/22
	33/28/21	22/23/28	21/20/20	23/23/23
	18/30/28	21/19/19	22/21/20	22/23/23

Таблица 2. Среднее значение и среднеквадратическое отклонение числа итераций, необходимое для решения задач

алг./размерн.	20 × 40	40 × 80	100 × 200	200 × 500
аффинно-масштабирующий	$it = 31, 1$	$it = 33, 0$	$it = 29, 1$	$it = 28, 6$
	$\sigma = 6, 39$	$\sigma = 6, 42$	$\sigma = 11, 62$	$\sigma = 2, 76$
комбинированный	$it = 25, 1$	$it = 24, 1$	$it = 23, 1$	$it = 22, 6$
	$\sigma = 4, 44$	$\sigma = 4, 97$	$\sigma = 7, 99$	$\sigma = 0, 66$
комб.расширенный $\mu \in [0; 2]$	$it = 23, 6$	$it = 21, 7$	$it = 23, 6$	$it = 22, 5$
	$\sigma = 5, 28$	$\sigma = 2, 87$	$\sigma = 8, 49$	$\sigma = 0, 67$

На основе проведенных экспериментов видно, что с помощью предложенной процедуры параметризации можно примерно на четверть сократить объем вычислений. На большинстве задач дополнительное ускорение было получено в результате расширения интервала $\mu \in [0; 1]$ до $\mu \in [0; 2]$.

Список литературы

- [1] Дикин И.И., Зоркальцев В.И. *Итеративное решение задач математического программирования: алгоритмы метода внутренних точек.* - Новосибирск: Наука, 1980, 220с.
- [2] Зоркальцев В.И. *Алгоритмы оптимизации в конусе центрального пути.* // Журн. вычисл. математики и мат. физики, 2000, т.40, №2, с. 318-327.
- [3] Monteiro R., Adler I. *Interior path-following primal-dual algorithms. Part 1: Linear programming.* // Mathematical programming, 1989, №44, p. 27-49.

COMBINED INTERIOR POINT ALGORITHMS FOR LINEAR PROGRAMMING PROBLEMS

V.I.Zorkaltsev, A.Yu.Filatov

*Institute of energy systems named after L.A.Melent'ev, Irkutsk
e-mail: fial@isem.sei.irk.ru*

Abstract. The combine interior point algorithms are proposed for solving linear programming problems. The key new feature is that the direction at each iteration is computed as a linear combination of the affine-scaling and the centering directions. We can find the last one as a solution of the logarithmic barrier function minimization problem. The results of the experimental research showed high efficiency of the proposed algorithms.

Key words: linear programming, interior point methods, affine-scaling algorithms, path-following algorithms

СУЩЕСТВОВАНИЕ КОМИТЕТА СИСТЕМЫ ЛИНЕЙНЫХ НЕРАВЕНСТВ¹

К.С. Кобылкин

*Институт математики и механики УрО РАН, Екатеринбург
e-mail: kobcom@narod.ru, ksy@sugres.pssr.ru*

Аннотация. В статье приводится одно достаточное условие существования комитета бесконечной системы строгих однородных линейных неравенств в многомерном пространстве. В случае системы m линейных неравенств дается критерий существования минимального комитета этой системы, состоящего из m членов.

Ключевые слова: система линейных неравенств, комитет.

Введение

Обобщением понятия решения системы линейных неравенств в \mathbf{R}^n на случай несовместных систем является *комитет* – конечный набор точек из \mathbf{R}^n такой, что каждому неравенству системы удовлетворяет более половины его членов. Если система совместна, комитетом будет, например, набор из одного или нескольких элементов, являющихся решениями этой системы. Одной из задач теории комитетов является нахождение условий существования комитета произвольной системы строгих линейных неравенств. В [1] дан критерий существования комитета конечной системы строгих линейных неравенств в многомерном пространстве, а в [2] получен критерий существования комитета бесконечной двумерной системы строгих однородных линейных неравенств.

1. Постановка задачи

Рассмотрим произвольную, в общем случае бесконечную систему строгих однородных линейных неравенств

$$(c_j, x) > 0, j \in J, x \in \mathbf{R}^n. \quad (1)$$

Приведем два определения, данные в [1].

Определение 1. *Комитетом* системы линейных неравенств (1) называется такой конечный набор (с возможными повторениями) векторов из \mathbf{R}^n , что каждому неравенству системы (1) удовлетворяет более половины членов этого набора. Комитет с наименьшим для данной системы числом членов (с учетом их кратности) называется *минимальным*.

Определение 2. *Комитетом аффинных функций*, разделяющим два подмножества точек \mathcal{A} и \mathcal{B} пространства \mathbf{R}^n , называется такой конечный набор аффинных функций, что в любой точке из \mathcal{A} (соответственно из \mathcal{B}) более чем половина функций этого набора положительна (соответственно отрицательна).

Для краткости комитет аффинных функций, разделяющий два множества, будем называть разделяющим комитетом. Рассмотрим задачу построения комитета, разделяющего два подмножества в \mathbf{R}^n . Кроме того, рассмотрим задачу нахождения комитета системы строгих однородных линейных неравенств в \mathbf{R}^n . Известно, что ([1]) набор аффинных

¹Работа выполнена при поддержке РФФИ (проект №01-01-96454)

функций $\{f_i\}_{i=1}^q$, $f_i(x) = (u_i, x) + \alpha_i$, $u_i, x \in \mathbf{R}^n$, $\alpha_i \in \mathbf{R}$, $i = \overline{1, q}$, является комитетом, разделяющим два подмножества точек \mathcal{A} и \mathcal{B} в \mathbf{R}^n , тогда и только тогда, когда набор векторов $\{(u_i, \alpha_i)\}_{i=1}^q \subset \mathbf{R}^{n+1}$ является комитетом системы

$$(c, z) > 0, c \in \mathcal{A}' \cup (-\mathcal{B}'), z \in \mathbf{R}^{n+1},$$

где $\mathcal{A}' = \{[a, 1] : a \in \mathcal{A}\}$ и $\mathcal{B}' = \{[b, 1] : b \in \mathcal{B}\}$.

Отсюда следует, что решив первую из приведенных задач, заданную в \mathbf{R}^n , можно получить решение второй задачи, заданной в \mathbf{R}^{n+1} . Отметим также ([3]), что решение произвольной конечной системы неоднородных линейных неравенств можно найти построив гиперплоскость, разделяющую некоторые два конечных точечных множества.

2. Условия существования комитета

Рассмотрим задачу о разделении комитетом двух множеств. Известно, что ([1]) любые два конечных непересекающихся подмножества в \mathbf{R}^n могут быть разделены комитетом. Следующая теорема обобщает этот результат на случай бесконечных множеств.

Теорема 1. Для любых двух непересекающихся замкнутых подмножеств в \mathbf{R}^n , одно из которых ограничено и имеет конечное число предельных точек, существует комитет, разделяющий эти два подмножества.

Приведем некоторое обобщение этой теоремы. Рассмотрим замкнутые непересекающиеся подмножества \mathcal{A} и \mathcal{B} в \mathbf{R}^n , одно из которых, скажем, множество \mathcal{A} , является ограниченным. Пусть \mathcal{A}_1 – совокупность всех предельных точек множества \mathcal{A} , \mathcal{A}_2 – совокупность предельных точек множества \mathcal{A}_1 , и т.д., \mathcal{A}_k – совокупность предельных точек множества \mathcal{A}_{k-1} , $k = 3, \dots$. Если для некоторого $k = k_0$ выполнено равенство $\mathcal{A}_{k_0} = \emptyset$, то существует комитет, разделяющий \mathcal{A} и \mathcal{B} .

Условие ограниченности одного из подмножеств в теореме 1 при отсутствии дополнительных условий на другое подмножество является существенным. Так, например, множество четных чисел на числовой прямой нельзя отделить комитетом от множества нечетных чисел. Аналогичный пример можно привести также и в многомерном пространстве.

Рассмотрим задачу о комитете системы строгих однородных линейных неравенств (1). Очевидно, что в случае, когда $0 \in \mathcal{C} = \{c_j : j \in J\}$, для системы (1) не существует комитета. Пусть $0 \notin \mathcal{C}$, и тогда без ограничения общности можно считать, что $\mathcal{C} \subset S$, где S – единичная сфера с центром в 0 . Пересечение сферы S с произвольным открытым полупространством, граница которого содержит 0 , назовем *открытой полусферой* на сфере S . Если T – открытая полусфера на пересечении открытого полупространства P со сферой S , то через $\text{bd } T$ обозначим пересечение сферы S с границей полупространства P .

Как известно ([1]), конечная система (1) имеет комитет тогда и только тогда, когда $-\mathcal{C} \cap \mathcal{C} = \emptyset$. Приводимое ниже следствие является обобщением этого результата на случай бесконечной системы (1).

Следствие. Пусть $\mathcal{C} \cap \text{bd } T = \emptyset$, где T – некоторая открытая полусфера сферы S . Если $-\mathcal{C} \cap \mathcal{C} = \emptyset$, множество $T \cap \mathcal{C}$ содержит все свои предельные точки, лежащие на полусфере T , а множество $-T \cap \mathcal{C}$ замкнуто и имеет конечное число предельных точек, то существует комитет системы (1).

Пусть система (1) конечна и $-\mathcal{C} \cap \mathcal{C} = \emptyset$.

Теорема 2. Для того, чтобы система (1) из m неравенств имела минимальный комитет из m членов, необходимо и достаточно, чтобы любые два ее неравенства составляли вместе с некоторым неравенством системы (1) несовместную подсистему. При этом,

если m – четно, то найдутся два ее неравенства, которые вместе с любым третьим неравенством образуют совместную подсистему.

Список литературы

- [1] Вл.Д. Мазуров *Метод комитетов в задачах оптимизации и классификации*. М. : Наука, 1990, 248 с.
- [2] А.И. Кривоногов *Некоторые вопросы обоснования комитетных алгоритмов*. - В сб.: Классификация и оптимизация в задачах управления. Свердловск: УНЦ АН СССР, 1981, с. 39-51.
- [3] К.С. Kobylkin *Constructing a committee of a system of linear inequalities*. - Proceedings of 7th Int. Conf. "Pattern Recognition and Image Analysis", 18-23 October, 2004, Vol. 1, P. 62-65.

EXISTENCE OF A COMMITTEE FOR A SYSTEM OF LINEAR INEQUALITIES

K.S. Kobylkin

*Institute of Mathematics and Mechanics, Ural Branch of RAS, Ekaterinburg
e-mail: kobcom@narod.ru, ksy@sugres.pssr.ru*

Abstract. A sufficient condition of the existence of a committee for an infinite system of strict homogeneous linear inequalities in a multidimensional space is given. In the case of a system of m linear inequalities a criterion of the existence of its minimal committee of m members is indicated.

Key words: linear inequality system, committee.

РЕШЕНИЕ СИСТЕМ ЛИНЕЙНЫХ УРАВНЕНИЙ С СИММЕТРИЧНЫМИ ПОЛОЖИТЕЛЬНО ОПРЕДЕЛЕННЫМИ МАТРИЦАМИ МЕТОДОМ СОПРЯЖЕННЫХ НАПРАВЛЕНИЙ ¹

С.М. Пержабинский, А.Ю. Филатов

Институт систем энергетики им. Л.А. Мелентьева, Иркутск
e-mail: fial@isem.sei.irk.ru

Аннотация. В работе исследуется возможность решения систем линейных уравнений с симметричными положительно определенными матрицами итеративными методами, в частности, методом сопряженных направлений. Данная задача представляет основную вычислительную сложность на каждой итерации алгоритмов внутренних точек для задач математического программирования и систем неравенств, и ускорение ее решения ускоряет работу всего алгоритма. Более того, специфика алгоритмов внутренних точек позволяет решать задачу приближенно. Поэтому использование алгоритмов, принципиально отличающихся от метода Халецкого, может оказаться перспективным. В работе дано описание алгоритма, а также приведены результаты экспериментального исследования на тестовых задачах.

Ключевые слова: системы линейных уравнений, метод сопряженных направлений, методы внутренних точек

В алгоритмах внутренних точек (например, [1]) для решения задач линейного программирования, систем линейных неравенств и нелинейных задач, решаемых с помощью методов линеаризации [2], итеративный переход на каждой итерации $k = 1, 2, \dots$ осуществляется по формуле $x^{k+1} = x^k + \lambda_k \Delta x^k$, где λ_k - шаг корректировки, а Δx^k - направление корректировки.

Для вычисления направления Δx^k требуется найти вектор u^k двойственных оценок, решив задачу

$$AD_k A^T u = r^k. \quad (1)$$

Здесь A - матрица размерности $m \times n$ ($\text{rank } A = m$) ограничений $Ax = b$ исходной задачи, D_k - диагональная матрица положительных весовых коэффициентов порядка n , а $r^k \in R^m$ - вектор правых частей, меняющиеся по итерациям.

Системы линейных уравнений с симметричной положительно определенной матрицей $AD_k A^T u$ размерности $m \times m$ можно решать точными методами, в частности, методом Халецкого, для чего требуется около $m^3/2$ арифметических операций. Альтернативой является использование итеративных методов, одним из которых является метод сопряженных направлений.

Задача (1) эквивалентна задаче безусловной минимизации квадратичной функции

$$f(u) = \frac{1}{2} u^T B u - r^T u, \quad (2)$$

где $B = AD_k A^T$, $r = r^k$.

¹Работа выполнена при финансовой поддержке интеграционного проекта СО РАН №6, выполняемого в содружестве с учеными УрО РАН, и Фонда содействия отечественной науке

Начиная с некоторого произвольно выбранного приближения u^0 , находим на каждой итерации $t = 1, 2, \dots$ метода сопряженных направлений вектор

$$\Delta u^t = \nabla f(u^{t-1}) - \beta_t \Delta u^{t-1} = Bu^{t-1} - r - \beta_t \Delta u^{t-1}, \quad (3)$$

где способ определения шага β_t задает тот или иной вариант метода. В нашей работе на первой итерации $\beta_1 = 0$, далее зададим

$$\beta_t = \frac{(Bu^{t-1} - r)^T B \Delta u^{t-1}}{\Delta u^{t-1} B \Delta u^{t-1}}. \quad (4)$$

Итеративный переход осуществляется по формуле

$$u^t = u^{t-1} + \alpha_t \Delta u^t, \quad (5)$$

где шаг α_t находим как решение задачи одномерной минимизации функции $f(u^{t-1} + \alpha \Delta u^t)$:

$$\begin{aligned} f(u^{t-1} + \alpha \Delta u^t) &= \frac{1}{2}(u^{t-1} + \alpha \Delta u^t)^T B(u^{t-1} + \alpha \Delta u^t) - r^T(u^{t-1} + \alpha \Delta u^t) = \\ &= \frac{1}{2}(u^{t-1})^T B u^{t-1} + \alpha(u^{t-1})^T B \Delta u^t + \frac{1}{2}\alpha^2(\Delta u^t)^T B \Delta u^t - r^T u^{t-1} - \alpha r^T \Delta u^t \rightarrow \min_{\alpha}. \end{aligned}$$

Найдем конкретное выражение для α_t , приравняв производную к нулю:

$$\begin{aligned} \frac{\partial f(u^{t-1} + \alpha \Delta u^t)}{\partial \alpha} &= (u^{t-1})^T B \Delta u^t + \alpha(\Delta u^t)^T B \Delta u^t - r^T \Delta u^t = 0, \\ \alpha_t &= -\frac{(Bu^{t-1} - r)^T B \Delta u^t}{\Delta u^t B \Delta u^t}. \end{aligned} \quad (6)$$

Известно (например, из [3]), что метод сопряженных направлений (3)–(6), начиная с произвольно выбранного u^0 , получает точку безусловного минимума u^* квадратичной функции (2) не более, чем за m итераций. Вектор u^* удовлетворяет условию (1). В то же время специфика алгоритмов внутренних точек позволяет решать задачу (1) приближенно. Действительно, на первых итерациях достаточно [4] искать направление корректировки Δx^k , используя вектор u^k , для которого $AD_k A^T u^k = \tilde{r}^k \neq r^k$, а на финальной стадии вычислительного процесса, где важна высокая точность, диагональная матрица весовых коэффициентов D_k изменяется по итерациям крайне незначительно. Таким образом, в методе сопряженных направлений имеется хорошее стартовое приближение $u^0 = u^{k-1}$, полученное как решение задачи (1) на предыдущей итерации алгоритма внутренних точек.

Дополнительным преимуществом метода сопряженных направлений относительно метода Халецкого является то, что формулы (3)–(6) легко адаптируются к работе с разреженными матрицами, что позволяет существенно повысить максимальную размерность решаемых задач.

Условия останова метода сопряженных направлений могут быть различными. Например, $\|Bu^t - r\| \leq \varepsilon$ при заданном $\varepsilon > 0$.

Экспериментальное исследование метода сопряженных направлений проводилось на нескольких сериях из 50 тестовых задач вида (1) различной размерности (до 300×300). Матрицы B генерировались следующим образом: сначала создавалась матрица A и диагональная матрица D , составленные из случайных целых чисел $a_{ij} \in [-5; 4]$, $d_{ii} \in [5; 15]$,

затем находилось произведение $B = ADA^T$. Векторы r формировались из случайных целых чисел $r_i \in [-500; 499]$.

В качестве начального приближения метода сопряженных направлений использовался вектор $u^0 = e$, состоящий из единиц. Таким образом, исследовался наихудший случай, когда полностью отсутствует стартовая точка, предполагающаяся близкой к решению системы (1). Критерием останова было условие $\|Bu^t - r\| \leq 0,01$. Если за 1000 итераций решение методом сопряженных направлений не было найдено, задача считалась нерешенной.

Эксперименты отличались соотношением между числом строк и столбцов в матрице A . В первом эксперименте $m = n$. Его результаты (минимальное, максимальное и среднее число итераций, среднеквадратическое отклонение числа итераций, а также количество нерешенных задач) сведены в табл.1.

Таблица 1. Число итераций, необходимое для решения задач при $m = n$

Размерность m	Число итераций			Среднеквадр. отклонение	Нерешенные задачи
	Миним.	Максим.	Среднее		
10	10	11	10,04	0,19	0
30	73	967	293,80	230,61	18
50	164	927	429,36	220,70	25
100	383	999	713,20	186,74	40
300	–	–	–	–	50

Как видно из таблицы, метод сопряженных направлений показал низкую эффективность. Причиной неудовлетворительных результатов стала высокая чувствительность метода к накоплению ошибок, а также далекое от решения начальное приближение u^0 .

Посмотрим, как метод работает в ситуации, когда $m < n$. Результаты второго эксперимента, сведенные в табл.2, справедливы для $n = 1, 2m$.

Таблица 2. Число итераций, необходимое для решения задач при $n = 1, 2m$

Размерность m	Число итераций			Среднеквадр. отклонение	Нерешенные задачи
	Миним.	Максим.	Среднее		
10	10	10	10	0	0
30	30	61	39,78	6,98	0
50	47	71	56,98	5,58	0
100	71	89	79,06	4,17	0
300	187	253	219,48	14,04	0

Увеличим разницу между m и n . Положим $n = 1, 5m$. Результаты третьего эксперимента сведем в табл.3:

Таблица 3. Число итераций, необходимое для решения задач при $n = 1, 5m$

Размерность m	Число итераций			Среднеквадр. отклонение	Нерешенные задачи
	Миним.	Максим.	Среднее		
10	10	10	10	0	0
30	24	35	29,04	2,11	0
50	34	44	39,26	1,93	0
100	47	58	51,14	1,98	0
300	55	65	60,54	2,35	0

Итоги второго и, особенно, третьего эксперимента (при увеличении различий между m и n скорость получения решения увеличивается, нерешенные задачи отсутствуют) подтверждают пригодность метода даже для решения отдельных задач вида (1). С учетом же указанных выше особенностей алгоритмов внутренних точек, а также того, что число переменных в практических задачах, как правило, вдвое и более превышает число ограничений, можно утверждать, что использование итеративных методов, в том числе, метода сопряженных направлений для решения вспомогательной задачи (1), является одним из наиболее перспективных способов ускорения алгоритмов внутренних точек.

Список литературы

- [1] Зоркальцев В.И., Филатов А.Ю. *Комбинированные алгоритмы внутренних точек для задач линейного программирования.* // Труды XIII Байкальской международной школы-семинара "Методы оптимизации и их приложения Иркутск, 2005.
- [2] Пшеничный Б.Н., Данилин Ю.М. *Численные методы в экстремальных задачах.* М.: Наука, 1975. - 319с.
- [3] Аоки М. *Введение в методы оптимизации.* М.: Наука, 1977. - 343с.
- [4] Зоркальцев В.И. *Решение систем линейных неравенств алгоритмами внутренних точек.* // "Современные методы оптимизации и их приложения к моделям энергетики": сб. науч. тр. - Новосибирск: "Наука", 2003. - с. 110-141.

THE SOLVING SYSTEMS OF LINEAR EQUATIONS WITH SYMMETRIC POSITIVE-DEFINITE MATRIX BY THE METHOD OF CONJUGATE DIRECTIONS

S.M.Perzhabinsky, A.Yu.Filatov

*Institute of energy systems named after L.A.Melent'ev, Irkutsk
e-mail: fial@isem.sei.irk.ru*

Abstract. It is investigated the possibility of solving systems of linear equations with symmetric positive-definite matrices by iterative methods, for example, by the method of conjugate directions. This problem is the most complicated one at each iteration of any interior point method for linear programming problems and for systems of linear inequalities. Moreover the specific features of interior point method make it possible to solve the problem approximately. So using methods different from the square root method can be very effective. We show here the description of the conjugate directions algorithm and the results of experimental research on some tests.

Key words: systems of linear equations, method of conjugate directions, interior point methods

СИММЕТРИЧЕСКИЕ СИСТЕМЫ И ФЕЙЕРОВСКИЕ ПРОЦЕССЫ ДЛЯ НЕСОБСТВЕННЫХ ЗАДАЧ ЛИНЕЙНОГО ПРОГРАММИРОВАНИЯ ¹

Л.Д Попов

Институт математики и механики УрО РАН, г.Екатеринбург

e-mail: popld@imm.uran.ru

Аннотация. Для численного анализа задач линейного программирования, в том числе несобственных, предложен новый класс итерационных процессов фейеровского типа, основанных на оригинальных схемах двойственности И.И.Еремина. Основное их отличие от известных алгоритмов состоит в том, что они находят обычное решение исходной задачи (и задачи, двойственной к ней) в случае ее разрешимости и некоторое ее обобщенное лексикографическое решение, если эта задача — несобственная. Имеется возможность настройки процесса на тип обобщенного решения и тип предполагаемой несобственности.

Ключевые слова: Линейное программирование, итерационные методы фейеровского типа, неразрешимость (несобственность), некорректно поставленные задачи.

Введение

Несобственными называются задачи математического программирования, для которых нарушены основные соотношения двойственности [1]. Основы качественной теории таких задач и разнообразные численные методы их анализа и коррекции можно найти в [2–4] и др. В линейном программировании (ЛП) причины возможной несобственности наиболее просты: это несовместность системы ограничений исходной задачи, или задачи, двойственной к ней, или той и другой одновременно. Соответственно различают 1-й, 2-й и 3-й род несобственности.

Несовместность системы ограничений задачи ЛП может быть как следствием неточного задания исходной информации, так и отражением вполне реальных противоречий объекта моделирования. Естественное стремление сделать процедуры разрешения этих противоречий более обоснованными породило подход к их анализу, в сооответствие с которым в исходную противоречивую модель включают некоторое число параметров, обеспечивающих те или иные возможности трансформации ее к разрешимому виду, и некоторую числовую (или векторную) функцию от этих параметров, играющую роль своеобразного измерителя величины «отклонения» трансформированной задачи от исходной. Нужно выбрать такие значения введенных параметров, которые обеспечивали бы трансформированной задаче свойство быть собственной и минимизировали бы указанное «отклонение». При этом обычные решения трансформированных задач выступают в роли обобщенных решений исходной несобственной постановки.

В работе предложен новый класс итерационных алгоритмов численного анализа и коррекции таких задач. В собственном случае алгоритмы находят обычное решение задачи, а в несобственном — некоторое ее обобщенное решение. В основе алгоритмов — теория фейеровских отображений [5]. При этом имеется возможность настройки как на тип обобщенного решения, так и на тип предполагаемой несобственности.

¹Работа выполнена при поддержке РФФИ, коды проектов НШ-792.2003.1 и 04-01-00108.

1. Постановка задачи

Рассмотрим пару взаимно двойственных (не обязательно разрешимых) задач ЛП

$$\max \{ (c, x) : Ax \leq b, x \geq 0 \}, \quad (1)$$

$$\min \{ (b, y) : A^T y \geq c, y \geq 0 \}; \quad (2)$$

здесь $c, x \in \mathbf{R}^{\hat{n}}$, $b, y \in \mathbf{R}^{\hat{m}}$ и A — матрица размерности $\hat{m} \times \hat{n}$.

Произведем некоторое разбиение ограничений прямой и двойственной задач на ряд подсистем, например, по степени их важности (приоритетности). Соответственно информационная составляющая этих задач разобьется на блоки (вертикальные и/или горизонтальные полосы) так, как показано на рис. 1.

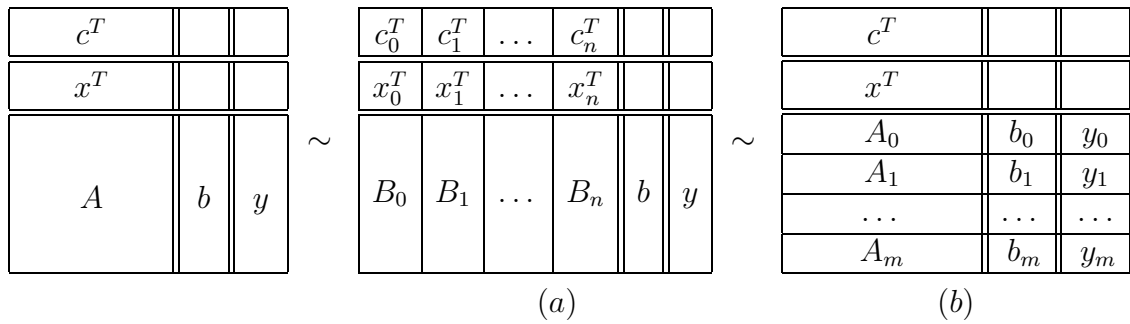


Рис. 1. Разбиение информационной составляющей задач (1), (2) на блоки: (a) — вертикальные и (b) — горизонтальные (здесь $c_i \in \mathbf{R}^{n_i}$, $x_i \in \mathbf{R}^{n_i}$, $b_j \in \mathbf{R}^{m_j}$, $y_j \in \mathbf{R}^{m_j}$, $B_i \in \mathbf{R}^{m \times n_i}$, $A_j \in \mathbf{R}^{m_j \times n}$, $\sum_{i=0}^n n_i = \hat{n}$, $\sum_{j=0}^m m_j = \hat{m}$)

С учетом введенного разбиения задачи (1), (2) можно переписать в виде

$$\max \left\{ \sum_{i=0}^n (c_i, x_i) : A_j x \leq b_j \quad (j = 0, \dots, m), \quad x \geq 0 \right\}, \quad (3)$$

$$\min \left\{ \sum_{j=0}^m (b_j, y_j) : B_i^T y \geq c_i \quad (i = 0, \dots, n), \quad y \geq 0 \right\}. \quad (4)$$

Включим в целевые функции и правые части ограничений задач (3), (4) наборы векторных параметров $\Delta c_i \in \mathbf{R}^{n_i}$ ($i = 1, \dots, n$) и $\Delta b_j \in \mathbf{R}^{m_j}$ ($j = 1, \dots, m$) по симметричной схеме так, чтобы первая из этих задач приобрела вид

$$\max \{ (c_0, x_0) + \sum_{i=1}^n (c_i - \Delta c_i, x_i) : A_j x \leq b_j + \Delta b_j \quad (j = 1, \dots, m), \quad A_0 x \leq b_0, \quad x \geq 0 \}, \quad (5)$$

а вторая — двойственный к ней вид

$$\min \{ (b_0, y_0) + \sum_{j=1}^m (b_j + \Delta b_j, y_j) : B_i^T y \geq c_i - \Delta c_i \quad (i = 1, \dots, n), \quad B_0^T y \geq c_0, \quad y \geq 0 \}. \quad (6)$$

Выбор тех или иных значений введенных выше параметров означает соответствующую корректировку правых частей ограничений исходной пары задач, а значит и их возможную трансформацию к разрешимому виду. Здесь предусмотрено также наличие в задачах так называемых «директивных» ограничений, корректировка которых не допустима. Они составляют подсистемы с нулевым индексом. В отличие от них все прочие ограничения будем называть «факультативными».

Определим формально множества разрешимости параметрических задач (5), (6):

$$\mathbb{M}_0 = \{ \Delta b = [\Delta b_1, \dots, \Delta b_m] \geq 0 : \text{ограничения (5) совместны} \},$$

$$\mathbb{N}_0 = \{ \Delta c = [\Delta c_1, \dots, \Delta c_n] \geq 0 : \text{ограничения (6) совместны} \}.$$

В силу предположения о совместности систем директивных ограничений множества \mathbb{M}_0 и \mathbb{N}_0 не пусты. Это позволяет перейти к рассмотрению следующих весьма важных и полезных в практическом отношении лексикографических (многоэтапных) оптимизационных задач, которые будем называть задачами *оптимальной коррекции* исходных несобственных постановок (1), (2):

I) найти элемент $\Delta \hat{b} \in \mathbb{M}_m$, где

$$\mathbb{M}_j = \text{Arg min} \{ \|\Delta b_j\|_j : \Delta b \in \mathbb{M}_{j-1} \} \quad (j = 1, \dots, m);$$

II) найти элемент $\Delta \hat{c} \in \mathbb{N}_n$, где

$$\mathbb{N}_i = \text{Arg min} \{ \|\Delta c_i\|_{m+i} : \Delta c \in \mathbb{N}_{i-1} \} \quad (i = 1, \dots, n);$$

здесь $\|\cdot\|_i$ ($i = 1, \dots, m+n$) — некоторый набор монотонных векторных норм в пространствах соответствующей размерности (см. рис. 1).

С содержательной стороны задачи I, II описывают естественный способ развязки «узких» мест в ограничениях задач (5), (6): всякий раз делается попытка *свести к минимуму* необходимые изменения, вносимые в ограничения подсистемы с более высоким приоритетом, и лишь затем совершается переход к корректировке ограничений из менее приоритетных подсистем. Если обе исходные системы ограничений совместны, задачи I, II имеют очевидное решение (нулевое).

2. Метод решения

Построение фейеровских процессов оптимальной коррекции исходной пары задач проведем на основе специальных схем формирования двойственности [1, 3]. В соответствие с этими схемами исходным задачам ставится в соответствие следующая пара задач выпуклого программирования:

$$\sup \left\{ (c, x) - \sum_{j=1}^m R_j \|(A_j x - b_j)^+\|_j : \|x_i\|_{m+i}^* \leq r_i \quad (i = 1, \dots, n), \right. \\ \left. A_0 x \leq b_0, \quad x \geq 0 \right\}, \quad (7)$$

$$\inf \left\{ (b, y) + \sum_{i=1}^n r_i \|(c_i - B_i^T y)^+\|_{m+i} : \|y_j\|_j^* \leq R_j \quad (j = 1, \dots, m), \right. \\ \left. B_0^T y \geq c_0, \quad y \geq 0 \right\}; \quad (8)$$

здесь $\| \cdot \|_i$ — упоминавшиеся выше монотонные векторные нормы в пространствах соответствующей размерности (см. рис. 1), $\| \cdot \|_i^*$ — сопряженные к ним нормы, $R_j > 0$ ($j = 1, \dots, m$), и $r_i > 0$ ($i = 1, \dots, n$) — штрафные параметры.

Выписанные задачи связаны классическими отношениями двойственности. В частности их оптимальные векторы и только они удовлетворяют *симметрической* системе выпуклых неравенств

$$\begin{cases} f_{Rr}(x, y) \leq 0; \\ A_0 x \leq b_0, \quad \|x_i\|_{m+i}^* \leq r_i \quad (i = 1, \dots, n), \quad x \geq 0; \\ B_0^\top y \geq c_0, \quad \|y_j\|_j^* \leq R_j \quad (j = 1, \dots, m), \quad y \geq 0; \end{cases} \quad (9)$$

где

$$f_{Rr}(x, y) = (b, y) + \sum_{i=1}^n r_i \|(c_i - B_i^\top y)^+\|_{m+i} - (c, x) + \sum_{j=1}^m R_j \|(A_j x - b_j)^+\|_j$$

— разность целевых функций этих задач.

Далее, пусть \hat{x}_{Rr} и \hat{y}_{Rr} — оптимальные векторы двойственных задач (7) и (8) соответственно. Определим

$$\Delta \hat{c}_{Rr}^i = (c_i - B_i^\top \hat{y}_{Rr})^+ \quad (i = 1, \dots, n), \quad (10)$$

$$\Delta \hat{b}_{Rr}^j = (A_j \hat{x}_{Rr} - b_j)^+ \quad (j = 1, \dots, m). \quad (11)$$

Аппроксимационный смысл найденного решения раскрывают следующие утверждения.

Теорема [1]. *Если задачи (7), (8) разрешимы и \hat{x}_{Rr} , \hat{y}_{Rr} — их оптимальные векторы, то эти же векторы оптимальны и для пары двойственных задач*

$$\max \left\{ (c_0, x_0) + \sum_{i=1}^n (c_i - \Delta \hat{c}_{Rr}^i, x_i) : A_0 x \leq b_0, \quad x \geq 0, \right. \\ \left. A_j x \leq b_j + \Delta \hat{b}_{Rr}^j \quad (j = 1, \dots, m) \right\},$$

$$\min \left\{ (b_0, y_0) + \sum_{j=1}^m (b_j + \Delta \hat{b}_{Rr}^j, y_j) : B_0^\top y \geq c_0, \quad y \geq 0, \right. \\ \left. B_i^\top y \geq c_i - \Delta \hat{c}_{Rr}^i \quad (i = 1, \dots, n) \right\},$$

где значения параметров $\Delta \hat{c}_{Rr}^i$, $\Delta \hat{b}_{Rr}^j$ определяются в силу формул (10), (11).

Теорема [6]. *Если системы директивных ограничений задач (3), (4) совместны, а нормы, участвующие в формулировках задач (7), (8), полиэдральны, то существуют такие конечные значения параметров штрафа $R_j > 0$ ($j = 1, \dots, m$), $r_i > 0$ ($i = 1, \dots, n$), при которых задачи (7), (8) одновременно разрешимы, а векторы коррекции*

$$\Delta \hat{c}_{Rr} = [\Delta \hat{c}_{Rr}^1, \dots, \Delta \hat{c}_{Rr}^n], \quad \Delta \hat{b}_{Rr} = [\Delta \hat{b}_{Rr}^1, \dots, \Delta \hat{b}_{Rr}^m],$$

определяемые по их оптимальным векторам в силу соотношений (10), (11), являются решением лексикографических задач I, II.

Правило построения системы штрафных параметров, о которых говорится в последней теореме, достаточно естественно для лексикографических задач: в целевой функции каждой из задач штрафной коэффициент при слагаемом, отвечающем более приоритетной

подсистеме ограничений, должен значительно превосходить коэффициенты при слагаемых, которые отвечают подсистемам с более низким приоритетом.

Таким образом, можно утверждать, что решение системы выпуклых неравенств (9) несет в себе важную информацию для реализации оптимальной коррекции (трансформации в собственный, разрешимый вид) несобственной задачи ЛП (1). С другой стороны имеется богатая теория фейеровских процессов решения такого рода систем (т. е. систем линейных и выпуклых неравенств) [3, 5], опираясь на которую можно формировать конкретные алгоритмы.

Итерационные фейеровские методы привлекательны простотой итерации, обычно имеющей формульный вид. Методы содержательны в ситуации эволюционирующей (изменяющейся) системы исходных данных, в этом случае вычислительный процесс может выступать в роли нестационарной модели отслеживания обчитываемого объекта (если речь идет о прикладной задаче, например, задаче о финансовых потоках). При использовании итерационных методов смягчается также проблема накопления ошибок, столь остро стоящая при применении конечных методов. Недостатком итерационных методов является обычно их медленная сходимость (в фейеровских методах — сходимость по геометрической прогрессии). Однако при больших вычислительных мощностях этот недостаток не является слишком ограничительным.

В основе фейеровских методов лежит операция проектирования на простейшие выпуклые множества, такие, например, как гиперплоскость, полупространство, линейное многообразие, шар в той или иной метрике и т. д. В частности, интересующая нас система как раз и является пересечением выпуклых множеств такой простой структуры. Итерация общего метода синтезируется в форме некоторой конечной последовательности таких проектирований (например, их суперпозиции или выпуклой комбинации). При этом простейшие проектирования выступают в роли элементарных «кирпичиков», из которых строится сам метод, что допускает большой параллелизм в выполнении тех или иных частей алгоритма.

Список литературы

- [1] И.И. Еремин, Вл.Д. Мазуров, Н.Н. Астафьев. *Несобственные задачи линейного и выпуклого программирования*. –М.: Наука, 1983. 336 с.
- [2] И.И. Еремин *Противоречивые модели оптимального планирования*. –М.: Наука, 1988. 160 с.
- [3] Eremin I.I. *Theory of Linear Optimization*. Inverse and Ill-Posed Problems Series. VSP. Utrecht, Boston, Keln, Tokyo, 2002. p.248.
- [4] Попов Л.Д. *Линейная коррекция несобственных минимаксных выпукло-вогнутых задач по максимумному критерию* – Журн. вычисл. матем. и матем. физики. 1986. т. 26, № 9. с. 1100–1110.
- [5] И.И. Еремин, Вл.Д. Мазуров. *Нестационарные процессы математического программирования*. –М.: Наука, 1979. 288 с.
- [6] Попов Л.Д. *Об одноэтапном методе решения лексикографических вариационных неравенств* – Известия вузов. Математика. № 12 (439), 1998, с. 71–81.

SYMMETRIC SYSTEMS AND NEW FEJER'S PROCESSES FOR IMPROPER LINEAR PROGRAMMING PROBLEMS

L.D. Popov

*Institute of mathematics and mechanics, UB RAS, Ekaterinburg
e-mail: popld@imm.uran.ru*

Abstract. We suggest a new family of iterative processes of Fejer type for linear programming problems which may be a priori proper as well as improper. In the last case our processes automatically generate a generalized solution in some lexicographic sense.

Key words: linear programming, iterative processes of Fejer type, improper and ill-posed problems

О ВЫЧИСЛИТЕЛЬНОЙ СЛОЖНОСТИ ЗАДАЧИ О МИНИМАЛЬНОМ КОМИТЕТЕ СИСТЕМЫ ЛИНЕЙНЫХ НЕРАВЕНСТВ¹

М.Ю. Хачай

*Институт математики и механики УрО РАН, Екатеринбург
e-mail: mkhachay@imm.uran.ru*

Аннотация. В работе исследована вычислительная сложность задачи о минимальном комитете конечной системы строгих линейных однородных неравенств (MCLE), являющейся специальным случаем более общей задачи о минимальном комитете (MC). Указанная задача тесно связана с задачей обучения слоистой нейронной сети (персептрона) и возникает, например, при обучении распознаванию образов в классе комитетных агрегированных решающих правил. Показано, что задача MСLE, как и МС — NP -трудна, приведены некоторые дополнительные ограничения, при добавлении которых задача остается труднорешаемой.

Ключевые слова: дискретная оптимизация, вычислительная сложность, NP -полнота

Введение

Несовместная система ограничений является часто встречающимся объектом при анализе трудно формализуемых данных, принятии решений, исследовании противоречивых оптимизационных моделей, обучении распознаванию образов и т.д. Активно развиваемая в последние десятилетия теория комитетных решений (см., обзор в [1]) предоставляет аппарат для исследования таких систем.

Пусть заданы множество X и набор D_1, D_2, \dots, D_m его непустых подмножеств. Рассмотрим абстрактную систему включений

$$x \in D_j, \quad (j \in \mathbb{N}_m = \{1, 2, \dots, m\}). \quad (1)$$

Система (1) не обязательно совместна, т.е. допустимо выполнение соотношения $\bigcap D_j = \emptyset$. Как обычно, назовем *комитетным решением* с q элементами системы (1) (или просто комитетом) конечную последовательность $Q = (x^1, x^2, \dots, x^q)$, удовлетворяющую условию

$$|\{i : x^i \in D_j\}| > \frac{q}{2}.$$

Комитет системы (1) с наименьшим возможным числом элементов принято называть *минимальным комитетом*. Очевидно, понятие минимального комитета является естественным дискретным обобщением понятия решения на случай несовместности системы ограничений.

Задача "МИНИМАЛЬНЫЙ КОМИТЕТ" (MC):

Заданы множество X и набор его непустых подмножеств D_1, D_2, \dots, D_m . Требуется найти комитетное решение системы (1) с наименьшим возможным q (или установить, что система не имеет комитетных решений).

¹Работа выполнена при поддержке РФФИ (проекты №№ НШ-792.2003.1, 04-01-96104 и 04-01-108) и Фонда содействия отечественной науке

Удобно вслед за [2] переформулировать задачу МС в терминах целочисленного линейного программирования. Пусть множества J_1, J_2, \dots, J_T суть индексные множества всех максимальных по включению совместных подсистем (м.с.п.) системы (1). Очевидно, система совместна тогда и только тогда, когда $T = 1$, в противном случае $1 < T < 2^m$. Определим две $m \times T$ матрицы инцидентий A и B по правилу

$$\begin{aligned} a_{ji} &= 1, \quad b_{ji} = 1, & \text{если } j \in J_i, \\ a_{ji} &= 0, \quad b_{ji} = -1, & \text{в противном случае} \end{aligned}$$

и рассмотрим задачи целочисленного линейного программирования

$$\min \{ (e, t) \mid Bt \geq f, t \in \mathbb{Z}_+^T \} \quad (2)$$

и

$$\min \left\{ \begin{array}{l} s : \alpha_s \leq 2s - 1, \quad s \in \mathbb{N}, \\ \alpha_s = \min \{ (e, t) \mid At \geq sf, \quad t \in \mathbb{Z}_+^T \} \end{array} \right\}. \quad (3)$$

Здесь e и f — векторы из единиц, принадлежащие пространствам E_T и E_m соответственно. Известна следующая

Теорема 1 ([3]). *Задачи МС, (2) и (3) разрешимы или неразрешимы одновременно. Множества оптимальных решений задач (2) и (3) изоморфно вкладываются в множество решений задачи МС (минимальных по числу элементов комитетных решений).*

Известно, что задача МС в общем случае NP -трудна. В работах [4]-[5] показано, что тем же свойством обладает и частный случай задачи, задача МСФС, в которой множество X конечно. В этих работах приведены также оценки порога аппроксимируемости задачи МСФС, аналогично известным результатам, справедливым для задачи о наименьшем покрытии множества (Set Cover).

В настоящей работе мы ограничимся рассмотрением другого частного случая задачи МС — задачи о минимальном комитете несовместной системы линейных неравенств.

1. Постановка задачи

Пусть множество X совпадает с множеством \mathbb{Q}^n векторов с рациональными коэффициентами, а подмножества D_j являются полупространствами:

$$D_j = \{x \in X \mid (a_j, x) > 0\} \quad 0 \neq a_j \in X.$$

В этом случае система (1) примет вид

$$(a_j, x) > 0 \quad (j \in \mathbb{N}_m), \quad (4)$$

а задача МС — вид задачи

"МИНИМАЛЬНЫЙ КОМИТЕТ СИСТЕМЫ ЛИНЕЙНЫХ ОДНОРОДНЫХ НЕРАВЕНСТВ" (MCLE):

Заданы натуральные числа m и $n > 1$ и векторы

$$a_1, a_2, \dots, a_m \in \mathbb{Q}^n.$$

Требуется найти комитетное решение (комитет) системы (4) с наименьшим числом элементов (или показать, что система не имеет комитетных решений)

Задача MСLE вызывает интерес как минимум по двум причинам. С одной стороны она имеет очевидное приложение при обучении распознаванию образов. К задаче MСLE приводит подход к обучению распознаванию образов, связанный с минимизацией емкости класса (VC-dimension) линейных (аффинных) комитетных решающих правил. С другой стороны, задача MСLE, в отличие от упомянутой выше задачи MСFS, не допускает традиционный подход исследования, основанный на сведении задачи к эквивалентной задаче целочисленной оптимизации (2) и изучении свойств последней. Нетрудно убедиться, что сводимость задачи MСLE к задаче (2) (или (3)) не является полиномиальной. В самом деле, для перехода к задаче (2) требуется найти все м.с.п. системы (4). Однако, как известно, задача поиска всех м.с.п. системы линейных неравенств труднорешаема. В самом деле, рассмотрим следующую задачу комбинаторной оптимизации.

Задача "НАИБОЛЬШАЯ М.С.П." (DENSEST HEMISPHERE)

Заданы числа $n > 1$ и m и векторы $a_1, a_2, \dots, a_m \in \mathbb{Q}^n$. Требуется найти наибольшую по мощности м.с.п. системы (4).

Справедлива

Теорема 2 ([6]). Задача DENSEST HEMISPHERE NP-трудна.

Таким образом, традиционная схема исследования вычислительной сложности задачи MСLE в общем случае не эффективна. Заметим, что традиционный подход, основанный на анализе задач (2)-(3) может быть тем не менее применен при исследовании следующей близкой к MСLE задачи комбинаторной оптимизации:

Задача "ОПТИМАЛЬНОЕ УЛУЧШЕНИЕ КОМИТЕТА" (COMIMP)

Заданы натуральные числа $n > 1$, m и q , векторы

$$a_1, a_2, \dots, a_m, x^1, x^2, \dots, x^q \in \mathbb{Q}^n$$

так, что последовательность $Q = (x^1, x^2, \dots, x^q)$ является комитетом системы (4). Требуется найти комитет $Q' = (y^1, y^2, \dots, y^{q'})$ с наименьшим возможным $q' \leq q$, в котором

$$y^i \in \{x^1, x^2, \dots, x^q\} \quad (i \in \mathbb{N}_{q'}).$$

В самом деле, проведем рассуждения, аналогичные проведенным во введении. Рассмотрим $m \times q$ матрицы инциденций A' и B' , элементы которых определим по правилу:

$$\begin{aligned} a'_{ji} &= 1, \quad b'_{ji} = 1, \quad \text{если } (a_j, x^i) > 0, \\ a'_{ji} &= 0, \quad b'_{ji} = -1, \quad \text{в противном случае.} \end{aligned}$$

Получим матрицы A и B путем исключения попарно доминируемых столбцов матриц A' и B' , обозначим через τ число их столбцов и рассмотрим задачи

$$\min \{ (e, t) \mid Bt \geq f, t \in \mathbb{Z}_+^\tau \} \quad (5)$$

и

$$\min \left\{ \begin{array}{l} s : \alpha_s \leq 2s - 1, s \in \mathbb{N}, \\ \alpha_s = \min \{ (e, t) \mid At \geq sf, t \in \mathbb{Z}_+^\tau \} \end{array} \right\}. \quad (6)$$

аналогичные задачам (2) и (3), соответственно. Учитывая, что проведенные построения могут быть произведены за полиномиальное от длины записи задачи СОМІМР время, сформулируем следующую теорему

Теорема 3. *Задачи СОМІМР, (5) и (6) полиномиально эквивалентны.*

2. Вычислительная сложность задачи MCLE

Основным результатом настоящей работы является

Теорема 4. *Задача MCLE NP-трудна.*

Доказательство теоремы будет получено в качестве следствия двух вспомогательных утверждений, при этом нам потребуется рассмотреть несколько дополнительных комбинаторных задач.

Задача "КОМИТЕТ ИЗ 3-Х ЭЛЕМЕНТОВ СИСТЕМЫ ЛИНЕЙНЫХ НЕРАВЕНСТВ" (3-COMLE):

Заданы натуральные числа m и $n > 1$ и векторы $a_1, a_2, \dots, a_m \in \mathbb{Q}^n$. Существует ли комитетное решение системы (4), состоящее из трех элементов?

В первом утверждении обосновывается полиномиальная сводимость задачи 3-COMLE к исследуемой задаче MCLE.

Утверждение 1. *Задача 3-COMLE сводится по Тьюрингу к задаче MCLE.*

Задача 3-COMLE, очевидно, принадлежит классу NP, поскольку проверить, является ли фиксированная последовательность $Q = (x^1, x^2, x^3)$ комитетом системы (4), можно за полиномиальное время от длины записи последней. Далее мы покажем, что она NP-полна. Для этого рассмотрим задачу

"РАСКРАСКА В 3 ЦВЕТА ОДНОРОДНОГО ГИПЕРГРАФА С МОЩНОСТЬЮ РЕБРА 3, РАСКРАШИВАЕМОГО В 2 ЦВЕТА" (3-УНС(2, 3))

Задан конечный гиперграф $\Gamma = (V, H)$, $V = \{v_1, v_2, \dots, v_n\}$, в котором условие $|h| = 3$ справедливо для каждого ребра $h \in H$. Известно, что Γ раскрашиваем в 2 цвета. Требуется определить, может ли он быть раскрашен в 3 цвета.

Известно [7], что задача 3-УНС(2, 3) NP -полна при условии, что n может принимать произвольные натуральные значения, большие трех. Ниже мы покажем ее сводимость (по Тьюрингу) к задаче 3-COMLE. Договоримся всюду в рамках этого параграфа, если явно не указано противное, полагать выполненным условие $n > 3$.

Утверждение 2. *Задача 3-COMLE NP -полна.*

Замечание 1. *В доказательстве утверждения фактически показывается, что задача MCLE (3-COMLE) остается NP -трудной (NP -полной), если ограничиться рассмотрением систем однородных неравенств, все коэффициенты которых принадлежат множеству $\{-1, 0, 1\}$, причем каждое неравенство содержит не более трех ненулевых коэффициентов*

Замечание 2. *Утверждение верно при условии, что n может принимать сколь угодно большие значения. При дополнительном ограничении на величину n задачи 3-COMLE и MCLE могут оказаться полиномиально разрешимыми. Известно (см., например, [2]), что задача MCLE при $n = 2$ обладает полиномиальным алгоритмом.*

Перепишем задачу MCLE в форме задачи распознавания свойства

Задача "КОМИТЕТ СИСТЕМЫ ЛИНЕЙНЫХ НЕРАВЕНСТВ" (COMLE)

Заданы натуральные числа $n > 1$ и m , векторы

$$a_1, a_2, \dots, a_m \in \mathbb{Q}^n$$

и нечетное число k . Существует ли комитет системы системы (4) с числом элементов, не превосходящим k ?

Учитывая утверждение теоремы 4 и знаменитой теоремы Мазурова (см., например, [1]), сформулируем

Следствие 1. *Задача COMLE полиномиально разрешима при $k \geq m$ или $k \leq 2$ и NP -полна в противном случае.*

Для задачи MCLE известен приближенный алгоритм (см., например, [8]). Как показано в этой работе, при следующих дополнительных предположениях, накладываемых на систему:

- 1). $m > n > 2$ и каждая подсистема из n неравенств совместна;
- 2). $m = 2k + n - 1$ для некоторого натурального k ,

алгоритм обладает следующими свойствами.

Теорема 5.

1. *Описанный выше алгоритм корректен и находит комитетное решение системы (4) после не более чем $\lceil \frac{k}{n-1} \rceil$ итераций.*

2. Пусть мощность наибольшей совместной подсистемы системы (4) не превосходит числа $k + (n - 1) + t$ для некоторого натурального t , тогда точность аппроксимации r алгоритма удовлетворяет соотношению

$$1 \leq r \leq \frac{2\lceil \frac{k}{n-1} \rceil + 1}{2\lceil \frac{k-t}{2t+n-1} \rceil + 1} \approx 1 + \frac{2t}{n-1}.$$

Любопытно, что для указанного алгоритма нетрудно указать нетривиальный класс систем неравенств, в котором алгоритм является точным. Такой класс включает в себя множество *равномерно распределенных (по Гейлу)* систем неравенств. Как известно, для произвольного натурального k и $m = 2k + n - 1$ система (4) равномерно распределена, если произвольный вектор $0 \neq x \in \mathbb{R}^n$ удовлетворяет не менее k неравенствам системы.

Теорема 6 ([9]). *Задача MCLE в классе равномерно распределенных систем линейных неравенств полиномиально разрешима.*

Список литературы

- [1] Mazurov V.I., Khachai M.Yu., Rybin A.I. Committee Constructions for Solving Problems of Selection, Diagnostics and Prediction. *Proceedings of the Steklov Institute of mathematics*. Suppl. 1, (2002), S67-S101.
- [2] Мазуров Вл.Д. *Метод комитетов в задачах оптимизации и классификации* - М.: Наука, 1990.
- [3] Еремин И.И., Мазуров Вл.Д. *Нестационарные процессы математического программирования*. - М.: Наука, 1979.
- [4] Khachay M.Yu. On Computational Complexity of the Minimal Committee of Finite Sets Problem // *In: Proc. of the 2nd International Workshop 'Discrete Optimization Methods in Production and Logistics'*. Omsk-Irkutsk. 2004, pp.176–179.
- [5] Хачай М.Ю. Об аппроксимационной сложности задачи о минимальном комитете // *Таврический вестник информатики и математики*. 2004, №1. С. 78–82.
- [6] Johnson D.S., Preparata F.P. The Densest Hemisphere Problem // *Theoretical Computer Science*. 1978, no. 6. pp. 93–107.
- [7] Dinur I., Regev O. and Smyth C. The hardness of 3-uniform hypergraph coloring. In: *Proc. of the 43rd Annual IEEE Symposium on Foundations of Computer Science, November 2002*.
- [8] Мазуров Вл.Д., Хачай М.Ю. Комитеты систем линейных неравенств // *Автоматика и телемеханика*. 2004. вып. 2, С. 43–54.
- [9] Khachay M.Yu. On Approximate Algorithm of a Minimal Committee of a Linear Inequalities System // *Pattern Recognition and Image Analysis*. 2003, vol. 13, no 3. pp. 459–464.

ON COMPUTATIONAL COMPLEXITY OF THE MINIMAL COMMITTEE OF LINEAR INEQUALITIES (MCLE) PROBLEM

M.Yu. Khachay

Institute of Mathematics and Mechanics, Ekaterinburg

e-mail: mkhachay@imm.uran.ru

Abstract. Computational complexity of the minimal committee of linear inequalities (MCLE) combinatorial optimization problem is studied. This problem is just a special case of the more general Minimal Committee (MC) problem. It is proven that MCLE problem is NP-hard. An approximation algorithm and some subclasses of the problem that can be solved for a polynomial time are discussed too.

Key words: discrete optimization, committee solutions, computational complexity, approximation algorithms

ВАРИАЦИОННЫЙ ПОДХОД К РЕШЕНИЮ НЕСОБСТВЕННЫХ ЗАДАЧ ЛИНЕЙНОГО ПРОГРАММИРОВАНИЯ

Н.Б. Шамрай

*Омский государственный технический университет, Омск
e-mail: nb_shamray@mail.ru*

Аннотация. В статье анализируются способы трансформации несобственных задач линейного программирования в собственные. Один из способов состоит в параметрической коррекции исходной задачи и в поиске значения параметра, обеспечивающего разрешимость задачи. Другой основан на сведении решения исходной оптимизационной задачи к решению эквивалентной системы неравенств и коррекции полученной системы. Кроме того, рассматривается вариационный подход релаксации системы линейных неравенств. Доказано свойство обратной сильной монотонности отображения соответствующего вариационного неравенства, что позволяет применить проективный метод для получения оптимальной коррекции.

Ключевые слова: несобственная задача линейного программирования, симметрическая задача, минимальная невязка, параметризованная коррекция, обобщенное решение, вариационное неравенство.

Введение

При моделировании прикладных задач возможна ситуация, когда полученная модель математического программирования противоречива. Практическая значимость рекомендаций к разрешению противоречий в таких моделях достаточно велика.

Задача математического программирования называется собственной, если она и двойственная к ней задача разрешимы и их оптимальные значения совпадают. В противном случае задача называется несобственной [1].

В данной работе рассматриваются несобственные задачи линейного программирования и исследуются некоторые способы их коррекции.

Один из способов трансформации несобственной задачи в собственную состоит в параметризованной коррекции исходной неразрешимой задачи и в поиске значения параметра, обеспечивающего ее разрешимость. Другой подход основан на сведении решения оптимизационной задачи к решению эквивалентной системы неравенств. Для несобственной задачи полученная система будет несовместной и в этом случае делается попытка ослабить ограничения системы с целью достижения совместности.

Ослаблять ограничения можно независимо друг от друга, а можно провести параметризованную коррекцию, с помощью которой напряженность некоторых наиболее важных ограничений можно смягчить, ослабляя не только их самих, но и другие неравенства.

В работе предлагается проводить коррекцию несовместной системы, используя аппарат вариационных неравенств.

1. Несобственная задача линейного программирования (НЗ ЛП)

Рассмотрим задачу линейного программирования

$$\max \{(c, x) \mid x \in M\}, \quad M = \{x \mid Ax \leq b, x \geq 0\}. \quad (1)$$

Двойственная задача к задаче (1) имеет вид

$$\min \{(b, y) \mid y \in M^*\}, \quad M^* = \{y \mid A^T y \geq c, y \geq 0\}. \quad (2)$$

Классификацию НЗ ЛП проводят по свойствам пустоты или непустоты множеств M и M^* . Здесь возможны три случая [1]:

1. $M = \emptyset, M^* \neq \emptyset$;
2. $M \neq \emptyset, M^* = \emptyset$;
3. $M = \emptyset, M^* = \emptyset$.

В зависимости от выполнения одного из условий 1) - 3) задачу (1) называют несобственной первого, второго или третьего рода соответственно.

Рассмотрим способы коррекции НЗ ЛП и дадим их сравнительную характеристику.

2. Коррекция НЗ ЛП по ее информационной составляющей

Данный способ был предложен в работе [1] и состоит в параметрической коррекции исходной задачи и нахождении оптимального значения параметра, при котором несобственная задача становится собственной.

Рассмотрим задачу (1). Этой задаче поставим в соответствие класс задач

$$\max \{(c - \Delta c, x) \mid Ax \leq b + \Delta b, x \geq 0\}, \quad (3)$$

где $\Delta c \geq 0, \Delta b \geq 0$ — параметры.

Для отыскания значений параметров Δc и Δb , которые приблизят исходную несобственную задачу к "ближайшей" разрешимой, предлагается решить две задачи квадратичного программирования

$$\min \{\|\Delta b\|^2 \mid Ax \leq b + \Delta b, x \geq 0, \Delta b \geq 0\}, \quad (4)$$

$$\min \{\|\Delta c\|^2 \mid A^T y \geq c - \Delta c, y \geq 0, \Delta c \geq 0\}. \quad (5)$$

Если Δb^* и Δc^* — оптимальные решения задач (4) и (5) соответственно, то задача

$$\max \{(c - \Delta c^*, x) \mid Ax \leq b + \Delta b^*, x \geq 0\} \quad (6)$$

— оптимально скорректированная задача для задачи (1).

Отметим, что задачи (4) и (5) являются задачами оптимальной коррекции правых частей линейных неравенств

$$Ax \leq b, \quad A^T y \geq c \quad (7)$$

в квадратичной метрике. Эти задачи более коротко могут быть записаны в виде

$$\min_{x \geq 0} \|(Ax - b)_+\|^2, \quad (8)$$

$$\min_{y \geq 0} \|(c - A^T y)_+\|^2 \quad (9)$$

соответственно.

Таким образом, вместо параметризации задачи (1) в форме (3) и решения задач (4), (5) можно рассматривать задачу коррекции системы неравенств (7).

3. Коррекция НЗ ЛП при помощи эквивалентной системы неравенств

Разрешимость или неразрешимость задачи (1), равным образом как и (2), следует из разрешимости или неразрешимости (несовместности) симметрической задачи [1]

$$Ax \leq b, \quad x \geq 0, \quad (10)$$

$$A^T y \geq c, \quad y \geq 0, \quad (11)$$

$$(c, x) \geq (b, y). \quad (12)$$

В свою очередь, несовместность системы (10)-(12) эквивалентна несовместности хотя бы одной из подсистем (10) или (11). Перепишем систему (10),(11) в виде

$$Dz + q \leq 0, \quad (13)$$

$$z \geq 0, \quad (14)$$

где

$$z = \begin{pmatrix} x \\ y \end{pmatrix}, \quad D = \begin{pmatrix} A & 0 \\ 0 & -A^T \end{pmatrix}, \quad q = \begin{pmatrix} -b \\ c \end{pmatrix}.$$

Для проверки совместности системы (13),(14) в работе [2] предлагается минимизировать невязку данной или альтернативной к ней системы.

По результатам минимизации невязки альтернативной системы можно определить совместна или нет система (13),(14).

Вектор минимальных невязок $v^* = (Dz^* + q)_+$ системы (13),(14) является результатом решения задачи квадратичного программирования

$$\min_{z \geq 0} \|(Dz + q)_+\|^2. \quad (15)$$

При этом решение задачи (15) $z^* = [x^*, y^*]$, дает решения x^* и y^* задач (8) и (9) соответственно. В результате получаем оптимально скорректированную задачу (6), где $\Delta c^* = (c - A^T y^*)_+$, $\Delta b^* = (Ax^* - b)_+$. В этом случае коррекция подхода [1] и подхода [2] дают один и тот же результат.

В любом случае при неразрешимости системы (13),(14) смягчение ее ограничений осуществляется пропорционально вектору минимальных невязок. Такой подход уместен, когда ограничения ослабляются независимо друг от друга. Но можно провести параметризованную коррекцию, с помощью которой напряженность некоторых наиболее важных ограничений можно смягчить, ослабляя не только их самих, но и другие неравенства.

4. Параметрическая коррекция систем неравенств

Предполагается, что задана некоторая положительно определенная матрица P параметрической подправки (релаксации) системы (13). Введем в рассмотрение неотрицательный вектор параметров (оценок) $v \geq 0$.

Обобщенным решением z^* системы(13) называется решение задачи [3]

$$Dz + q \leq Pv, \quad v \geq 0, \quad (16)$$

$$v^T(Dz + q) = v^T Pv, \quad (17)$$

$$v^T D = 0. \quad (18)$$

Задача (16),(17) называется параметрической линейной задачей дополнителности. Обозначим ее $PLCP(P, Dz+q)$. Если матрица P положительно определена, то обобщенное решение существует [3].

В случае совместной системы (13) в качестве обобщенного решения z^* получается решение системы в обычном смысле с нулевыми оценками $v^* = v(z^*) = 0$, и вектор невязок Pv^* системы будет нулевой.

В случае несовместной системы (13), обобщенное решение z^* имеет ненулевые оценки $v^* = v(z^*) \neq 0$, и ограничения подсистемы (13) можно ослабить пропорционально вектору невязок $Pv^* \neq 0$.

Для отыскания обобщенного решения системы (13),(14) предлагается решить задачу нахождения обобщенного решения при ограничениях [4].

Если P единичная матрица, то ограничения системы (13),(14) ослабляются пропорционально вектору параметров v^* . Тогда параметризованная коррекция эквивалентна коррекции системы согласно вектору минимальных невязок [5]. В этом случае коррекция линейной системы неравенств по подходу [2] совпадает с релаксацией системы [3].

Для несобственных задач первого и второго рода, можно ослаблять ограничения только несовместной системы, определяющей допустимую область прямой или двойственной задачи.

Принципиально другой подход решения несобственных задач состоит в том, что коррекция осуществляется итеративно, одновременно с оптимизацией исходной задачи [6]. Если исходная задача (1) является несобственной первого рода, то вводится некоторый желательный уровень θ функции цели двойственной задачи (2) и релаксируется только система

$$Ax \leq b, \quad x \geq 0, \quad c^T x \geq \theta.$$

Если исходная задача (1) является несобственной второго рода, то вводится некоторый желательный уровень θ функции цели задачи (2) и релаксируется только система

$$A^T y \geq c, \quad y \geq 0, \quad b^T y \leq \theta.$$

5. Вариационный подход к решению НЗ ЛП

Выделим в неравенствах (10),(11) совместную подсистему и обозначим через Z множество ее решений. Отметим, что Z непустое замкнутое выпуклое множество. Пусть D матрица несовместной подсистемы. Рассмотрим задачу

$$Dz + q \leq 0, \quad z \in Z. \quad (19)$$

Утверждение. Вектор z^* является обобщенным решением системы (19) тогда и только тогда, когда удовлетворяет вариационному неравенству

$$\langle D^T v(z^*), z - z^* \rangle \geq 0, \quad \forall z \in Z, \quad (20)$$

где $v(z^*)$ решение параметрической линейной задачи дополнителности $PLCP(P, Dz^* + q)$.

В силу данного утверждения можно применить проективный метод решения вариационных неравенств для нахождения обобщенного решения.

Для вариационного неравенства (20) итерационный процесс проективного метода запишется в виде

$$z^{k+1} = \pi_Z(z^k - \alpha_k D^T v^k), \quad (21)$$

где $v^k = v(z^k)$ решение задачи $PLCP(P, Dz^k + q)$.

Теорема 1. Отображение $G(z) = D^T v(z)$ вариационного неравенства (20) удовлетворяет условию обратной сильной монотонности

$$\langle D^T v(z^1) - D^T v(z^2), z^1 - z^2 \rangle \geq \tau \|D^T v(z^1) - D^T v(z^2)\|^2, \quad \forall z^1, z^2 \in Z,$$

где $v^1 = v(z^1)$, $v^2 = v(z^2)$ решения задач $PLCP(P, Dz^1 + q)$ и $PLCP(P, Dz^2 + q)$ соответственно, $\tau > 0$ — константа, если матрица P положительно определена.

Теорема 2. Если $0 < \alpha_k < 2\tau$, где $\tau > 0$ константа обратной сильной монотонности отображения $G(z) = D^T v(z)$, то итерационный процесс (21) линейно сходится к решению z^* вариационного неравенства (20).

Проведенные численные эксперименты подтверждают теоретические результаты.

Список литературы

- [1] И.И. Еремин *Теория линейной оптимизации*. Екатеринбург: УрО РАН, 1998, 248 с.
- [2] А.И.Голиков, Ю.Г.Евтушенко *Теоремы об альтернативах и их применение в численных методах* - Журнал вычислительной математики и математической физики, 2003, т. 43, № 3, с.354-375..
- [3] В.А.Булавский *Методы релаксации для систем неравенств*. Новосибирск: НГУ, 1981, 84 с.
- [4] А.В.Зыкина *Обобщенное решение при ограничениях*. - Математическое программирование: Труды XII Байкальской международной конференции "Методы оптимизации и их приложения", Иркутск, Байкал, 24 июня-1 июля 2001 года. Том 1. Иркутск: Изд-во ИСЭМ СО РАН, 2001, с.324-329.
- [5] Н.Б.Шамрай *О двух подходах к решению систем неравенств*. - Омский научный вестник, 2003, № 3(24), сентябрь, с.55-57.
- [6] А.В.Зыкина *Анализ решений в противоречивых моделях оптимального планирования*. - Доклады СО АН ВШ, 2001, № 1(3), с.11-16.

THE VARIATIONAL APPROACH TO A SOLUTION OF IMPROPER LINEAR PROGRAMMING PROBLEMS

N.B. Shamray

Omsk State Technical University, Omsk
e-mail: nb_shamray@mail.ru

Abstract. The methods of transformation of improper linear programming problems into proper ones are analysed in the article. One of the methods is to use a parametric correction of an initial problem and find a value of parameter which guarantees a solvability of a problem. Another approach is based on the reduction of the initial optimization problem to the equivalent system of inequalities and its correction in the case of infeasibility. The third approach is based on variational inequality formulation for the relaxation of a system of linear inequalities. It is proved the reverse strong monotonicity of the corresponding variational inequality which allows to apply projection methods to obtain an optimal correction.

Key words: improper problem of linear programming, symmetrical problem, minimal discrepancy, parameterized correction, generalize solution, variational inequality

ВЫПУКЛОЕ ПРОГРАММИРОВАНИЕ

О НЕКОТОРЫХ КОНЕЧНОШАГОВЫХ МЕТОДАХ ОПТИМИЗАЦИИ В ВЫПУКЛЫХ ЗАДАЧАХ ¹

А.З. Ишмухаметов, Ю.Г. Карюкина

*Вычислительный центр им. А.А.Дородницына РАН, Москва
e-mail: aleks@ccas.ru; trebla@nline.ru*

Аннотация. Для решения выпуклых конечномерных задач минимизации с ограничениями типа неравенств предлагаются методы с конечношаговыми внутренними вычислительными процедурами. Эти методы основаны на методах регуляризации, проекции градиента и двойственного метода.

Ключевые слова: оптимизация, численные методы.

В теории оптимизации, в частности, в задачах математического программирования при разработке численных методов актуальными являются вопросы их практической реализуемости, эффективности и доведения их до алгоритмов. К таким вопросам относятся разработка методов, алгоритмов без бесконечных внутренних вычислительных процедур, поиск и формулировка критериев, правил останова. Предлагаемые в данной работе методы направлены на решение этих вопросов, где в качестве внутреннего, используется регуляризованный двойственный метод. Методы построены на основе метода регуляризации, методов проекции, условного градиентов и двойственного метода [1-5]. Для них получены критерии останова, доказаны оценки скорости сходимости по функционалу, сходимость по аргументу ко множеству оптимальных элементов и к нормальному оптимальному элементу. Они в абстрактном, для конечномерных гильбертовых пространств предложены в [6,7]. Отметим, что в конечномерных задачах они имеют свои особенности, в частности, это связано с эквивалентностью слабой и сильной топологий и отсутствием аппроксимаций, которые присутствуют в бесконечномерных задачах. Предлагаемые методы эффективны для решения задач с квадратичными, выпуклыми целевыми функциями и с квадратичными функциями, задающих ограничения на допустимые элементы. В этом случае методы сводятся к последовательному решению систем линейных алгебраических уравнений.

1. В евклидовом пространстве E^n со скалярным произведением и нормой $\langle a, b \rangle = \sum_{i=1}^n a_i b_i$ и $\|a\|^2 = \sum_{i=1}^n a_i^2$ рассмотрим задачу

$$J(u) \rightarrow \inf, u \in U = \{u \in V \subset E^n : g_i(u) \leq 0, i \in I\}, I = \{1, \dots, m\}, \quad (1.1)$$

где V - выпуклое множество, $J(u)$, $g_i(u)$, $i \in I$, $u \in V$ - выпуклые, непрерывно дифференцируемые функции. Обозначим $J^* = \inf_{u \in U} J(u)$, $U^* = \{u \in U : J(u) = J^*\}$ и

$$S(u) = J(u) + \bar{g}(u), \bar{g}(u) = \sum_{i=1}^m g_i(u), u \in V.$$

Пусть $\varepsilon_N \geq 0$, $\varepsilon_N \rightarrow 0$, $N \rightarrow \infty$, а $\bar{u}_N \in U$ - приближенные решения:

$$\|\bar{u}_N - \bar{w}_N^0\| \leq \varepsilon_N, \bar{w}_N^0 = P_U(\bar{u}_N - J'(\bar{u}_N)), N = 1, 2, \dots, \quad (1.2)$$

¹Работа выполнена при финансовой поддержке РФФИ (проект 04-01-00619)

где P_U - оператор проектирования на U .

Пусть $\Omega(u)$, $u \in E^n$ - сильно выпуклая, дифференцируемая функция. Обозначим $\Omega_* = \inf_{E^n} \Omega(u)$, $u^{**} \in U^* - \Omega$ - нормальное решение: $\Omega(u^{**}) = \Omega_{**} = \inf_{U^*} \Omega(u)$ и введем регуляризирующие задачи

$$T_N(u) = J(u) + \alpha_N \Omega(u) \rightarrow \inf, u \in U, \alpha_N > 0, N = 1, 2, \dots, \quad (1.3)$$

$\alpha_N \rightarrow 0, N \rightarrow \infty$. Определим аналогичные (1.2) условия приближенности решений:

$$u_N \in U : \|u_N - w_N^0\| \leq \varepsilon_N, w_N^0 = P_U(u_N - T'_N(u_N)). \quad (1.4)$$

2. Для (1.1) определим двойственную задачу

$$\chi(\lambda) = \inf_V L(u, \lambda) \rightarrow \sup, \lambda \in \Lambda, \quad (2.1)$$

$$L(u, \lambda) = J(u) + \sum_{i=1}^m \lambda_i g_i(u), u \in V, \lambda \in \Lambda = \{\lambda_i \geq 0, i \in I\}.$$

Пусть $\chi^* = \sup_{\Lambda} \chi(\lambda)$, $\Lambda^* = \{\lambda \in \Lambda : \chi(\lambda) = \chi^*\}$ - ее решения, $\Lambda_0 = \{\lambda : \lambda_i > 0, i \in I\}$,

$\Lambda_1 = \{\lambda \in \Lambda : \chi(\lambda) > -\infty\}$, $\{\lambda\} = \min\{1, \lambda_1, \dots, \lambda_m\}$.

Определим отображение $u(\lambda) : \Lambda_0 \rightarrow V$, где $u(\lambda) : \chi(\lambda) = L(u(\lambda), \lambda), \forall \lambda \in \Lambda_0$. Нахождение $u(\lambda)$, $\lambda \in \Lambda_0$ во многих случаях является более простой задачей. В частности, при квадратичности функций $J(u)$, $g_i(u)$, $i \in I$, $V = E^n$ элементы $u(\lambda)$ - решения линейных задач: $L'_u(u, \lambda) = 0$.

Обозначим $J(\lambda) = J(u(\lambda))$, $g(\lambda) = g(u(\lambda))$, $\forall \lambda \in \Lambda_0$ и в (1.3) положим

$$T_N(u) = J(u) + \alpha_N \bar{g}(u) \rightarrow \inf, \alpha_N > 0, N = 1, 2, \dots, \alpha_N \rightarrow 0, N \rightarrow \infty. \quad (2.2)$$

Двойственные к ней задачи имеют вид

$$\chi(\lambda) = \inf_V L(u, \lambda) = L(u(\lambda), \lambda) \rightarrow \sup, \lambda \in \Lambda_N = \{\lambda_i \geq \alpha_N, i \in I\}. \quad (2.3)$$

Пусть $u_N^* \in U : T_N(u_N^*) = T_N^* = \inf_U T(u)$, $\chi_N^* = \sup_{\Lambda_N} \chi(\lambda) = \chi(\lambda_N^*)$, $\lambda_N^* \in \Lambda_N^* = \{\lambda \in \Lambda_N : \chi(\lambda) = \chi_N^*\}$ и $\lambda_N \in \Lambda_N, N = 1, 2, \dots$ - приближенные решения задач (2.3):

$$\max_{1 \leq i \leq m} |(\lambda_N - P_N(\lambda_N + g(u_N)))_i| = \max_{1 \leq i \leq m} |\max\{\alpha_N - \lambda_{Ni}; g_i(u_N)\}| \leq \varepsilon_N, \quad (2.4)$$

где P_N - оператор проектирования на $\Lambda_N : (P_N \lambda)_i = \max\{\alpha_N; \lambda_i\}$, $i \in I$, а $u_N = u_N(\lambda_N)$. Для поиска λ_N воспользуемся следующим вариантом метода проекции градиента. Пусть произвольно $\lambda_1^0 \in \Lambda_0$. Следующие $\lambda_N^k, k = 0, 1, \dots, K(N), N = 1, 2, \dots$ такие, что

$$\lambda_N^{k+1} = \lambda_N^k + \beta_N^k p_N^k, p_N^k = P_N(\lambda_N^k + g(\lambda_N^k)) - \lambda_N^k = \{\max[\alpha_N - \lambda_{Ni}^k; g_i(u_N(\lambda_N^k))], i \in I\},$$

где шаг $\beta_N^k \in (0, 1]$ вычисляется конечным алгоритмом Армийо. Этот процесс будет сходящимся и через конечное число шагов приведет к выполнению (2.4).

Предположение 1. Функция $S(u)$, $u \in V$ строго равномерно выпукла и $\bar{g}^* = \inf_V \bar{g}(u) > -\infty$.

Обозначим g -оптимальный элемент $u^{**} \in U^* : \bar{g}(u^{**}) = \bar{g}^{**} = \inf_{U^*} \bar{g}(u)$.

Теорема 1. Пусть в (1.1) выполняется условие Слейтера, предположение 1 и U ограничено. Тогда

1) значения $\lambda_N \rightarrow \Lambda^*$, $N \rightarrow \infty$, элементы $u_N \rightarrow U^*$, $N \rightarrow \infty$ и справедливы оценки

$$\alpha_N \bar{g}^* - C\varepsilon_N \leq \chi_N(\lambda_N) - J^* \leq 0, \quad \alpha_N \bar{g}^* - C\varepsilon_N \leq T_N(u_N) - J^* \leq C\varepsilon_N,$$

$$\alpha_N \bar{g}^* + \varepsilon_N \leq J_N(u_N) - J^* \leq -\alpha_N \bar{g}^* + C\varepsilon_N, \quad C = const;$$

2) в случае $\varepsilon_N/\alpha_N \rightarrow 0$, $N \rightarrow \infty$ элементы $u_N \rightarrow u^{**}$, $N \rightarrow \infty$.

Теорема 2. Пусть в (1.1) $\bar{g}(u)$, $u \in V$ строго равномерно выпукла, выполняется условие Слейтера. Тогда справедливы утверждения теоремы 1.

3. Введем следующие аппроксимации для множества (1.2):

$$U_N = \{u \in E^n : g_{Ni}(u) = g_i(u) + \alpha_N^i \Omega_i(u) \leq 0, \quad i \in I\}, \quad (3.1)$$

где $\alpha_N^i \geq \alpha_{N+1}^i > 0$, $N = 1, 2, \dots$, $\alpha_N^i \rightarrow 0$, $N \rightarrow \infty$, а $\Omega_i(u)$, $u \in E^n$, $i \in I$ - неотрицательные, сильно выпуклые и дифференцируемые функции.

Обозначим $g(u) = (g_1(u), \dots, g_m(u))$, $g_N(u) = (g_{N1}(u), \dots, g_{Nm}(u))$, P_N - оператор проектирования на U_N и определим аппроксимирующие задачи

$$J(u) \rightarrow \inf, \quad u \in U_N, \quad N = 1, 2, \dots \quad (3.2)$$

Пусть $\hat{u}_N \in U_N$, $N = 1, 2, \dots$ - ее приближенные решения:

$$\|\hat{u}_N - \hat{w}_N^0\| \leq \varepsilon_N, \quad \hat{w}_N^0 = P_N(\hat{u}_N - J'_N(\hat{u}_N)). \quad (3.3)$$

Теорема 3. Для $\hat{u}_N \in U_N$, $N = 1, 2, \dots$, удовлетворяющей в (3.1), (3.2) условиям (3.3) имеет место сходимость $\rho(\hat{u}_N; U^*) \rightarrow 0$, $N \rightarrow \infty$ и справедлива оценка по функционалу

$$0 \leq J(\hat{u}_N) - J^* \leq (D + \|J'(\hat{u}_N)\|)(\varepsilon_N + C \max_{i \in I} \alpha_N^i), \quad C = const.$$

Рассмотрим регуляризованные задачи на U_N :

$$T_N(u) = J(u) + \alpha_N \Omega(u) \rightarrow \inf, \quad u \in U_N, \quad \alpha_N > 0, \quad N = 1, 2, \dots, \quad \alpha \rightarrow 0, \quad N \rightarrow \infty. \quad (3.4)$$

Пусть $\tilde{u}_N \in U_N$, $N = 1, 2, \dots$ - ее приближенные решения:

$$\|\tilde{u}_N - \tilde{w}_N^0\| \leq \varepsilon_N, \quad \tilde{w}_N^0 = P_N(\tilde{u}_N - T'_N(\tilde{u}_N)). \quad (3.5)$$

Теорема 4. Для $\tilde{u}_N \in U_N$, $N = 1, 2, \dots$, удовлетворяющей в задачах (3.4) условиям (3.5), $\max_{i \in I} \alpha_N^i/\alpha_N \rightarrow 0$, $\varepsilon_N/\alpha_N \rightarrow 0$, $N \rightarrow \infty$, имеет место $\|\tilde{u}_N - u^{**}\| \rightarrow 0$ и

$$\alpha_N \Omega_* \leq T_N^* - J^* \leq \alpha_N \Omega_{**} + C \max_{i \in I} \alpha_N^i, \quad T_N^* = \inf_{U_N} T_N(u);$$

$$\alpha_N \Omega_* \leq T_N(\tilde{u}_N) - J^* \leq \alpha_N \Omega_{**} + C \left[\max_{i \in I} \alpha_N^i + (\varepsilon_N/\alpha_N)^{1/2} \right];$$

$$0 \leq J(\tilde{u}_N) - J^* \leq \alpha_N [\Omega_{**} - \Omega_*] + C \left[\max_{i \in I} \alpha_N^i + (\varepsilon_N/\alpha_N)^{1/2} \right], \quad C = const.$$

4. Для нахождения приближенных решений задач (3.1), (3.2), $v_N = \hat{u}_N \in U_N$ в смысле (3.3), а также решений регуляризованных задач (3.1), (3.4), $v_N = \tilde{u}_N \in U_N$ в смысле (3.5) определим итерационный, релаксационный процесс

$$v_{N,k+1} = v_{N,k} + \beta_{N,k} p_{N,k}, \quad k = 0, 1, \dots, \quad \forall v_{N,0} \in U_N, \quad (4.1)$$

где $p_{N,k} : \|p_{N,k}\| = 1$ - направление спуска, а $\beta_{N,k} > 0$ выбирается по условию минимума $T_N(v_{N,k} + \beta_{N,k} p_{N,k}) = \min_{v_{N,k} + \beta p_{N,k} \in U_N} T_N(v_{N,k} + \beta p_{N,k})$ или по алгоритму Армийо.

Пусть направление спуска

$$p_{N,k} = (w_{N,k} - v_{N,k}) \|w_{N,k} - v_{N,k}\|^{-1} \quad (4.2)$$

выбирается по условию приближенному к методу проекции градиента:

$$\|w_{N,k} - w_{N,k}^0\| \leq \varepsilon_N/2, \quad w_{N,k}^0 = P_N(v_{N,k} - T'_N(v_{N,k})). \quad (4.3)$$

Введем для $w_{N,k}$, кроме (4.3) ещё условия

$$v_{N,k} + \bar{\beta}(w_{N,k} - v_{N,k}) \in U_N, \quad \bar{\beta} \in (0, 1); \quad (4.4)$$

$$\langle T'_{N,k}(v_{N,k}), w_{N,k} - v_{N,k} \rangle \leq -\|v_{N,k} - w_{N,k}\|^2. \quad (4.5)$$

Допустим при фиксированном N для процесса (4.1) с выбором направления спуска по (4.2), (4.3), шага по условию минимума или по алгоритму Армийо, выполняются условия (4.4), (4.5). Тогда получаем существование k_N такого, что выполняется условие [7]

$$\|v_{N,k_N} - w_{N,k_N}\| \leq \varepsilon_N/2. \quad (4.6)$$

Определим v_N равенством

$$v_N = v_{N,k_N} \in U_N. \quad (4.7)$$

Теорема 5. В задаче (1.1) с аппроксимацией (3.1), (3.2) для (4.7), (4.6), где $v_{N,k}$ определяется по (4.1), (4.2), $w_{N,k}$ удовлетворяет (4.3)-(4.5), с выбором шага по минимуму или по алгоритму Армийо справедливы утверждения теоремы 3 при $\hat{u}_N = v_N$, $N = 1, 2, \dots$.

Теорема 6. Пусть в задаче (1.1) с аппроксимацией (3.1), (3.4) выполняются условия $\max_{i \in I} \alpha_N^i / \alpha_N \rightarrow 0$, $\varepsilon_N / \alpha_N \rightarrow 0$, $N \rightarrow \infty$. Тогда для (4.7), (4.6), где $v_{N,k}$ определяется по (4.1), (4.2), $w_{N,k}$ удовлетворяет (4.3)-(4.5), с выбором шага по минимуму или по алгоритму Армийо справедливы утверждения теоремы 4 при $\tilde{u}_N = v_N$, $N = 1, 2, \dots$.

Далее рассмотрим вопрос вычисления $w_{N,k}$ при $\alpha_N > 0$, $N = 1, 2, \dots$.

5. Для приближенного к $w_{N,k}^0$ вычисления элементов $w_{N,k}$ из решения задач

$$\|w - (\tilde{u}_{N,k} - T'_N(\tilde{u}_{N,k}))\|^2 \rightarrow \inf, \quad w \in U_N, \quad (5.1)$$

удовлетворяющих условиям теоремы 6, воспользуемся двойственным методом.

Для этого введем двойственные задачи

$$\chi_{N,k}(\lambda) = \inf_{u \in E^n} L_{N,k}(w, \lambda) = L_{N,k}(w_{N,k}(\lambda), \lambda) \rightarrow \sup,$$

$$L_{N,k}(w, \lambda) = \|w - (\tilde{u}_{N,k} - T'_N(\tilde{u}_{N,k}))\|^2 + \sum_{i=1}^m \lambda^i g_{N_i}(w).$$

Обозначим $\chi_{N,k}^* = \max_{\Lambda} \chi_{N,k}(\lambda)$, $\Lambda_{N,k}^* = \{\lambda \in \Lambda : \chi_{N,k}(\lambda) = \chi_{N,k}^*\}$. Для решения двойственных задач воспользуемся методом проекции градиента, аналогичному описанному в 4:

$$\lambda_{N,k,s+1} = \lambda_{N,k,s} + \sigma_{N,k,s} q_{N,k,s}, \quad \forall \lambda_{N,k,0} \in \Lambda, \quad (5.2)$$

$$q_{N,k,s} = (P_{\Lambda}(\lambda_{N,k,s} + g_{N,k}(\lambda_{N,k,s}) - \lambda_{N,k,s}) \| P_{\Lambda}(\lambda_{N,k,s} + g_{N,k}(\lambda_{N,k,s})) - \lambda_{N,k,s} \|^{-1}, \quad s = 0, 1, \dots,$$

где P_{Λ} - оператор проектирования на множество Λ , который в данном случае имеет простой вид: $P_{\Lambda}b = (\max\{0; b^1\}, \dots, \max\{0; b^m\})$, $\forall b \in E^m$, а шаг $\sigma_{N,k,s}$ определяется конечношаговым алгоритмом Армийо. Обозначим элементы $w_{N,k,s} = w_{N,k}(\lambda_{N,k,s})$. Из сходимости процесса (5.2) вытекает существование s_k , такого, что

$$\max_{1 \leq i \leq m} |P_{\Lambda}(\lambda_{N,k,s_k} + g_{N,k}(\lambda_{N,k,s_k}))^i - \lambda_{N,k,s_k}^i| \leq \varepsilon_N; \quad (5.3)$$

$$\tilde{u}_{N,k} + \bar{\beta}(w_{N,k,s_k} - \tilde{u}_{N,k}) \in U_N, \quad \bar{\beta} \in (0, 1). \quad (5.4)$$

$$\langle T'_N(\tilde{u}_{N,k}), w_{N,k,s_k} - \tilde{u}_{N,k} \rangle + \|w_{N,k,s_k} - \tilde{u}_{N,k}\|^2 < 0. \quad (5.5)$$

Тогда согласно [7] для

$$w_{N,k} = w_{N,k,s_k} \quad (5.6)$$

выполняются условия (4.3)-(4.5).

Теорема 7. Пусть в задаче (1.1) с аппроксимацией (3.1), (3.4) выполняются условия $\sup_{N,k,s} \lambda_{N,k,s} < \infty$, $\max_{i \in I} \alpha_N^i / \alpha_N \rightarrow 0$, $\varepsilon_N / \alpha_N \rightarrow 0$, $N \rightarrow \infty$. Тогда для последовательности (4.7), (4.6), где $v_{N,k}$ определяется по (4.1), (4.2), с выбором шага по минимуму или по алгоритму Армийо, а $w_{N,k}$ из условий (5.6), (5.3)-(5.5), в которых $w_{N,k,s} = w_{N,k}(\lambda_{N,k,s})$, $\lambda_{N,k,s}$ строится по (5.2) справедливы утверждения теоремы 4 при $\tilde{u}_N = v_N$, $N = 1, 2, \dots$.

В заключение отметим, что метод, аналогичный описанному в 4 и 5, строится и на основе метода условного градиента.

Список литературы

- [1] Тихонов А.Н., Арсенин В.Я. *Методы решения некорректных задач*. М.: Наука, 1986.
- [2] Васильев Ф.П. *Методы оптимизации*. М.: Факториал Пресс, 2002.
- [3] Евтушенко Ю.Г. *Методы решения экстремальных задач и их применение в системах оптимизации*. М.: Наука, 1982.
- [4] Карманов В.Г. *Математическое программирование*. М.: Физматлит, 2000.
- [5] Ишмухаметов А.З. *Методы решения задач оптимизации*. М.: Изд-во МЭИ, 1998.
- [6] Ишмухаметов А.З. *Двойственный метод решения одного класса выпуклых задач минимизации*. // ЖВМиМФ, 2000, т.40, N 7, с. 1045-1060.
- [7] Ишмухаметов А.З. *Регуляризованные приближенные методы проекции и условного градиента с конечношаговыми внутренними алгоритмами*. // ЖВМиМФ, 2003, т. 43, № 12с. 1896-1909.

A FINITE-STEP METHODS OF OPTIMIZATION IN CONVEX PROBLEMS

A.Z. Ishmukhametov, Ju.G. Karukina

Dorodnicyn Computing Center of RAS, Moscow
e-mail: aleks@ccas.ru; trebla@nline.ru

Abstract. Numerical methods with finite-step inner procedures are proposed for solving convex finite-dimensional minimization problems with inequality constraints. The methods are based on regularization, gradient projection and dual techniques.

Key words: optimization, numerical methods.

УНИВЕРСАЛЬНЫЕ ПРОСТРАНСТВА ЛИНЕЙНЫХ ОПЕРАТОРОВ И СВОЙСТВА ВЛОЖЕНИЙ В НИХ СУБДИФФЕРЕНЦИАЛОВ СУБЛИНЕЙНЫХ ОПЕРАТОРОВ

Ю.Э. Линке

*Институт динамики систем и теории управления СО РАН, Иркутск
e-mail: linke@icc.ru*

Аннотация. В статье изложены методы нахождения универсальных пространств линейных операторов, которые содержат гомеоморфные образы субдифференциалов непрерывных сублинейных операторов, определенных на сепарабельных банаховых пространствах. Доказана универсальность широкого класса пространств $L(U, C(X))$ линейных операторов, определенных на любых бесконечномерных банаховых пространствах U и любых компактах X . Изучены свойства гомеоморфного вложения субдифференциалов в эти пространства. Ранее эти результаты были известны только для бесконечномерного гильбертова сепарабельного пространства $U := \ell_2$. В докладе будет рассмотрен иллюстративный пример применения универсальных пространств в задаче линейного программирования.

Ключевые слова и выражения: универсальное пространство, сублинейный оператор, субдифференциал, многозначное отображение, непрерывный селектор.

Введение

В монографиях [1], [2], [3] показана важность применения субдифференциалов для решения экстремальных задач. В монографиях [2], [3], кроме того, построено эффективное субдифференциальное исчисление для всех функционалов и тех операторов, чьими значениями являются пространства Канторовича, то есть полные векторные решетки, в которых каждое ограниченное в смысле порядка множество имеет точную верхнюю грань. Отметим, что используемая там техника пространств Канторовича, а также методов булевозначного и инфинитезимального анализа по принципиальным соображениям применимы только к операторам, чьими пространствами значений являются пространства Канторовича. Известно также, что методы булевозначного и инфинитезимального анализа не могут быть перенесены на операторы с другими пространствами значений, в частности, на сублинейные операторы со значениями в пространствах непрерывных функций.

В связи с этим обстоятельством возникает естественный вопрос: Какую технику и какие методы можно использовать для построения эффективного субдифференциального исчисления для сублинейных операторов, значения которых представляют собой более широкий, нежели пространства Канторовича, класс упорядоченных пространств?

В статьях [4] и [5] для построения субдифференциального исчисления для указанных сублинейных операторов предложено вместо методов булевозначного и инфинитезимального анализа использовать методы многозначного анализа.

В начале прошлого века один из основателей функционального анализа М. Фреше поставил задачу обнаружения универсальных пространств, содержащих гомеоморфные образы классов банаховых или метрических пространств. Напомним здесь лишь, доказанную Банахом и Мазуром, универсальность $C([0, 1])$ относительно изометричного и изоморфного вложения в него класса сепарабельных банаховых пространств.

Основная цель данной статьи — показать, что техника универсальных пространств линейных операторов вместе с методами многозначного анализа являются удобными средствами построения эффективного субдифференциального исчисления для достаточно широких классов сублинейных операторов.

1. Постановка задачи

Пусть V — банахово пространство, а $C(X)$ — банахово пространство непрерывных вещественных функций на компакте X с sup -нормой. Через $L = L(V, C(X))$ обозначим пространство всех непрерывных линейных операторов $u : V \rightarrow C(X)$, а через $L^c = L^c(V, C(X))$ его подпространство, состоящее из компактных линейных операторов, в топологии простой сходимости [6].

Рассмотрим непрерывный сублинейный оператор $P : V \rightarrow C(X)$. Здесь сублинейность означает субаддитивность и положительную однородность отображения V в $C(X)$, т.е. для всех $v_1, v_2 \in V$ и для чисел $\lambda \geq 0$

$$P(v_1 + v_2) \leq P(v_1) + P(v_2); \quad P(\lambda v_1) = \lambda P(v_1).$$

Субдифференциал $\partial P := \{u \in L : uv \leq Pv \ (\forall v \in V)\}$. Если V — сепарабельно, то субдифференциал ∂P для всякого непрерывного сублинейного оператора P является непустым, выпуклым, замкнутым и операторно выпуклым множеством [7]. Субдифференциалы ∂P непрерывных сублинейных операторов $P : V \rightarrow C(X)$, определенных на сепарабельных банаховых пространствах V , являются основным объектом исследования, для которых требуется найти универсальное пространство.

Определение. Пространство линейных операторов $L(U, C(X))$ или пространство линейных компактных операторов $L^c(U, C(X))$ называется *универсальным*, если оно содержит гомеоморфные образы субдифференциалов ∂P непрерывных сублинейных операторов $P : V \rightarrow C(X)$, определенных на сепарабельных банаховых пространствах V .

Основные вопросы. Для каких банаховых пространств U пространство линейных операторов $L(U, C(X))$ или пространство линейных компактных операторов $L^c(U, C(X))$ для любого компакта X является универсальным? Какими свойствами (помимо непрерывности) обладают гомеоморфные вложения?

Замечание 1. В докладе на предыдущей конференции [8] доказано, что для сепарабельного гильбертова пространства ℓ_2 пространство компактных линейных операторов $L^c(\ell_2, C(X))$ универсально для любого компакта X .

Чтобы распространить полученный ранее результат на любые бесконечномерные банаховы пространства, в следующем пункте доклада установим достаточные критерии универсальности пространств линейных операторов.

2. Критерии универсальности

Прежде чем сформулировать теоремы, устанавливающие достаточные критерии универсальности пространств линейных операторов, условимся называть гомеоморфизм *аффинным*, если он сохраняет выпуклые множества, и *абсолютно аффинным*, если он сохраняет абсолютно аффинные множества. Как показывают следующие две теоремы, существование абсолютно аффинных гомеоморфизмов является достаточным критерием универсальности пространств линейных операторов.

Теорема 1. Для банахова пространства U пространство линейных операторов $L(U, C(X))$ для любого компакта X универсально, если для шара $B_{V^*}^1$ единичного радиуса с центром в нуле в сопряженном пространстве V^* любого сепарабельного банахова пространства V со слабой $(*)$ -топологией существует абсолютно аффинный гомеоморфизм этого шара в шар $B_{U^*}^1$, рассматриваемый в той же слабой $(*)$ -топологии.

Теорема 2. Для банахова пространства U пространство компактных линейных операторов $L^c(U, C(X))$ для любого компакта X универсально, если для шара $B_{V^*}^1$ единичного радиуса с центром в нуле в сопряженном пространстве V^* любого сепарабельного банахова пространства V со слабой $(*)$ -топологией существует абсолютно аффинный гомеоморфизм этого шара в шар $B_{U^*}^1$, рассматриваемый в топологии, определяемой нормой сопряженного пространства U^* .

Доказательства теорем 1 и 2 опирается на работы автора [7], [4] и [5] о представлении сублинейных операторов многозначными отображениями и о вычислении их субдифференциалов с помощью непрерывных селекторов этих многозначных отображений.

Ключевым звеном наших дальнейших построений является одна лемма [9], о существовании аффинно гомеоморфных вложений, которая была доказано авторам в связи с некоторыми задачами теории меры. Приведем её формулировку:

Лемма Вейцекера. Пусть K_1 — компактное метризуемое подмножество локально выпуклого пространства E_1 и пусть K_2 является выпуклым компактным и бесконечномерным подмножеством топологического векторного пространства E_2 , то есть линейная оболочка spK_2 множества K_2 является бесконечномерным векторным пространством. Тогда существует аффинный гомеоморфизм, действующий из замкнутой выпуклой оболочки K_1 в некоторое подмножество K_2 .

Замечание 2. Незначительно модифицируя лемму Вейцекера и ее доказательство, можно утверждать, что абсолютно аффинные гомеоморфизмы также существуют, если компакт K_2 является дополнительно абсолютно аффинным. Разумеется, что при этом вместо замкнутой выпуклой оболочки компакта K_1 следует брать замыкание его абсолютно выпуклой оболочки.

3. Основные результаты

Из теоремы 1 и модификации леммы Вейцекера вытекает важная

Теорема 3. Для любого бесконечномерного банахова пространства U пространство линейных операторов $L(U, C(X))$ универсально для каждого компакта X .

Доказательство. Возьмем для любого бесконечномерного банахова пространства U в качестве K_2 шар $B_{U^*}^1$ со слабой $(*)$ -топологией. В качестве топологического векторного пространства E_2 рассмотрим U^* с той же самой топологией. Тогда $B_{U^*}^1$ является компактным, абсолютно аффинным и бесконечномерным подмножеством в U^* . Далее в качестве компакта K_1 теперь следует взять шар $B_{V^*}^1$ любого сепарабельного банахова пространства V с той же $(*)$ -топологией. Как известно, этот шар будет тогда метризуемым и компактным. Следовательно, в силу модификации леммы Вейцекера, эти шары $B_{V^*}^1$ можно абсолютно аффинно и гомеоморфно вложить в шар $B_{U^*}^1$. Таким образом, опираясь на теорему 1 и модификацию леммы Вейцекера, завершаем доказательство теоремы 3.

Замечание 3. Фактически теорема 3 утверждает только существование гомеоморфного вложения $\varphi : \partial P \rightarrow A$, где $P : V \rightarrow C(X)$ — непрерывный сублинейный оператор, а A — некоторое подмножество из $L(U, C(X))$. Здесь, напомним, V — любое сепарабельное банахово пространство, X — фиксированный компакт. Гомеоморфизм φ , подчеркнем

особо, зависит от выбора сублинейного оператора P .

В следующих трёх теоремах изучаются свойства гомеоморфного вложения φ и свойства его образа, т.е. свойства множества A .

Прежде всего напомним определения операторной выпуклости, о которой, напомним, уже шла речь выше, когда вводились субдифференциалы и которая является естественным обобщением понятия выпуклого множества, и операторно аффинного отображения.

Множество A из $L(U, C(X))$ называется *операторно выпуклым* [7], если для всяких операторов $a_1, a_2 \in A$ оператор $\lambda \circ a_1 + (I - \lambda) \circ a_2$ принадлежит A для любого мультипликатора λ . Здесь I — тождественный оператор в $C(X)$, а мультипликатор λ — линейный оператор в $C(X)$, удовлетворяющий неравенству $0 \leq \lambda \leq I$, где 0 — нулевой оператор в $C(X)$. Пусть A и B — операторно выпуклые множества. Отображение $h : A \rightarrow B$ называется *операторно аффинным отображением*, если оно сохраняет операторно выпуклые комбинации

$$h\left(\sum_{i=1}^n \lambda_i \circ a_i\right) = \sum_{i=1}^n \lambda_i \circ h(a_i), \quad \sum_{i=1}^n \lambda_i = I.$$

Назовем множества A и B в пространствах линейных операторов *операторно аффинно гомеоморфными*, если существует гомеоморфизм h множества A на B , который является операторно аффинным отображением (h^{-1} при этом тоже операторно аффинно). Если множества A и B являются выпуклыми множествами в линейных топологических пространствах, то будем, как обычно, говорить об их аффинном гомеоморфизме, заменяя в предыдущем определении мультипликаторы на числа, а операцию суперпозицию на умножение.

Сформулируем свойства гомеоморфного вложения в следующих трех теоремах. Первая из них устанавливает свойства гомоморфного вложения для индивидуального оператора, вторая — для конечного числа сублинейных операторов и их суммы, а третья — для ограниченного семейства сублинейных операторов и ограниченного семейства чисел.

Теорема 4. *В условиях теоремы 3 можно выбрать гомеоморфное вложение $\varphi : \partial P \rightarrow A$, где $P : V \rightarrow C(X)$ — непрерывный сублинейный оператор, а A — некоторое подмножество из $L(U, C(X))$, такое что выполнены следующие свойства:*

1. Множество A выпукло и отображение φ аффинно;
2. Множество A операторно выпукло и отображение φ операторно аффинно;
3. Множество A совпадает с субдифференциалом ∂P_0 непрерывного сублинейного оператора $P_0 : U \rightarrow C(X)$ и отображение φ осуществляет операторно аффинный гомеоморфизм между ∂P и ∂P_0 .

Допустим, что нам задано конечное множество непрерывных сублинейных операторов $P_i : V \rightarrow C(X)$, где $1 \leq i \leq n$. Тогда $P := \sum_{i=1}^n P_i$, очевидно, является непрерывным сублинейным оператором. Можно ли указать гомеоморфное вложение общее для всех выбранных сублинейных операторов и их суммы? Положительный ответ дает следующая

Теорема 5. *В условиях теоремы 3 можно выбрать гомеоморфное вложение φ общее для всех непрерывных сублинейных операторов $P_i : V \rightarrow C(X)$, где $1 \leq i \leq n$ и их суммы, т.е. найдется непрерывное отображение φ , определенное на $\partial P \cup \partial P_1 \cup \partial P_2 \dots \cup \partial P_n$. При этом сужение φ на каждый субдифференциал являются гомеоморфным вложением в универсальное пространство $L(U, C(X))$, причем таким, что*

$$\varphi(\partial P) = \partial P_0, \quad \varphi(\partial P_i) = \partial P_{0i} \quad (1 \leq i \leq n)$$

для непрерывных сублинейных операторов $P_0 : U \rightarrow C(X)$, $P_{0i} : U \rightarrow C(X)$ ($1 \leq i \leq n$). Кроме того, верна следующая формула:

$$\varphi(\partial(\sum_{i=1}^n P_i)) = \sum_{i=1}^n \varphi(\partial P_i).$$

Пусть задано ограниченное (по норме) семейство \mathcal{P} непрерывных сублинейных операторов $P : V \rightarrow C(X)$, т.е. найдется константа D такая, что $\|P\| \leq D$ для всех $P \in \mathcal{P}$. Зададим также ограниченное семейство Λ неотрицательных чисел $0 \leq \lambda \leq d$. Можно ли указать гомеоморфное вложение φ , обслуживающее субдифференциалы всех сублинейных операторов λP , где λ и P выбраны из указанных выше ограниченных семейств? Положительный ответ на этот вопрос дает

Теорема 6. В условиях теоремы 3 для любого ограниченного семейства \mathcal{P} сублинейных операторов P и ограниченного семейства Λ вещественных чисел λ можно выбрать гомеоморфное вложение φ общее для всех непрерывных сублинейных операторов λP , т.е. для каждого такого оператора сужение φ на $\partial(\lambda P)$ является гомеоморфным вложением в $L(U, C(X))$, причем таким, что $\varphi(\lambda \partial P) = \lambda \varphi(\partial P)$ для непрерывных сублинейных операторов $P_0 : U \rightarrow C(X)$. Кроме того, верна следующая формула:

$$\varphi(\lambda \partial P) = \lambda \varphi(\partial P)$$

для всех $P \in \mathcal{P}$ и всех чисел $\lambda \in \Lambda$.

Список литературы

- [1] А.Д. Иоффе, В.М. Тихомиров *Теория экстремальных задач*. М.:Наука, 1974, 479 с.
- [2] А.Г. Кусраев, С.С. Кутателадзе *Субдифференциалы. Теория и приложения. Ч. 1*. Новосибирск: Изд-во Ин-та матем. СО РАН, 2002, 372 с.
- [3] А.Г. Кусраев, С.С. Кутателадзе *Субдифференциалы. Теория и приложения. Ч. 2*. Новосибирск: Изд-во Ин-та матем. СО РАН, 2003, 412 с.
- [4] Ю. Э. Линке *Представления многозначных отображений сублинейными операторами и их приложения* - Труды по геометрии и анализу. Новосибирск: Изд-во ИМ СО РАН., 2002, с. 318–349.
- [5] Ю. Э. Линке *Метод сублинейных операторов и задачи о селекторах* - Докл. РАН, 1996, т. 347, № 4. с. 446–448.
- [6] Н. Бурбаки *Топологические векторные пространства* М.:ИЛ, 1959, 410 р.
- [7] Ю.Э. Линке *Об опорных множествах сублинейных операторов* - Докл. АН СССР, 1972, т. 207, №3, с. 531–533.
- [8] Ю.Э. Линке *Универсальные пространства для субдифференциалов* - В трудах XII Байкальской международной конференции, Том 1. Иркутск: 2001, с. 102–109.
- [9] Н. Von Weizäcker *A note on infinite dimensional convex sets* - Math. Scand., 1976, v. 38, p. 321–324.

THE UNIVERSAL SPACES OF LINEAR OPERATORS AND PROPERTIES OF IMBEDDING IN THEM SUBDIFFERENTIALS OF THE SUBLINEAR OPERATORS

Yu.E. Linke

*Institute of Systems Dynamics and Control Theory SB RAS, Irkutsk
e-mail: linke@icc.ru*

Abstract. In the paper the methods of finding the universal spaces of the linear operators, which contain homeomorphic images of subdifferential continuous sublinear operators determined on separable Banach spaces are stated. The universality of a wide class of spaces $L(U, C(X))$ is proved. Here the linear operators are determined on any infinite-dimensional Banach spaces U , and $C(X)$ is the space of continuous functions defining on compact space X . Properties of the homeomorphic imbedding subdifferential in the universal space also are investigated. Earlier this result was known Only for infinite-dimensional separable Hilbert spaces $U := \ell_2$. In the report will be considered illustrative example of application the universal spaces in a task linear programming.

Key words and phrases: universal space, sublinear operator, subdifferential, set-valued mapping, continuous selector.

О ПРИМЕНЕНИИ ШТРАФНЫХ ФУНКЦИЙ ДЛЯ КОРРЕКЦИИ НЕСОБСТВЕННЫХ ЗАДАЧ ВЫПУКЛОГО ПРОГРАММИРОВАНИЯ¹

В. Д. Скарин

*Институт математики и механики УрО РАН, Екатеринбург
e-mail: skavd@imm.uran.ru*

Аннотация. Для задачи выпуклого программирования с противоречивой системой ограничений обсуждается возможность ее оптимальной коррекции с помощью метода штрафных функций. Формулируются условия, при которых решение задачи со штрафом будет аппроксимировать исходную несобственную постановку.

Ключевые слова: выпуклое программирование, несобственная задача, метод штрафных функций, лексикографическая оптимизация.

Введение

В последнее время широкое распространение получили исследования несобственных задач [1] оптимизации. Особенность таких задач состоит в том, что для них не выполняются классические соотношения двойственности в математическом программировании. Свойство несобственности тесно связано с несовместностью систем ограничений в исходной и двойственной к ней задачах.

Интерес к противоречивым моделям обусловлен как потребностями математической теории (анализ систем уравнений и неравенств, некорректные задачи, задачи идентификации, распознавания образов и др.), так и необходимостью анализа прикладных противоречивых задач, прежде всего экономических. Практика показывает, что появление несовместных моделей в сфере управления производственной деятельностью — это обычное явление. Распространенность и актуальность несобственных задач порождает острую необходимость разработки теории и методов их численного анализа. В этой связи существенную роль может играть метод штрафных функций, в частности, при обосновании теории двойственности для несобственных задач [2], а также при построении процедур оптимальной коррекции противоречивых моделей [1].

1. Постановка задачи

Рассмотрим задачу выпуклого программирования (ВП)

$$\min \{f_0(x) : x \in X\}, \quad (1)$$

где $X = \{x : f(x) \leq 0\}$, $f(x) = [f_1(x), \dots, f_m(x)]$, $f_i(x)$ — выпуклые функции, определенные на \mathbb{R}^n ($i = 0, 1, \dots, m$).

Задача (1) называется [1] несобственной задачей ВП, если для нее не выполняются соотношения

$$-\infty < f^* = L^* < +\infty,$$

¹Работа выполнена при финансовой поддержке РФФИ (проект 04-01-00108) и программы поддержки ведущих научных школ НШ-792.2003.1.

где f^* и L^* — оптимальные значения соответственно (1) и задачи, двойственной к (1), т.е. $f^* = \inf_{x \in X} f_0(x)$, $L^* = \sup_{\lambda \geq 0} \inf_x L(x, \lambda)$, где $L(x, \lambda)$ — функция Лагранжа, поставленная в соответствие (1): $L(x, \lambda) = f_0(x) + (\lambda, f(x))$, $\lambda \in \mathbb{R}_+^m$.

Предположим, что в задаче (1) $X = X_1 \cap X_2$, где $X_1 = \{x : f_i(x) \leq 0, i = \overline{1, m_1}\}$, $X_2 = \{x : f_i(x) \leq 0, i = \overline{m_1 + 1, m}\}$. Пусть $X_2 \neq \emptyset$, в то время как сведения о непустоте множества X отсутствуют.

Наряду с (1) построим следующую задачу

$$\min \{f_0(x) : \varphi(x) \leq \tilde{\varphi}, x \in X_2\}, \quad (2)$$

где $\tilde{\varphi} = \inf \{\varphi(x) : x \in X_2\}$, $\varphi(x)$ — функция внешнего штрафа за нарушение ограничений, определяющих множество X_1 . Будем считать, что

$$\varphi(x) = \omega(z(x)), \quad z(x) = [f_1^+(x), \dots, f_{m_1}^+(x)],$$

$\omega(z)$ — выпуклая функция, определенная на $\mathbb{R}_+^{m_1}$ и удовлетворяющая условиям

$$\omega(0) = 0, \quad \omega(z) = \omega(z_1, \dots, z_{m_1}) \geq \beta \sum_{i=1}^{m_1} z_i^{p_i}, \quad (3)$$

где $\beta > 0$, $p_i \geq 1$ ($i = \overline{1, m_1}$).

Заметим, что в случае $X \neq \emptyset$ из определения функции $\varphi(x)$ вытекает $\tilde{\varphi} = 0$, при этом задача (2) будет совпадать с исходной постановкой (1). Если же $X = \emptyset$, то (2) представляет из себя один из возможных способов аппроксимации задачи (1) с противоречивой системой ограничений.

2. Применение метода штрафных функций

Построим штрафную функцию

$$F(x, r) = f_0(x) + r \varphi(x), \quad r > 0,$$

и сформулируем задачу

$$\inf \{F(x, r) : x \in X_2\}. \quad (4)$$

Теорема 1.. Пусть в задаче (1) $f_0(x) \geq \gamma_0 > -\infty$ ($\forall x \in X_2$) и существуют индексы $i_1, \dots, i_q \in \overline{1, m}$ такие, что множество $S^{\gamma_1} = \{x : f_{i_t}(x) \leq \gamma_1, t = \overline{1, q}\}$ непусто и ограничено при некотором $\gamma_1 > 0$. Тогда

1. $X^* \neq \emptyset$, где X^* — множество решений задачи (2).
2. Для любого $r > 0$ задача (4) разрешима в некоторой точке $x^*(r)$.
3. При $r \nearrow \infty$ выполняются соотношения

$$f_0(x^*(r)) \nearrow \bar{f}, \quad \varphi(x^*(r)) \searrow \tilde{\varphi}, \quad \rho(x^*(r), X^*) \rightarrow 0,$$

где \bar{f} — оптимальное значение задачи (2), $\rho(x, Y) \equiv \inf_{y \in Y} \|x - y\|$.

Доказательство. Из выпуклости функций $f_i(x)$ и условий теоремы вытекает ограниченность множеств $S^\gamma = \{x : f_{i_t}(x) \leq \gamma, t = \overline{1, q}\}$ для $\gamma \geq \gamma_1$. Если $m_1 + 1 \leq i_t \leq m$ для всех $t = \overline{1, q}$, то множество X_2 ограничено. Тогда ограниченным будет и \tilde{X} —

допустимое множество задачи (2), откуда $X^* \neq \emptyset$. Поэтому будем считать, что $I \neq \emptyset$, где $I = I_q \cap \{\overline{1, m_1}\}$, $I_q = \{i_1, \dots, i_q\}$. Определим последовательность точек $x_k \in X_2$ так, чтобы $\lim_{k \rightarrow \infty} \varphi(x_k) = \tilde{\varphi}$. Очевидно, что $\tilde{\varphi} \geq 0$. Выберем число $\gamma_2 \geq \gamma_1$ из условий

$$\gamma_2 > \max \left\{ 1, \sqrt[p_0]{\frac{\tilde{\varphi}}{\beta} + 1} \right\}, \quad S^{\gamma_2} \cap X_2 \neq \emptyset \quad (p_0 = \min_{i \in I_q} p_i).$$

Возможны два случая.

а) Существует номер K такой, что $x_k \in S^{\gamma_2} \cap X_2$ для всех $k \geq K$. Тогда, очевидно, любая предельная точка x^* последовательности $\{x_k\}$ минимизирует функцию $\varphi(x)$ на множестве X_2 , т.е. $x^* \in \tilde{X}$.

б) Можно выделить подпоследовательность $\{x_{k_j}\} \subset \{x_k\}$, для которой $x_{k_j} \in X_2$, $x_{k_j} \notin S^{\gamma_2}$. Тогда в силу (3) и выбора γ_2 имеем

$$\varphi(x_{k_j}) \geq \beta \sum_{i=1}^{m_1} f_i^{+p_i}(x_{k_j}) > \beta \gamma_2^{p_0} > \tilde{\varphi} + \beta$$

для любого j . Получили противоречие с определением $\{x_k\}$.

Таким образом, $\tilde{X} \neq \emptyset$ и $\tilde{X} \subset S^{\gamma_2} \cap X_2$. Отсюда \tilde{X} — ограниченное множество и, следовательно, $X^* = \text{Arg min}_{x \in \tilde{X}} f_0(x)$ непусто.

Пусть $x^* \in X^*$. Обозначим $M_r = \{x : F(x, r) \leq F(x^*, r), x \in X_2\}$. Очевидно, $X^* \subset M_r$ ($\forall r > 0$). Поскольку для $x \in M_r$ выполняется

$$\gamma_0 + r \varphi(x) \leq F(x, r) \leq \bar{f} + r \tilde{\varphi} \quad (\forall r > 0),$$

то $M_r \subset X_r$ ($\forall r > 0$), где $X_r = \{x : \varphi(x) \leq \tilde{\varphi} + \frac{1}{r}(\bar{f} - \gamma_0), x \in X_2\}$. Из ограниченности множества \tilde{X} следует ограниченность как X_r , так и M_r ($\forall r > 0$). Поэтому в силу равенства $\min_{x \in X_2} F(x, r) = \min_{x \in M_r} F(x, r)$ задача (4) разрешима для любого $r > 0$.

Пусть $\{r_k\}$ — числовая последовательность такая, что $r_{k+1} > r_k > 0$ ($\forall k$) и $x^k = x^*(r_k)$. Из определения x^k имеем

$$\begin{aligned} f_0(x^k) + r_k \varphi(x^k) &\leq f_0(x^{k+1}) + r_k \varphi(x^{k+1}), \\ f_0(x^k) + r_{k+1} \varphi(x^k) &\geq f_0(x^{k+1}) + r_{k+1} \varphi(x^{k+1}). \end{aligned} \quad (5)$$

Вычитая в (5) второе неравенство из первого, получим

$$(r_k - r_{k+1}) [\varphi(x^k) - \varphi(x^{k+1})] \leq 0.$$

Отсюда

$$\varphi(x^k) \geq \varphi(x^{k+1}), \quad k = 1, 2, \dots \quad (6)$$

Из соотношений (5) с учетом (6) также следует

$$f_0(x^k) \leq f_0(x^{k+1}), \quad k = 1, 2, \dots \quad (7)$$

Так как $x^k \in M_{r_k} \subset X_{r_1}$, то последовательность $\{x^k\}$ при $r_k \rightarrow \infty$ имеет предельную точку \bar{x} . Тогда из неравенств (6), (7) и непрерывности функций $f_0(x)$ и $\varphi(x)$ следует

$$f_0(x^k) \nearrow f_0(\bar{x}), \quad \varphi(x^k) \searrow \varphi(\bar{x}) \quad (k \rightarrow \infty).$$

Поскольку $x^k \in M_{r_k}$, то $\varphi(x^k) \leq \tilde{\varphi} + \frac{1}{r_k}(\bar{f} - \gamma_0)$, $f_0(x^k) - \bar{f} \leq r^k(\tilde{\varphi} - \varphi(x^k))$. Поэтому $\varphi(\bar{x}) \leq \tilde{\varphi}$, $f_0(\bar{x}) = \bar{f}$, т.е. \bar{x} — решение задачи (2).

Теорема доказана.

Примером функции $\varphi = \varphi(x)$, удовлетворяющей условиям (3), может служить

$$\varphi(x) = \sum_{i=1}^{m_1} [f_i^+(x)]^{p_i}, \quad p_i \geq 1 \quad (i = \overline{1, m_1}).$$

Задача (4) является экстремальной задачей с ограничениями. Введением дополнительного штрафного слагаемого ее можно свести к задаче безусловной минимизации. Положим

$$P(x, R) = f_0(x) + r\varphi(x) + \rho\psi(x),$$

где $R = [r, \rho] > 0$, $\psi(x)$ — функция штрафа за нарушение ограничений, определяющих множество X_2 . Выберем $\psi(x)$ следующим образом:

$$\psi(x) = \sigma(t(x)), \quad t(x) = [t_1(x), \dots, t_l(x)], \quad t_j(x) = f_{m_1+j}^+ \quad (j = 1, \dots, l = m - m_1),$$

$\sigma = \sigma(t)$ — выпуклая функция, определенная на \mathbb{R}_+^l и удовлетворяющая соотношениям

$$\sigma(0) = 0, \quad \inf_{\|t\|_1=1} \frac{\partial \sigma(0)}{\partial t} = \nu > 0. \quad (8)$$

Теорема 2.. *Предположим, что для задачи (1) выполнены условия теоремы 1 и существует точка x^0 , для которой $f_i(x^0) < 0$ ($i = m_1 + 1, \dots, m$). Пусть последовательность векторов $R_k = [r_k, \rho_k] \in \mathbb{R}^2$ такова, что $r_1 \geq 1$, $r_k \nearrow \infty$, $\rho_k \geq Dr_k$, где $D = \frac{\sqrt{l}(f_0(x^0) - \gamma_0 + \varphi(x^0))}{\nu \min_{1 \leq j \leq l} |f_{m_1+j}(x^0)|}$. Тогда задача*

$$\inf_x P(x, R_k) \quad (9)$$

разрешима для любого k в некоторой точке x_k^ , при этом*

$$\begin{aligned} \psi(x_k^*) &= 0 \quad (\forall k), & f_0(x_k^*) &\nearrow \bar{f}, \\ \varphi(x_k^*) &\searrow \tilde{\varphi}, & \rho(x_k^*, X^*) &\rightarrow 0 \quad (k \rightarrow \infty). \end{aligned} \quad (10)$$

Доказательство. Согласно теореме 1 задача (4) разрешима для любого $r > 0$ в некоторой точке $x^*(r)$. Задача (4) удовлетворяет условию Слейтера, поэтому существует вектор $u(r) = [u_1^r, \dots, u_l^r] \geq 0$ такой, что пара $[x^*(r), u(r)]$ будет седловой точкой функции $L_r(x, \lambda) = F(x, r) + \sum_{j=1}^l \lambda_j f_{m_1+j}(x)$ в области $\mathbb{R}^n \times \mathbb{R}_+^l$. Из неравенства $L_r(x^*(r), u(r)) \leq L_r(x^0, u(r))$ следует

$$0 \leq u_j^r \leq \frac{F(x^0, r) - F(x^*(r), r)}{|f_{m_1+j}(x^0)|} \leq \frac{r}{|f_{m_1+j}(x^0)|} \left(\frac{f_0(x^0) - \gamma_0}{r} + \varphi(x^0) \right). \quad (11)$$

Пусть $r = r_k$. Из (11) следует $\rho_k \geq \frac{\sqrt{l}}{\nu} \max_{1 \leq j \leq l} u_j^k$, где $u^k = u(r_k)$. В этом случае $P(x, R_k)$ будет [3] точной штрафной функцией для задачи (4). Поэтому задача (9) разрешима для

любого k , при этом множества решений задач (4) и (9) совпадают. Поэтому оптимальное множество задачи (9) лежит в X_2 и, следовательно, $\psi(x_k^*) = 0$ ($\forall k$). Справедливость остальных соотношений в (10) вытекает из теоремы 1.

Теорема доказана.

Заметим, что на (1) можно смотреть как на трехэтапную задачу последовательного программирования [4] (лексикографической оптимизации [5]). На первом этапе с помощью функции $\psi(x)$ мы определяем множество X_2 , на втором — с помощью $\varphi(x)$ находится множество X (или \tilde{X} , если $X = \emptyset$), и на третьем этапе решается исходная задача (1) (или ее аппроксимация (2)). При решении задачи лексикографической оптимизации обычно переходят к минимизации единого скалярного критерия, который образуется как сумма частных критериев с некоторыми весовыми коэффициентами. При этом требуется найти зависимости между этими коэффициентами, которые обеспечивали бы сходимость результирующего процесса.

В применении к нашей ситуации итоговая задача лексикографической оптимизации примет вид

$$\inf_x [Q(x, \mu) = \mu_0 f_0(x) + \mu_1 \varphi(x) + \mu_2 \psi(x)],$$

где $\mu = [\mu_0, \mu_1, \mu_2] > 0$. Тогда аналог теоремы 2 будет выглядеть следующим образом.

Теорема 2'. *Предположим, что для задачи (1) выполнены условия теоремы 1 и существует точка x^0 , для которой $f_i(x^0) < 0$ ($i = m_1 + 1, \dots, m$). Пусть последовательность положительных векторов $\mu^k = [\mu_0^k, \mu_1^k, \mu_2^k]$ такова, что $\left\{ \beta_k = \frac{\mu_0^k}{\mu_1^k} \right\} \searrow 0$ ($k \rightarrow \infty$), $\mu_2^k = \frac{d}{\nu} \mu_1^k$, где ν — из (8), $d = \text{const}$, $d \geq d_0 = d_1^{-1}[\varphi(x^0) + \beta_0(f_0(x^0) - \gamma_0)]$, $d_1 = \min_{m_1+1 \leq i \leq m} |f_i(x^0)|$. Тогда для любого k задача*

$$\inf_x Q(x, \mu_k)$$

разрешима в некоторой точке x_k^ , при этом выполняются соотношения (10).*

Список литературы

- [1] И.И.Еремин, В.Д.Мазуров, Н.Н.Астафьев *Несобственные задачи линейного и выпуклого программирования*. М.: Наука, 1983, 336 с.
- [2] И.И.Еремин *Двойственность в линейной оптимизации*. Екатеринбург: УрО РАН, 2001, 180 с.
- [3] В.Д.Скарин *О методе штрафных функций для задач нелинейного программирования*. — Журн. вычисл. матем. и матем. физики, 1973, т. 13, № 5, с. 1186–1199.
- [4] И.И.Еремин *О задачах последовательного программирования*. — Сиб. матем. журн., 1973, т. 14, № 1, с. 53–63.
- [5] В.В.Федоров *Численные методы максимина*. М.: Наука, 1979, 280 с.

ON THE USE OF PENALTY FUNCTIONS FOR THE CORRECTION OF IMPROPER CONVEX PROGRAMMING PROBLEMS

V.D.Skarin

*Institute of Mathematics and Mechanics, Ural Branch of RAS, Ekaterinburg
e-mail: skavd@imm.uran.ru*

Abstract. An approach to optimal correction of convex programming problems with infeasible constraints systems based of using the penalty function method is considered. Some conditions which provide an approximation of the initial improper problem by the penalty function method are formulated.

Key words: convex programming, improper problem, penalty function method, lexicographic optimization

ОБ ОДНОМ АЛГОРИТМЕ ЗАДАННОЙ ТОЧНОСТИ В МЕТОДЕ БАРЬЕРНЫХ ФУНКЦИЙ

И.А. Фукин

Казанский госуниверситет, Казань

e-mail: Igor.Fukin@ksu.ru

Аннотация. В статье предлагается алгоритм решения с заданной точностью задачи выпуклого программирования, основанный на принципе аппроксимации допустимого множества. Этот принцип, использовавшийся ранее при решении задач математического программирования методами центров и штрафов, применен здесь для построения алгоритмов в методе барьерных функций.

Ключевые слова: оптимизация, выпуклая, аппроксимация, барьер, штраф, алгоритм, точность.

Введение

Как известно, метод барьерных функций [1], иногда именуемый в литературе как метод внутренних штрафов, универсален и прост в реализации, чем привлекает внимание исследователей [2]-[5] уже более 30 лет. Не ослабевает к нему интерес и в последнее время [6, 7].

В данной статье предлагается алгоритм, позволяющий находить решение с заданной по целевому функционалу точностью задачи выпуклого программирования. Оценкам точности решения в методах внутренних и внешних штрафов посвящено много работ [5], [7]-[11]. В отличие от них здесь применен принцип аппроксимации допустимого множества, предложенный Я.И. Заботиным в [12] для построения алгоритмов заданной точности в методе внешних штрафов. В качестве аппроксимации используется лебегово множество вспомогательной функции. Она строится таким образом, чтобы любая точка минимума вспомогательной функции, принадлежащая разности допустимого множества и его аппроксимации являлась решением с заданной точностью исходной задачи.

1. Постановка задачи

В данной работе задача

$$f(x) \rightarrow \min, \quad (1)$$

$$f_i(x) \leq 0, i = 1..m, \quad (2)$$

где $f(x), f_i(x), i = 1..m$ определены, непрерывны и выпуклы в n -мерном евклидовом пространстве R_n , решается с заданной по $f(x)$ точностью ε . То есть, необходимо по заданному числу $\varepsilon > 0$ найти точку $x' \in X_\varepsilon^* = \{x \in D(0) : f(x) - f^* \leq \varepsilon\}$, где $D(p) = \{x \in R_n : f_i(x) \leq p, i = 1..m\}$, $f^* = \min\{f(x), x \in D(0)\}$.

Считается, что f^* достигается, множество $D(0)$ удовлетворяет условию Слейтера и существует точка $\hat{x} \in R_n$ такая, что $f(x) < f^*$.

Потребуем также выполнение следующих условий

1. Функция $f(x)$ удовлетворяет на множестве X_ε^* условию Липшица с константой L .

2. Функция $g(x) = \max\{f_i(x), i = 1..m\}$ является (ρ, β, λ) -аппроксимируемой снизу на множестве X_ε^* .

Определение (ρ, β, λ) -аппроксимируемости введено в [13] и основано на известном ([10], с. 245) понятии ρ -регулярности ограничений. Условию 2 удовлетворяет довольно широкий класс функций. В частности, если множество X_ε^* ограничено, функции $f_i(x), i = 1..m$ выпуклы на X_ε^* , а множество $D(\lambda')$ при некотором λ' удовлетворяет условию Слейтера, то для любого $\lambda \geq \lambda'$ найдется число $\beta = \beta(\lambda) > 0$ такое, что функция $g(x)$ является (ρ, β, λ) -аппроксимируемой снизу на множестве X_ε^* .

Положим $f_p = \min\{f(x), x \in D(p)\}$.

В работе [13] было показано, что при $p < 0$ имеет место оценка $f_p - f^* \leq \frac{-pL}{\beta}$. Отсюда при $p \geq -\frac{\varepsilon\beta}{L}$ выполняется $f_p - f^* \leq \varepsilon$. Таким образом решения исходной и вспомогательной задач будут отличаться не более, чем на заданную величину ε .

На этой идее в [12]-[15] построены алгоритмы заданной точности в методе штрафов. В данной работе принцип аппроксимации допустимого множества применен для построения подобных алгоритмов в методе барьерных функций. Для этого используется выпуклая барьерная функция $B(x)$, определенная следующим образом ([5], с. 193). Пусть $bdD(0)$ - граница множества $D(0)$, $intD(0) = D(0) \setminus bdD(0)$. Всюду на $intD(0)$ функция $B(x)$ конечна, непрерывна и неотрицательна, для любой бесконечной последовательности точек $\{x_i\}$, принадлежащих $intD(0)$ и сходящихся к точке из $bdD(0)$, предел $\lim_{i \rightarrow \infty} B(x_i) = +\infty$.

2. Алгоритм с аппроксимацией допустимого множества

Пусть $A(\alpha) = \{x \in R_n : B(x) \leq \alpha\}, \alpha' = \min\{\alpha : D(p) \subset A(\alpha)\}$. Тогда $\min\{f(x), x \in A(\alpha')\} - f^* \leq f_p - f^*$.

На этой оценке основан следующий принципиальный

Алгоритм 1. Задаются требуемая точность $\varepsilon > 0$ решения задачи (1)-(2), число $p \in [-\min(\frac{\varepsilon\beta}{L}, \lambda), 0)$. Выбираются последовательность $\{\tau_k\}_{k=0}^\infty \rightarrow 0, \tau_k > 0$, число $\bar{\alpha} \geq \alpha'$. Полагается $k = 0$.

1. Находится $x_k \in \text{Argmin}\{F(x, \tau_k), x \in R_n\}$, где $F(x, \tau_k) = f(x) + \tau_k B(x)$.

2. Если $x_k \notin intA(\bar{\alpha})$, то вычисления останавливаются и точка x_k принимается в качестве ε -оптимального решения задачи (1)-(2). Иначе переход к п.1 при k замененном на $k + 1$.

Теорема 1. *Найдется такой номер N , что в последовательности $\{x_k\}$ построенной по алгоритму 1 точка $x_N \notin intA(\bar{\alpha})$. При этом $x_N \in X_\varepsilon^*$.*

Доказательство. Предположим от противного, что для любого k точка $x_k \in intA(\bar{\alpha})$. Тогда, в силу замкнутости множества $A(\bar{\alpha})$, точка $x' = \lim_{k \rightarrow \infty} x_k \in A(\bar{\alpha})$.

Из существования точки \hat{x} такой, что $f(\hat{x}) < f^*$ и выпуклости функции $f(x)$ и множества $D(0)$ следует, что $X^* \in bdD(0)$, где $X^* = \{x \in D(0) : f(x) \leq f^*\}$. По теореме сходимости метода барьерных функций (см., например, [5], с. 194) точка $x' \in X^*$ и, следовательно, $x' \in bdD(0)$.

Но $A(\bar{\alpha})$ погружено в $D(0)$, так как $D(0)$ содержит открытое множество $\{x \in R_n : g(x) < 0\}$, содержащее $A(\bar{\alpha})$. Следовательно $A(\bar{\alpha}) \cap bdD(0) = \emptyset$.

Полученное противоречие доказывает существование номера N , при котором $x_N \notin intA(\bar{\alpha})$.

Далее выберем точку $\bar{x} \in \text{Argmin}\{f(x), x \in A(\bar{\alpha})\}$. Так как $\bar{\alpha} \geq \alpha'$, то в силу выпуклости

барьерной функции $A(\alpha') \subset A(\bar{\alpha})$ и

$$f(\bar{x}) \leq \min\{f(x), x \in A(\alpha')\}. \quad (3)$$

По определению точки минимума $F(x_N, \tau_N) \leq F(\bar{x}, \tau_N)$. Отсюда $f(x_N) + \tau_N B(x_N) \leq f(\bar{x}) + \tau_N B(\bar{x})$. Для удобства преобразуем это неравенство к виду

$$f(x_N) - f(\bar{x}) \leq \bar{\tau}(B(\bar{x}) - B(x_N)).$$

Так как $B(\bar{x}) \leq \bar{\alpha}$, $B(x_N) > \bar{\alpha}$, то $B(\bar{x}) - B(x_N) \leq 0$ и $f(x_N) \leq f(\bar{x})$. Тогда $f(x_N) - f^* \leq f(\bar{x}) - f^* \leq \min\{f(x) : x \in A(\alpha')\} - f^* \leq f_p - f^* \leq \varepsilon$. Теорема доказана.

В алгоритме 1 используется величина α' , оценить которую можно лишь налагая дополнительные условия на барьерную функцию. Приведем здесь оценку для наиболее распространенной в литературе функции $B(x) = \varphi(-\frac{1}{f_1(x)}, -\frac{1}{f_2(x)}, \dots, -\frac{1}{f_m(x)})$, где $\varphi(t_1, t_2, \dots, t_m)$ - положительная возрастающая функция по каждой переменной.

Лемма 1. Верно неравенство $\varphi(-\frac{1}{p}, -\frac{1}{p}, \dots, -\frac{1}{p}) \geq \alpha'$.

Доказательство. Предположим обратное. Тогда по определению величины α' множество $D(p) \setminus A(\alpha')$ не пусто. Выберем произвольно точку $x \in D(p) \setminus A(\alpha')$. По определению множества $D(p)$ неравенство $f_i(x) \leq p$ верно для всех $i = 1..m$. Отсюда из возрастания функции $\varphi(t_1, t_2, \dots, t_m)$ и условия $x \notin A(\alpha')$ следует

$$\alpha' < B(x) = \varphi(-\frac{1}{f_1(x)}, -\frac{1}{f_2(x)}, \dots, -\frac{1}{f_m(x)}) \leq \varphi(-\frac{1}{p}, -\frac{1}{p}, \dots, -\frac{1}{p}).$$

Получили противоречие, доказывающее лемму.

Таким образом, если в алгоритме 1 на подготовительном этапе выбирать величину $\bar{\alpha} \geq \varphi(-\frac{1}{p}, -\frac{1}{p}, \dots, -\frac{1}{p})$, то требуемое неравенство $\bar{\alpha} \geq \alpha'$ будет выполняться.

Список литературы

- [1] А. Фиакко, Г. Мак-Кормик *Нелинейное программирование. Методы последовательной безусловной минимизации.* - М.:Мир, 1972, 240 с.
- [2] Ю.Г. Евтушенко *Численные методы решения задач нелинейного программирования.* - Ж.вычислит.матем.и математ. физики, 1976, т.16, №2, с.307-324.
- [3] Ф.П. Васильев, М.О. Ковач *О регуляризации некорректных экстремальных задач с использованием штрафных и барьерных функций.*- Вест.Московск.ун-та. Сер.ВМК. - М.:МГУ, 1980, №2, с.29-35.
- [4] В.Д. Скарин *О скорости сходимости метода барьерных функций.* Методы оптимизации и распознавание образов в задачах планирования, УНЦ АН СССР, 1980, с.27-36.
- [5] Ю.Г. Евтушенко *Методы решения экстремальных задач и их применение в системах оптимизации.* - М.:Наука, 1982, 432 с.
- [6] В.Д. Скарин *О методе коррекции несобственных задач выпуклого программирования, использующем функцию внутреннего штрафа.* - Алгоритмический анализ неустойчивых задач: Тез. Докл. Всерос. Науч. Конф., Екатеринбург, 26 февр.- 2 марта 2001 года. Екатеринбург:Изд-во Урал.ун-та, 2001, с. 241-242.

- [7] В.Г. Жадаан *Численные методы линейного и нелинейного программирования (вспомогательные функции в условной оптимизации)*. - Москва:Вычислительный центр им. А.А.Дородницына РАН, 2002, 160 с.
- [8] И.И. Еремин *Метод "штрафов" в выпуклом программировании*. - ДАН, 1967, т.173, №4, с.748-751.
- [9] В.Г. Жадаан *О некоторых оценках коэффициента штрафа в методах точных штрафных функций*. - Ж.вычислит.матем.и математ. физики, 1984, т.24, №8, с.1164-1171.
- [10] А.Г. Сухарев, А.В. Тимохов, В.В. Федеров *Курс методов оптимизации*. - М.:Наука, 1986, 328 с.
- [11] С.А. Казаров *Оценки близости решений экстремальных задач в методе штрафных функций*. - Изв.вузов.Математика, 1978, №9, с.40-48.
- [12] Я.И. Заботин, И.А. Фукин *Об одной модификации сдвига штрафов для задач нелинейного программирования*. // Изв.вузов.Математика, 2000, N12, с.49-54.
- [13] Я.И. Заботин, И.А. Фукин *Алгоритмы в методе штрафов с аппроксимацией допустимого множества*. - Изв.вузов. Математика, 2004, N1, с. 36-47.
- [14] И.А. Фукин *Решение задачи линейного программирования с заданной точностью методом штрафов*. - Алгебра и линейная оптимизация: Труды международного семинара, посвященного 90-летию со дня рождения С.Н. Черникова. Екатеринбург: УрО РАН, 2002, с. 309-313.
- [15] И.А. Фукин *Алгоритмы заданной точности в методе штрафов с аппроксимацией допустимого множества*. Автореф. канд. дисс. Казань, 2004, 17 с.

ON AN ALGORITHM WITH GIVEN ACCURACY IN THE METHOD OF BARRIER FUNCTIONS

I.A. Fukin

*Kazan States University, Kazan
e-mail: Igor.Fukin@ksu.ru*

Abstract. This articles proposes a convex programming problem algorithm based on the principle of the feasible set approximation. This principle used before for mathematical programming problem solution in methods of centres and penalties, is applied here for algorithm development in the method of barrier functions.

Key words: optimization, convex, approximation, barrier, penalty, algorithm, accuracy

ОБ ОДНОЙ ЗАДАЧЕ КУСОЧНО-ЛИНЕЙНОГО ВЫПУКЛОГО ПРОГРАММИРОВАНИЯ ¹

В.И.Шмырев

*Институт математики СО РАН, Новосибирск
e-mail: shvi@math.nsc.ru*

Аннотация. В статье рассматривается специальный класс оптимизационных задач, возникших при моделировании финансового лизинга. Показано, что эти задачи являются задачами кусочно-линейного выпуклого программирования. Рассматриваются два подхода к построению алгоритмов решения этих задач. Первый из них основан на сведении исходной задачи к задаче линейного программирования с дополнительными условиями комплементарности на выделенные пары переменных. Второй подход использует известную схему метода одновременного решения прямой и двойственной задач линейного программирования. Оба подхода позволяют предложить эффективные алгоритмы симплексного типа.

Ключевые слова: выпуклая оптимизация, кусочно-линейная, комплементарность, алгоритмы.

Введение

Лизинг как особая форма финансовых отношений (финансовая аренда) широко распространён в экономической практике и ему посвящается достаточно обширная научная литература. В качестве примера укажем работы [1], [3],[4]. Строгое математическое моделирование лизинга было предложено в работах [2], [5]. В этих работах модель лизинга описывается в виде задачи математического программирования с линейной целевой функцией, но нелинейными ограничениями. Показано, что нелинейные ограничения можно заменить линейными, но при этом возникают дополнительные условия комплементарности на выделенные пары переменных.

В работе [9] предложен иной подход к анализу возникшего типа задач математического программирования. Показано, что речь идёт о задаче кусочно-линейного выпуклого программирования и, тем самым, исходная задача сводится просто к задаче линейного программирования (без каких-либо условий комплементарности), хотя и ценой значительного увеличения числа ограничений: если лизинговый договор включает n периодов, то ограничений будет порядка 2^{n+1} . Тем самым возникает вопрос о применимости процедур симплексного типа для возникающего типа задач. В [9] рассмотрена возможность построения алгоритма на основе идей метода одновременного решения прямой и двойственной задачи [7], [8]. Настоящая работа содержит обзорное изложение упомянутых результатов и некоторые обобщения.

1. Модель

В лизинге участвуют два субъекта: лизингодатель (лизинговая компания) и лизингополучатель (хозяйство). Лизингодатель выполняет роль финансового посредника между

¹Работа выполнена при финансовой поддержке Российского фонда фундаментальных исследований (проект 03-01-00877) и гранта Президента РФ № НШ 80.2003.6

продавцом имущества и его покупателем (хозяйством), который является лизингополучателем.

Возможны разные схемы лизинга. Ниже речь пойдёт о той, которая рассматривалась в [2] и которая состоит в следующем.

Лизингодатель берёт кредит в банке, приобретает предмет лизинга и передаёт его во временное пользование лизингополучателю на определённых условиях.

Лизингополучатель периодически выплачивает лизингодателю в оговоренные сроки лизинговые платежи, которые должны покрыть все расходы лизингодателя, а также включают определённое вознаграждение лизингодателя ("маржу"). Лизингодатель ведёт все расчёты с банком по погашению кредита и выплачивает налоги.

Не проводя подробного описания математической формализации данной схемы лизинга (см. [2]), приведём лишь формулировку возникающей в результате задачи.

Введём следующие обозначения.

x_j — лизинговый платёж в j -й период,

c_j — коэффициенты приведения платежей к одному (начальному) моменту времени,

$v_j \geq 0$ — аналогичные коэффициенты для исчисления приведённой суммы вознаграждения лизингодателя

s — заданный уровень суммарного приведенного вознаграждения лизингодателя

t — ставка налога на прибыль

Отметим, что коэффициенты c_j и v_j подчинены условию монотонности:

$$c_{j+1} < c_j, \quad v_{j+1} < v_j.$$

Пусть $x = (x_1, \dots, x_n)$ — вектор лизинговых платежей. Прибыль в i -й период зависит от платежа x_i , а также от убытков в предшествующие периоды и характеризуется (с точностью до множителя) функцией

$$g_i(x) = x_i - b_i - \sum_{j=1}^{i-1} k_{ij} g_j^-(x). \quad (1)$$

Здесь $g_j^-(x) = \max\{0, -g_j(x)\}$ — характеризует убытки j -го периода; b_i — заданная величина (включающая амортизацию и платежи, исключаемые из налогооблагаемой базы при исчислении прибыли); $k_{ij} \geq 0$ — заданные коэффициенты, подчинённые условию:

$$k_{ij} = 0 \quad i = 1, \dots, i; \quad \sum_{i=j+1}^n k_{ij} \leq 1.$$

Заданы также величины a_i , характеризующие затраты лизингодателя в i -й период помимо налога на прибыль.

В результате получается следующая математическая формулировка задачи:

$$\sum_{j=1}^i c_j x_j \rightarrow \min! \quad (2)$$

$$\sum_{j=1}^i x_j \geq \sum_{j=1}^i a_j + t \sum_{j=1}^i g_j^+(x) \quad i = 1, \dots, n, \quad (3)$$

$$g_i(x) = x_i - b_i - \sum_{j=1}^{i-1} k_{ij} g_j^-(x), \quad i = 1, \dots, n, \quad (4)$$

$$\sum_{j=1}^i v_j (x_j - a_j - t g_j^+(x)) \geq s, \quad (5)$$

$$x_j \geq 0, \quad j = 1, \dots, n. \quad (6)$$

Здесь $g_j^+(x) = \max\{0, g_j(x)\}$ — собственно прибыль в j -й период. Для $g_j(x)$ имеем $g_j(x) = g_j^+(x) - g_j^-(x)$. Это задача математического программирования с линейной целевой функцией, но нелинейными ограничениями (3), (4), (5). Однако после введения новых неотрицательных переменных p_j, l_j формулами

$$p_j = g_j^+(x), \quad l_j = g_j^-(x)$$

имеем $g_i(x) = p_i - l_i$ и ограничения (3), (4), (5) принимают уже линейный вид. Но возникают дополнительные условия комплементарности на новые переменные:

$$p_j l_j = 0 \quad \forall j = 1, \dots, n. \quad (7)$$

В итоге задача (2)–(6) преобразуется в задачу линейного программирования с целевой функцией (2) и ограничениями

$$\sum_{j=1}^i x_j \geq \sum_{j=1}^i a_j + t \sum_{j=1}^i p_j \quad i = 1, \dots, n, \quad (3')$$

$$p_i - l_i = x_i - b_i - \sum_{j=1}^{i-1} k_{ij} l_j, \quad i = 1, \dots, n \quad (4')$$

$$\sum_{j=1}^n v_j (x_j - a_j - t p_j) \geq s \quad (5')$$

$$x_j, p_j, l_j \geq 0 \quad j = 1 \dots, n. \quad (6')$$

И, кроме того, должно выполняться условие комплементарности (7).

Как уже отмечалось выше, основной результат, полученный в работах [2], [5], состоит в том, что решение полученной задачи линейного программирования без учёта ограничений (7) даёт оптимальные значения лизинговых платежей для исходной задачи (2)–(6).

Таким образом, проблема оптимизации лизинговых платежей свелась к задаче линейного программирования. Если привести систему ограничений этой задачи к каноническому виду, когда все переменные должны быть неотрицательны, а ограничения общего вида являются уравнениями, то, как легко видеть, будем иметь $2n + 1$ уравнений и $4n + 1$ переменных.

2. Некоторые обобщения

Приведенная схема с использованием комплементарных переменных может быть применена к более общим задачам.

Введем вспомогательные переменные z_i , сводя неравенства (3') к равенствам, а также величины $w_j = v_j - v_{j+1}$ при $j = 1, \dots, n-1$, $w_n = v_n$. Это позволяет преобразовать неравенство (5') к виду

$$\sum_{j=1}^n w_j z_j \geq s.$$

В результате получаем следующую краткую форму задачи:

$$\begin{aligned} (c, x) &\rightarrow \min! \\ \begin{array}{r} Tp \\ p \end{array} \quad \begin{array}{r} -z \\ +Kl \end{array} \quad \begin{array}{r} +Fx \\ +Gx \end{array} &= \begin{array}{r} g \\ h \end{array}, \\ (w, z) &\geq s, \\ p, l, z, x &\geq 0, \\ (p, l) &= 0. \end{aligned}$$

Здесь $w, c, p, l, z, x \in R_+^n$; T, K, F, G — матрицы $n \times n$. При этом G — единичная матрица,

$$F = \begin{pmatrix} 1 & 0 & \dots & 0 \\ 1 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & \dots & 1 \end{pmatrix}, \quad K = \begin{pmatrix} -1 & 0 & \dots & 0 \\ k_{21} & -1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ k_{n1} & k_{n2} & \dots & -1 \end{pmatrix}$$

и $T = -tF$.

Анализ доказательства основного результата о несущественности условия комплементарности показывает, что он остается в силе при замене условия $wz \geq s$ более общим условием $Wz \geq S$, где W — неотрицательная матрица $m \times n$, $S \in R^m$. Можно также ослабить требования на матрицу T , заменяя ее матрицей

$$T = \begin{pmatrix} -t_{11} & 0 & \dots & 0 \\ -t_{21} & -t_{22} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ -t_{n1} & -t_{n2} & \dots & -t_{nn} \end{pmatrix},$$

где элементы t_{ij} удовлетворяют условию монотонности в строках:

$$t_{ij} \geq t_{ik} \text{ при } k > j.$$

Кроме того, элементы матриц F и G могут быть произвольными.

3. Сведение к задаче линейного программирования

Упростим несколько постановку задачи, считая, что выплаты вознаграждения лизингодателю уже фиксированы по периодам, т.е. содержатся в величинах a_i . Тем самым устраняется ограничение (5). Это допущение не является существенным и может быть легко учтено, см. [9].

Заметим, что функции $g_j(x)$, $j = 1, \dots, n$ получают последовательно рекуррентным образом в соответствии с формулой (1).

Подставляя функции $g_j(x)$, $j = 1, \dots, n$ в неравенства (3), придадим этим условиям вид:

$$\sum_{j=1}^i x_j \geq f_i(x), \quad i = 1, \dots, n, \quad (8)$$

где $f_i(x) = \sum_{j=1}^i a_j + t \sum_{j=1}^i g_j^+(x)$.

Утверждение. Функции $f_i(x)$ являются выпуклыми.

Доказательство этого утверждения основано на том факте, что введенные функции f_i и g_i являются кусочно-линейными и участки их линейности совпадают с множествами знакопостоянства значений функций g_i . Для их описания вводятся кортежи знаков $\sigma = (\sigma_1, \dots, \sigma_n)$, где $\sigma_i \in \{+, -\}$. Тогда участки линейности — это множества Ω_σ , определяемые формулой

$$\Omega_\sigma = \left\{ x \in R^n \mid g_i(x) \begin{cases} \geq 0, & \sigma_i = +, \\ \leq 0, & \sigma_i = -, \end{cases} i = 1, \dots, n \right\}$$

при всевозможных кортежах σ .

Несложно убедиться, что все множества Ω_σ являются многогранными конусами с общей вершиной в точке $b = (b_1, \dots, b_n)$.

Обозначим через $g_j^\sigma(x)$ линейную функцию, совпадающую с $g_j(x)$ на конусе Ω_σ :

$$g_j(x) = g_j^\sigma(x), \quad x \in \Omega_\sigma.$$

Аналогично через $f_i^\sigma(x)$ будем обозначать линейную функцию, совпадающую с $f_i(x)$ на Ω_σ .

Сопоставим функции $f_i(x)$ кусочно-постоянное многозначное отображение, определяемое формулой

$$F_i(x) = \text{co}\{\nabla f_i^\sigma \mid x \in \Omega_\sigma\}. \quad (9)$$

Доказательство выпуклости функции f_i сводится к доказательство локальной монотонности этого отображения [6]. Из приведенного утверждения вытекает следующее представление для функций $f_i(x)$:

$$f_i(x) = \max_{\sigma \in S_i} f_i^\sigma(x),$$

где S_i — совокупность 2^i кортежей $\sigma = (\sigma_1, \dots, \sigma_i)$, $\sigma_k \in \{+, -\}$, а $f_i^\sigma(x)$ — линейная функция, совпадающая с $f_i(x)$ на конусе $\Omega_\sigma \subset R^n$, задаваемого в соответствии с σ системой условий знакопостоянства функций $g_1(x), \dots, g_i(x)$. Это означает, что фигурирующее в описании задачи лизинга нелинейное неравенство

$$\sum_{j=1}^i x_j \geq f_i(x)$$

можно заменить системой 2^i линейных неравенств

$$\sum_{j=1}^i x_j \geq f_i^\sigma(x), \quad \sigma \in S_i \quad (10)$$

Таким образом, без ограничения по "марже" (условие (5)) задача оптимизации лизинговых платежей получает эквивалентную формулировку в виде задачи линейного программирования. Несложно показать, что наличие ограничения (5) незначительно меняет ситуацию.

Результатом этих рассмотрений является

Теорема. Задача оптимизации лизинговых платежей является задачей линейного программирования.

4. Использование схемы одновременного решения прямой и двойственной задач

В связи с полученным сведением задачи оптимизации лизинговых платежей к обычной задаче линейного программирования представляет интерес вопрос о применимости в рассматриваемом случае схем последовательного улучшения (симплекс - метода). Осложняющим обстоятельством для непосредственного использования стандартных процедур является большое число ограничений возникающей задачи — порядка 2^{n+1} . Дополнительную трудность привносит тот факт, что система ограничений задачи не удовлетворяет условию невырожденности. К тому же имеющаяся вырожденность носит структурный характер и не устраняется малыми вариациями исходных данных (т.е. величин a_j, b_j).

Однако отмеченные преграды устраняются, если учесть тот факт, что ограничения задачи генерируются рекуррентным образом, а для преодоления указанной вырожденности воспользоваться методом одновременного решения прямой и двойственной задачи [7] (методом корректировки множителей Л.В. Канторовича [8]), применяя его к двойственной задаче. Так как метод одновременного решения прямой и двойственной задачи является вариантом процедуры улучшения двойственного вектора, то, применяя этот метод к двойственной задаче, получим процедуру улучшения допустимого решения в прямой задаче, т.е. процедуру улучшения лизинговых платежей. Детальная конкретизация возникающего алгоритма приведена в [9].

Список литературы

- [1] Прилуцкий Л. *Помощь в расчете лизинговых рент* - Лизинг-курьер, 1999, № 6(6).
- [2] Шмырев В.И., Осадчий М.С. *Задача оптимизации лизинговых платежей*. - Сиб. журн. индустр. математики, 2001, т.IV, № 2(8), с. 205-211.
- [3] Штельмах В. *Расчет лизинговых платежей исходя из потока денежных средств. Плюсы и минусы* - Лизинг-курьер, 2000, №1(7).
- [4] Чекмарева Е.Н. *Лизинговый бизнес* - М.: Экономика, 1994.
- [5] Осадчий М.С., Шмырев В.И. *Оптимизация лизинговых платежей* - Экономика и мат. методы, 2002, т.38, №2, с.111-117.

- [6] Шмырев В.И. *Об отыскании неподвижных точек кусочно-постоянных монотонных отображений в R^n* - Докл.АН СССР, 1981. т.259, №2, с.299-301.
- [7] Данциг Дж., Форд Л., Фулкерсон Д. *Алгоритм для одновременного решения прямой и двойственной задач линейного программирования* - Линейные неравенства и смежные вопросы. М: Изд-во иностр. лит., 1959, с.277-286.
- [8] Канторович Л.В. *Математические методы организации и планирования производства* Л: Изд-во ЛГУ, 1939.
- [9] Шмырев В.И., Сафронова И.А. *Новый метод для задачи оптимизации лизинговых платежей* - Сиб. журн. индустр. математики, 2004, т. VII, №4(20), с.148-162.

ONE PIECEWISE LINEAR CONVEX PROGRAMING PROBLEM

V.I.Shmyrev

Sobolev Institute of Mathematics, Novosibirsk
e-mail: shvi@math.nsc.ru

Abstract. A special class of optimization problems arising by modeling a leasing is investigated. We consider two approach for solving these problems. The first approach leads to linear programming with additional condition of complementarity. The second approach reduces the initial problem to a proper one of linear programming, but with large number of inequalities. The effective algorithms like the simplex-method for these problems are proposed.

Key words: leasing, programming, convex, linear, complementarity, algorithm

НЕЛИНЕЙНОЕ ПРОГРАММИРОВАНИЕ

ОБ ОДНОМ КЛАССЕ МИНИМАКСНЫХ ЗАДАЧ¹

А.С. Апарцин

Институт систем энергетики им. Л.А. Мелентьева СО РАН, Иркутск
e-mail: apartsyn@isem.sei.irk.ru

Аннотация. Решены специальные минимаксные задачи, связанные с построением математической модели нелинейной динамической системы типа вход-выход в виде полинома Вольтерра.

Ключевые слова: нелинейная динамическая система, идентификация, оптимизация, полином Вольтерра, product integration method.

1. При построении математической модели нелинейной динамической системы типа вход-выход в виде полинома Вольтерра N -го порядка, дающего представление отклика $y(t)$ на входное возмущение $x(t)$ в форме

$$y(t) = \sum_{m=1}^N \int_0^t \cdots \int_0^t K_m(s_1, \dots, s_m) \prod_{i=1}^m x(t - s_i) ds_i, \quad t \in [0, T] \quad (1)$$

((1) соответствует случаю, когда $x(t)$ скалярна, а система стационарна, т.е. ядра Вольтерра K_m не зависят явно от времени; при этом из скалярности $x(t)$ вытекает симметрия K_m по всем переменным), центральной является проблема идентификации ядер. Это — типичная обратная задача, поскольку для восстановления K_m , по существу, приходится решать m -мерное интегральное уравнение I рода, ядро которого образуют те или иные семейства тестовых входных сигналов, а правую часть — соответствующие отклики динамической системы.

В то же время для целей моделирования реакции системы на внешнее возмущение $x(t)$ знание K_m , вообще говоря, избыточно. Достаточно уметь подсчитывать многомерные свертки, входящие в (1). Если для их приближенного вычисления применить product integration method (см., например, [1]), согласно которому при выбранном шаге сетки h в одномерном случае

$$\int_0^{ih} K(s)x(t-s)ds \approx \sum_{j=1}^i x_{i-j+\frac{1}{2}} \int_{(j-1)h}^{jh} K(s)ds, \quad i = \overline{1, n}, \quad nh = T \quad (2)$$

(этот метод особенно эффективен, если $K(t)$ — сильно осциллирующая функция), то дело сводится к идентификации не самих ядер, а соответствующих интегралов от них.

Пусть, к примеру, в (1) $N = 2$ и требуется построить квадратичный полином Вольтерра

$$y_{sq}(t) = \int_0^t K_1(s)x(t-s)ds + \int_0^t \int_0^t K_2(s_1, s_2)x(t-s_1)x(t-s_2)ds_1ds_2, \quad (3)$$

¹Работа выполнена при поддержке гранта РФФИ 05-01-00336, гранта НАТО NR RIG 9812876

аппроксимирующий тестовую (эталонную) динамическую систему

$$y_{et}(t) = \sum_{m=1}^3 \frac{1}{m!} \left(\int_0^t x(s) ds \right)^m, \quad t \in [0, T] \quad (4)$$

((4) образована первыми тремя слагаемыми (1) с $K_m = \frac{1}{m!}$).

Предположим, симметричное ядро K_2 в (3) представимо в форме $K_2(s_1, s_2) = \psi(s_1)\psi(s_2)$ с некоторой $\psi(t) \in C_{[0, T]}$, тогда

$$y_{sq}(t) = \int_0^t K_1(s)x(t-s)ds + \left(\int_0^t \psi(s)x(t-s)ds \right)^2, \quad t \in [0, T], \quad (5)$$

а сеточный аналог (5), построенный на базе квадратуры (2), имеет вид

$$y_{sq}^h(t_i) = \sum_{j=1}^i m_j x_{i-j+\frac{1}{2}} + \left(\sum_{j=1}^i \ell_j x_{i-j+\frac{1}{2}} \right)^2, \quad i = \overline{1, n}, \quad nh = T, \quad (5^h)$$

где

$$m_j = \int_{(j-1)h}^{jh} K_1(s)ds; \quad \ell_j = \int_{(j-1)h}^{jh} \psi(s)ds, \quad j = \overline{1, n}. \quad (6)$$

Для идентификации $2n$ неизвестных (6) достаточно использовать отклики эталонной модели (4) на 2 тестовых сигнала

$$x^{\alpha_i}(t) = \alpha_i e(t), \quad i = 1, 2, \quad (7)$$

где $e(t)$ — функция Хевисайда, $\alpha_1 \neq \alpha_2 \neq 0$ — вещественные числа.

Так как $y_{et}^{\alpha_i}(t) = \alpha_i t + \frac{\alpha_i^2}{2} t^2 + \frac{\alpha_i^3}{6} t^3, i = 1, 2$, то нетрудно подсчитать, что

$$m_j = t_j - t_{j-1} - \frac{\alpha_1 \alpha_2}{6} (t_j^3 - t_{j-1}^3),$$

$$\ell_j = \sqrt{\frac{t_j^2}{2} + \frac{\alpha_1 + \alpha_2}{6} t_j^3} - \sqrt{\frac{t_{j-1}^2}{2} + \frac{\alpha_1 + \alpha_2}{6} t_{j-1}^3}, \quad t_j = jh,$$

при этом разность между откликами (4) и (5^h) на сигналы (7) равна нулю при любом i и h . Очевидно, если

$$x^\beta(t) \in X_1 = \{\beta e(t), \quad \beta \in [0, B]\} \quad (8)$$

или

$$x^\beta(t) \in X_2 = \{\beta e(t), \quad \beta \in [-B, B]\}, \quad (9)$$

то невязка между $y_{et}^\beta(t_i)$ и моделями (5), (5^h) (в случае (8), (9) модели (5), (5^h) идентичны) является функцией β, α_1 и α_2 .

Постановке и решению задачи оптимизации выбора α_1 и α_2 в (7) для класса допустимых входных сигналов (8), (9) и посвящена данная работа.

2. Обозначим разность между откликами эталонной и квадратичной моделей на сигнал $x^\beta(t)$ через $N_{sq}(t, \beta, \alpha_1, \alpha_2)$.

Можно показать, что

$$N_{sq}(t, \beta, \alpha_1, \alpha_2) = \frac{t^3}{6}(\beta^3 - (\alpha_1 + \alpha_2)\beta^2 + \alpha_1\alpha_2\beta). \quad (10)$$

Поскольку в приложениях важную роль играет значение отклика динамической системы в конце переходного процесса ($t = T$), сформулируем следующие экстремальные задачи:

Найти

$$I) \quad (\alpha_1^*, \alpha_2^*) = \arg \min_{\alpha_1, \alpha_2 \in [0, B]} \left\{ \max_{\beta \in [0, B]} |N_{sq}(T, \beta, \alpha_1, \alpha_2)| \right\}; \quad (11)$$

$$II) \quad (\hat{\alpha}_1^*, \hat{\alpha}_2^*) = \arg \min_{\alpha_1, \alpha_2 \in [-B, B]} \left\{ \max_{\beta \in [-B, B]} |N_{sq}(T, \beta, \alpha_1, \alpha_2)| \right\}. \quad (12)$$

Решение задачи (11).

Не уменьшая общности, примем $T = 1$, $B = 1$, $\alpha_1 < \alpha_2$. В силу положительности α_1 , α_2 (10) как функция β имеет два внутренних экстремума в точках

$$\beta_{1,2} = \frac{\alpha_1 + \alpha_2 \mp \sqrt{(\alpha_1 + \alpha_2)^2 - 3\alpha_1\alpha_2}}{3} \quad (13)$$

(в β_1 — максимум, в β_2 — минимум). Равенства

$$N_{sq}(1, \beta_1, \alpha_1, \alpha_2) = -N_{sq}(1, \beta_2, \alpha_1, \alpha_2), \quad (14)$$

$$N_{sq}(1, \beta_1, \alpha_1, \alpha_2) = N_{sq}(1, 1, \alpha_1, \alpha_2) \quad (15)$$

образуют систему двух нелинейных алгебраических уравнений относительно α_1 , α_2 . В силу (10), (11) из (14) находим:

$$(\alpha_1 + \alpha_2) \left(-\frac{4}{27}(\alpha_1 + \alpha_2)^2 + \frac{2}{3}\alpha_1\alpha_2 \right) = 0. \quad (16)$$

Но $\alpha_1 + \alpha_2 \neq 0$, поэтому из (16) имеем

$$(\alpha_1 + \alpha_2)^2 = \frac{9}{2}\alpha_1\alpha_2, \quad (17)$$

откуда следует, что $\alpha_2 = 2\alpha_1$, так что $\beta_1 = (1 - \frac{\sqrt{3}}{3})\alpha_1$ и из (15) получаем кубическое уравнение, единственный корень которого, принадлежащий отрезку $[0, 1]$,

$$\alpha_1^* = 2\sqrt{3} - 3 \approx 0.4641. \quad (18)$$

Следовательно,

$$\alpha_2^* = 4\sqrt{3} - 6 \approx 0.9282. \quad (19)$$

Пара (18), (19) и является решением задачи I). При этом

$$N_{sq}^* \triangleq |N_{sq}(1, \beta_{1,2}^*, \alpha_1^*, \alpha_2^*)| \approx 0,0064.$$

График функции $|N_{sq}(1, \beta_1^*, \alpha_1^*, \alpha_2^*)|$ приведен на рис. 1.

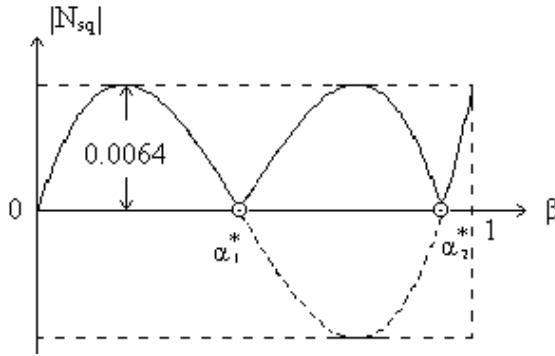


Рис. 1

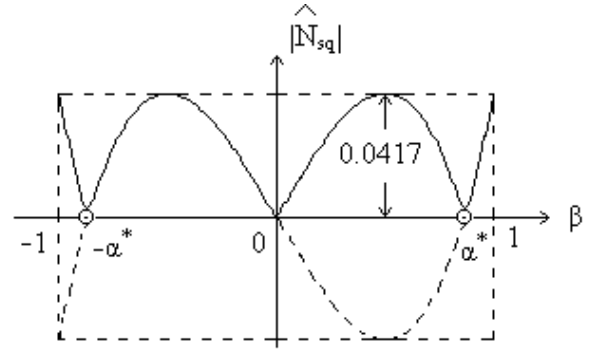


Рис. 2

Решение задачи (12).

В случае тестов (9) из симметричности (10) как функции α_1, α_2 с необходимостью следует равенство

$$\alpha_1 + \alpha_2 = 0.$$

Обозначим $\alpha_2 = -\alpha_1 = \alpha > 0$, так что в силу (10)

$$N_{sq}(1, \beta, -\alpha, \alpha) = \frac{1}{6}(\beta^3 - \alpha^2\beta). \quad (20)$$

Точка минимума (20) $\beta_1 = \frac{\alpha}{\sqrt{3}}$, поэтому условие

$$-N_{sq}(-1, \beta_1, -\alpha, \alpha) = N_{sq}(1, 1, -\alpha, \alpha)$$

дает кубическое уравнение

$$\alpha^3 + \frac{3\sqrt{3}}{2}\alpha^2 - \frac{3\sqrt{3}}{2} = 0$$

с единственным корнем в допустимой области

$$\alpha^* = \frac{\sqrt{3}}{2} \approx 0,8660, \quad (21)$$

который и определяет решение задачи II), причем

$$\widehat{N}_{sq}^* \triangleq |N_{sq}(1, \beta_1^*, -\alpha_1^*, \alpha_2^*)| \approx 0,0417.$$

График функции $|\widehat{N}_{sq}(1, \beta, -\alpha_1^*, \alpha_2^*)|$ приведен на рис. 2.

3. Используя дополнительно к (17) тестовый сигнал $x^{\alpha_3}(t) = \alpha_3 e(t)$, $\alpha_3 \neq \alpha_{1,2}$, можно построить на базе product integration method и кубичную модель, при этом невязка между y_{cub} и y_{et} , имеющей вид (4) с четырьмя слагаемыми, такова:

$$N_{cub}(t, \beta, \alpha_1, \alpha_2, \alpha_3) = \frac{t^4}{24} \left(\beta^4 - (\alpha_1 + \alpha_2 + \alpha_3)\beta^3 + (\alpha_1\alpha_2 + \alpha_1\alpha_3 + \alpha_2\alpha_3)t^2 - \alpha_1\alpha_2\alpha_3\beta \right). \quad (29)$$

Математическая система компьютерной алгебры MAPLE-8 дает следующие приближенные решения задач типа I), II):

$$(\alpha_1^*, \alpha_2^*, \alpha_3^*) = \arg \min_{\alpha_1, \alpha_2, \alpha_3 \in [0,1]} \left\{ \max_{\beta \in [0,1]} \left| N_{cub}(1, \beta, \alpha_1, \alpha_2, \alpha_3) \right| \right\} \approx (0, 2831; 0, 6772; 0, 9603);$$

$$|N_{cub}^*| \approx 0, 00038;$$

$$(\hat{\alpha}_1^*, \hat{\alpha}_2^*, \hat{\alpha}_3^*) = \arg \min_{\alpha_1, \alpha_2, \alpha_3 \in [-1,1]} \left\{ \max_{\beta \in [-1,1]} \left| N_{cub}(1, \beta, \alpha_1, \alpha_2, \alpha_3) \right| \right\} \approx (0, 9102; 0; 0, 9102);$$

$$|N_{cub}^*| \approx 0, 0071;$$

Графики соответствующих функций приведены на рис. 3, 4.

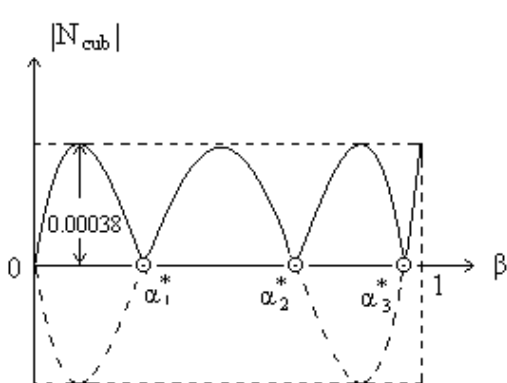


Рис. 3

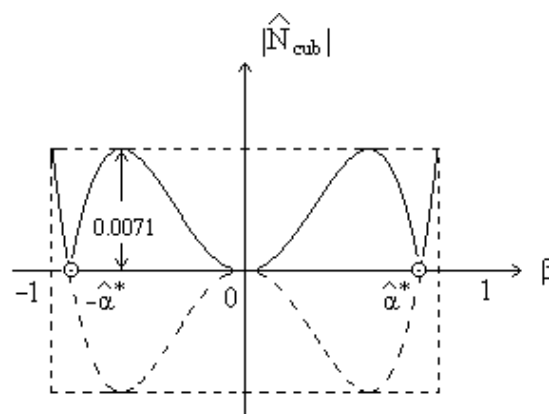


Рис. 4

Замечание 1. При B и T , отличных от 1, α_i^* следует умножить на B , а оптимальные невязки — на $(BT)^3$ для квадратичных и на $(BT)^4$ для кубических моделей.

Замечание 2. В [2] минимаксные задачи, аналогичные рассмотренным выше, были решены для математической модели, построенной с помощью метода идентификации ядер Вольтерра K_m , предложенного в [3] – [5]. Оказывается, применительно к эталонной модели (4) методика данной работы существенно выигрывает не только в простоте, но и в точности. Это объясняется обременительными ограничениями на параметры α_i в прежнем подходе, вытекающими из условий разрешимости специальных интегральных уравнений относительно K_m в классе непрерывных симметричных m -мерных функций.

Замечание 3. Дальнейшее развитие рассмотренного способа моделирования связано с применением product integration method непосредственно к многомерным сверткам, а не к их одномерным аппроксимациям.

Список литературы

- [1] Linz P. Product integration method for Volterra integral equations of the first kind // ВИТ, 1971. — Vol. 11. — P. 413–421.
- [2] А.С. Апарцин. Постановка и решение некоторых минимаксных задач // Труды XII Байкальской международной конференции, Иркутск, 2001. — Т. 1. — С. 68–73.

- [3] Apartsyn A.S. *Mathematical modeling the dynamic systems and objects with the help of the Volterra integral equations* // EPRI-SEI Joint seminar of methods for solving the problems on energy power systems development and control. Beijing, China, 1991. — P. 117–132.
- [4] Apartsyn A.S. *On some identification method for nonlinear dynamic systems* // ISEMA-92. — Shenzhen, China, 1992. — P. 288–292.
- [5] Апарцин А.С. *О новых классах линейных многомерных уравнений I рода типа Вольтерра*// Изв. вузов. Математика. — 1995. — № 11. — С. 28-42.

ON ONE CLASS OF MINIMAX PROBLEMS

A.S. Apartsyn

Melentiev Energy Systems Institute SB RAS, Irkutsk
e-mail: apartsyn@isem.sei.irk.ru

Abstract. Special minimax problems connected with constructing mathematical models of nonlinear dynamic systems of input-output type by Volterra polynomial are solved.

Key words: nonlinear dynamic system, identification, optimization, Volterra polynomial, product integration method.

РАЗРЕШИМОСТЬ И УСТОЙЧИВОСТЬ ЗАДАЧ МАТЕМАТИЧЕСКОГО ПРОГРАММИРОВАНИЯ ¹

Е.Г. Белоусов

Московский государственный университет, Москва
e-mail: beg@sumail.ru

Аннотация. Среди общих задач математического программирования выделяются классы разрешимых и устойчивых задач. В частности, показывается, что задача выпуклого полиномиального программирования является разрешимой и устойчивой по функционалу, но не устойчивой по аргументу; а задача кубического программирования с линейными ограничениями является разрешимой и устойчивой по аргументу, но не устойчивой по функционалу.

Ключевые слова: разрешимость, максимизирующая последовательность, устойчивость по аргументу, устойчивость по функционалу.

1. Постановка задачи

Рассматривается общая задача математического программирования (задача МП):

$$\sup \{f(x) \mid x \in A\} \quad (= \varphi), \quad (1)$$

где A – выпуклое замкнутое множество n -мерного евклидова пространства E_n , f – конечная в каждой точке E_n функция. Чаще всего множество A задается конечной системой неравенств

$$A = \{x \in E_n \mid f_i(x) \leq 0, i = 1, \dots, m\}, \quad (2)$$

где f_1, \dots, f_m – конечные в каждой точке E_n функции. Обозначим через $\psi = \{x \in A \mid f(x) = \varphi\}$ оптимальное множество задачи (1) и через $\rho(x, M)$ – расстояние от точки x до множества $M \subseteq E_n$. Не исключено, что для рассматриваемой задачи МП $\varphi = +\infty$, или что $\varphi < +\infty$, но $\psi = \emptyset$. Мы изучаем три свойства, характеризующих устойчивость рассматриваемой задачи; именно, мы выделяем классы задач МП, обладающих свойствами:

1) Условие $\varphi < +\infty$ влечет условие $\psi \neq \emptyset$, т.е., если из ограниченности задачи сверху следует ее разрешимость.

2) Условие $x^k \in A, f(x^k) \rightarrow \varphi$ влечет за собой, что $\rho(x^k, \psi) \rightarrow 0$, т.е., если всякая максимизирующая последовательность задачи сходится к оптимальному множеству задачи. Задачу МП в случае 2) мы называем устойчивой по аргументу.

3) Условие $x^k \in A, \rho(x^k, \psi) \rightarrow 0$ влечет за собой, что последовательность x^k является максимизирующей, т.е., выполняется условие $f(x^k) \rightarrow \varphi$. Задачу МП в случае 3) мы называем устойчивой по функционалу.

Хорошо известно, что для задачи линейного программирования, т.е. для задачи МП, в которой в (1) и (2) f, f_1, \dots, f_m – линейные функции, свойства 1), 2), 3) выполняются. Наша основная цель – выяснить в какой мере указанные свойства задачи линейного программирования распространяются на общую задачу МП.

¹Работа выполнена при поддержке РФФИ (проект 01-01-00056)

2. Классы разрешимых и устойчивых задач

О разрешимости задачи МП. В 1956 г. в работе [1] Франк и Вульф обобщили фундаментальную теорему о разрешимости задачи линейного программирования, доказав, что если в (1) и (2) f - произвольный квадратичный полином, f_1, \dots, f_m - линейные функции, $A \neq \emptyset$, $\varphi < +\infty$, то $\psi \neq \emptyset$. Альтернативное доказательство этого же результата было в дальнейшем дано в работах [2] и [3].

В 1977 г. Куммер в работе [4] обобщил результат Франка-Вульфа о разрешимости задачи квадратичного программирования с многогранными множествами на множества, представимые в виде суммы Минковского компакта и конечного конуса.

В 1999 г. Лю и Цанг в работе [5] обобщили теорему Франка и Вульфа доказав, что если в (1) и (2) f есть произвольный квадратичный полином и среди функций f_1, \dots, f_m имеется не более одной нелинейной, но выпуклой квадратичной, $A \neq \emptyset$, $\varphi < +\infty$, то $\psi \neq \emptyset$. Лю и Цанг построили пример, показывающий, что их результат не распространяется на случай, когда среди функций f_1, \dots, f_m есть уже две выпуклые квадратичные. Вот их пример: рассматривается задача (1), (2), в которой

$$f(x) = 2x_1x_2 - x_3x_4 - x_1^2, \quad f_1(x) = x_1^2 - x_3, \quad f_2(x) = x_2^2 - x_4.$$

Очевидно, что для любого $x \in A$ мы имеем

$$f(x) = 2x_1x_2 - x_3x_4 - x_1^2 \leq 2x_1x_2 - x_1^2x_2^2 - x_1^2 = -x_1^2 - (x_1x_2 - 1)^2 + 1 < 1,$$

но на последовательности $x^k = (1/k, k, 1/k^2, k^2)$, $k = 1, 2, \dots$ выполняется $x^k \in A$, $f(x^k) = 1 - 1/k^2 \rightarrow 1$ и, значит, $\varphi = 1$, но $\psi = \emptyset$.

Гипотеза. Если в (1) и (2) f есть произвольный квадратичный полином и среди функций f_1, \dots, f_m есть не более одной выпуклой полиномиальной (не обязательно выпуклой квадратичной), тогда условия $A \neq \emptyset$, $\varphi < +\infty$ влекут $\psi \neq \emptyset$.

В 1982 г. (см. [6], также [9]) результат Франка и Вульфа был обобщен на задачу кубического программирования:

Теорема 1. Если в (1) и (2) f - полином степени ≤ 3 , f_1, \dots, f_m - линейные функции, то из условий $A \neq \emptyset$, $\varphi < +\infty$ следует разрешимость рассматриваемой задачи МП, т.е. $\psi \neq \emptyset$, и ее устойчивость по аргументу.

Отметим, что до недавнего времени Теорема 1 была неизвестна на Западе-см.[5], [7]. Так, например, в [5] она сформулирована как "открытый вопрос". Заметим также, что Теорема 1 не обобщается на задачи максимизации полиномов степени ≤ 4 - достаточно рассмотреть задачу $\sup \{-(x_1^2 + (1 - x_1x_2)^2) \mid x \in E_2\}$.

В работе [6] построен пример, показывающий, что Теорема 1 не обобщается с многогранных множеств на множества A , представимые в виде суммы Минковского компакта и конечного конуса, а также на множества, задаваемые даже одним выпуклым квадратичным ограничением. Действительно, пусть в (1) и (2)

$$f(x) = -x_1^2x_2 + 2x_1x_3, \quad A = \{x \in E_3 \mid (x_2 - 1)^2 + x_3^2 \leq 1\}$$

Для любого $x \in A$ мы имеем $x_2 \geq 0$, $x_3^2 \leq -x_2^2 + 2x_2$. Поэтому, если

$$x \in A, \quad x_2 = 0, \quad \text{то } x_3 = 0 \text{ и, следовательно, } f(x) = f(x_1, 0, 0) = 0. \quad (3)$$

Для любого $x \in A$, $x_2 > 0$ мы будем иметь

$$f(x) = -x_2 \left(x_1 - \frac{x_3}{x_2} \right)^2 + \frac{x_3^2}{x_2} \leq \frac{x_3^2}{x_2} \leq \frac{-x_2^2 + 2x_2}{x_2} = 2 - x_2 < 2. \quad (4)$$

На допустимой последовательности $x^t = \left(\sqrt{-1 + 2/t}, t, \sqrt{2t - t^2} \right)$ мы получим $f(x^t) = 2 - t \rightarrow 2$ при $t \rightarrow 0, t > 0$. Отсюда, а также из (3) и (4), следует, что $\varphi = 2$, но $\psi = \emptyset$, т.е. рассматриваемая задача не разрешима.

Приведенный пример также показывает, что вышеупомянутый результат Куммера не обобщается с квадратичных на кубические целевые полиномы.

Хорошо известно, что задача выпуклого программирования уже в E_1 не является, вообще говоря, разрешимой в случае ее ограниченности сверху - достаточно рассмотреть задачу $\sup\{-\exp x \mid x \in E_1\}$. Однако по другому обстоит дело с задачей выпуклого полиномиального программирования (задачей ВПП). Именно, в 1977 г. в [8] была установлена

Теорема 2. Если в (1) и (2) $(-f), f_1, \dots, f_m$ - выпуклые полиномы, то из условий $A \neq \emptyset, \varphi < +\infty$ следует $\psi \neq \emptyset$, т.е. разрешимость рассматриваемой задачи МП.

Следуя схеме доказательства Теоремы 2, ее можно обобщить на случай задачи МП вида (1) и (2), где $(-f), f_1, \dots, f_m$ - квазивыпуклые полиномы, т.е. полиномы, для которых все множества Лебега $A(\lambda) = \{x \in E_n \mid f_i(x) \leq \lambda\}$ - выпуклые (либо пустые).

Теорема 2 не обобщается с выпуклых замкнутых полиномиальных множеств на выпуклые замкнутые конические множества A даже если f - это квадратичный вогнутый полином. Действительно, рассмотрим задачу (1), (2), в которой

$$f(x) = -(x_1 - 1)^2 - x_2^2, \quad A = \{x \in E_3 \mid x_1^2 \leq x_2x_3, x_2 \geq 0, x_3 \geq 0\}$$

- выпуклый замкнутый неконечный конус. Нетрудно видеть, что мы будем иметь $\varphi = 0$, но $f(x) < 0$ для всех $x \in A$ и, значит, $\psi = \emptyset$.

Другие классы разрешимых задач МП систематизированы в [8].

Об устойчивости по аргументу задачи МП. Как отмечалось в Теореме 1 задача кубического программирования с линейными ограничениями является устойчивой по аргументу. Покажем, что задача максимизации полинома 4-ой степени на E_3 уже не является устойчивой по аргументу в случае ее разрешимости. Действительно, рассмотрим задачу

$$\sup \{-(x_1^2 + (x_3 - x_1x_2)^2) \mid x \in E_3\} \quad (= \varphi)$$

Мы, очевидно, будем иметь $\varphi = 0, \psi = \{x_1 = x_3 = 0\}$ - прямая, и на последовательности $x^k = (1/k, \alpha k, \alpha)$, где $\alpha \neq 0$, выполняется $f(x^k) = -1/k^2 \rightarrow 0$, но $\rho(x^k, \psi) > |\alpha| = const > 0$.

Покажем, что и разрешимая задача выпуклого программирования в E_2 не является, вообще говоря, устойчивой по аргументу. Рассмотрим задачу

$$\sup \{-\sqrt{x_1^2 + x_2^2} + x_1 \mid x \in E_2\} \quad (= \varphi)$$

Мы, очевидно, будем иметь $\varphi = 0, \psi = \{x_1 \geq 0, x_2 = 0, \}$ - луч, и на последовательности $x^k = (k, \alpha)$, где $\alpha = const \neq 0$, будем иметь $f(x^k) \rightarrow 0$, но $\rho(x^k, \psi) = |\alpha| > 0$.

Покажем, наконец, что и разрешимая задача ВПП в E_4 не является устойчивой по аргументу. Рассмотрим задачу

$$\sup \{f_0(x) \mid x \in A\}, \quad \text{где} \quad f_0(x) = x_3,$$

$$A = \{x \in E_n \mid f(x_1, x_2, x_3) - x_4 \leq 0, x_3 \leq \lambda\}, \quad \lambda = \text{const}, \text{ где}$$

$$f(x_1, x_2, x_3) = x_1^3 x_2^3 x_3 + x_1^2 x_2^4 x_3^2 + 3000 x_1^4 x_2^2 + x_2^6 x_3^4 + 2 \cdot 10^7 x_1^6 + 63 x_2^8 + x_2^6 + x_3^{16}$$

– выпуклый полином, свойства которого изучены в [9]. В частности, механизм выпуклости этого полинома (см.[9] стр.22) дается следующей схемой

$$\begin{array}{ccc|ccc} 3 & 3 & 1 & & 4 & 2 & 0 \\ 4 & 2 & 0 & 1/2 & 6 & 0 & 0 \\ 2 & 4 & 2 & 1/2 & 0 & 6 & 0 \\ 4 & 2 & 0 & 1/2 & & & \\ 0 & 6 & 4 & 1/2 & & & \\ 0 & 8 & 0 & 3/4 & & & \\ 0 & 0 & 16 & 1/4 & & & \end{array} \left| \begin{array}{l} \\ 2/3 \\ 1/3 \\ \\ \\ \\ \end{array} \right.$$

Мы, очевидно, будем иметь $\varphi = \lambda$, $\psi = (\text{epi } f) \cap \{x_3 = \lambda\}$. Рассмотрим последовательность

$$x^t = \left(t, t^q, \lambda - \frac{1}{t^\theta}, f(t, t^q, \lambda - \frac{1}{t^\theta}) \right).$$

Легко видеть, что $x^t \in A$, $f_0(x^t) = \lambda - 1/t^\theta \rightarrow \lambda = \varphi$ при $t \rightarrow +\infty$, $\theta > 0$, т.е. последовательность x^t является максимизирующей. Нетрудно видеть, что при $q \in (2/3, 3/4)$, $0 < \theta < \min\{3q - 2, 3 - 4q\}$ и ограниченных по модулю α и β имеет место предельное соотношение

$$f(t + \alpha, t^q + \beta, \lambda) - f\left(t, t^q, \lambda - \frac{1}{t^\theta}\right) \rightarrow +\infty \quad \text{при } t \rightarrow +\infty$$

и следовательно $\rho(x^t, \psi) \rightarrow +\infty$ при $t \rightarrow +\infty$. Таким образом, рассматриваемая задача разрешима, но неустойчива по аргументу.

Гипотеза. Если в (1), (2) все выпуклые полиномы $(-f), f_1, \dots, f_m$ имеют степень < 16 , то соответствующая задача ВПП является устойчивой по аргументу.

Об устойчивости по функционалу задачи МП. Нетрудно видеть, что задача квадратичного программирования не является, вообще говоря, устойчивой по функционалу. Действительно, рассмотрим задачу

$$\sup \{-x_1 x_2 \mid x_1 \geq 0, x_2 \geq 0\}.$$

Мы, очевидно, будем иметь $\varphi = 0$, $\psi = \{x_1 = 0, x_2 \geq 0\} \cup \{x_1 \geq 0, x_2 = 0\}$ и на последовательности $x^k = (k, 1/k)$, $k = 1, 2, \dots$ будет выполняться $\rho(x^k, \psi) \rightarrow 0$, но $f(x^k) = -1 \not\rightarrow \varphi = 0$ и, значит, последовательность x^k не является максимизирующей.

Покажем, что и задача выпуклого программирования в E_2 не является, вообще говоря, устойчивой по функционалу. Действительно, рассмотрим задачу

$$\sup \{\min\{-x_1^2 + x_2, 0\} \mid x \in E_2\}.$$

Мы, очевидно, будем иметь $\varphi = 0$, $\psi = \{x_2 \geq x_1^2\}$ и на последовательности

$$x^k = (k + 1/\sqrt{k}, k^2) \text{ выполняется } \rho(x^k, \psi) \rightarrow 0, \text{ но } f(x^k) = -2\sqrt{k} - \frac{1}{k} \rightarrow -\infty$$

и, значит, последовательность x^k не является максимизирующей.

Однако широкие классы задач выпуклого программирования и в том числе задача ВПП в E_n являются все же, как это явствует из нижеследующей теоремы, устойчивыми по функционалу.

Теорема 3. Пусть A – выпуклое множество и f – вогнутый на E_n полином. Тогда, если задача $\sup \{f(x) \mid x \in A\}$ разрешима, то она и будет устойчивой по функционалу.

Список литературы

- [1] M.Frank, P.Wolf *An algorithm for quadratic programming*. Naval research Logistics quarterly, vol.3, pp.95-110, 1956.
- [2] B.C.Eaves *On quadratic Programming*. Management Science, vol.17, pp.698-711, 1971.
- [3] E.Blum, W.Oettli *Direct proof of the existence theorem in quadratic programming*. Operations research, vol.20, pp.165-167, 1972.
- [4] B.Kummer *Globale Stabilitat quadratischer Optimierungsprobleme*. Wissenschaftliche zeitschrift der Humboldt-Universitat zu Berlin, Math.-nat.R., vol.5, pp.565-569, 1997.
- [5] Z-Q.Luo, S.Zhang *On extensions of the Frank-Wolfe theorems*. Computational optimization and applications, vol.13, pp.87-110, 1999.
- [6] В.Г.Андронов, Е.Г.Белоусов, В.М.Широнин *О разрешимости задачи полиномиального программирования* Известия АН СССР.Техническая кибернетика. №4, стр.194-197, 1982.
- [7] E.G.Belousov, D.Klatte *A Frank-Wolfe type theorem for convex polynomial programs*. Computational optimization and applications, vol.22, pp.37-48, 2002.
- [8] Е.Г.Белоусов *Введение в выпуклый анализ и целочисленное программирование* М.: Изд-во МГУ, 1977, 196 с.
- [9] Е.Г.Белоусов, В.Г.Андронов *Разрешимость и устойчивость задач полиномиального программирования* М.: Изд-во МГУ, 1993, 272 с.

SOLVABILITY AND STABILITY FOR PROBLEM OF MATHEMATICAL PROGRAMMING

E.G. Belousov

Moscow State University, Moscow

e-mail: beg@sumail.ru

Abstract. Among general problems of mathematical programming classes of solvable and stable problems are selected. We show that convex polynomial program is solvable and stable according to functional, but not stable according to argument, and cubic program with linear constraints is solvable and stable according to argument, but not stable according to functional.

Key words: solvability, maximizing sequence, stability according to argument, stability according to functional.

ПРИМЕРЫ РАСПАРАЛЛЕЛИВАНИЯ ПРОЦЕССА РЕШЕНИЯ В НЕКОТОРЫХ ЗАДАЧАХ ОПТИМИЗАЦИИ

Г.И. Забиняко, Е.А. Котельников

*Институт вычислительной математики и математической геофизики СО РАН,
Новосибирск*

Аннотация. В статье рассматриваются параллельные алгоритмы решения задач целочисленного линейного, целочисленного квадратичного программирования и невыпуклого квадратичного программирования. Приводятся примеры решения задач в последовательном и параллельном режимах.

Ключевые слова: целочисленное линейное программирование, целочисленное квадратичное программирование, невыпуклое квадратичное программирование, система параллельного программирования MPI.

Введение

В статье рассматривается распараллеливание метода ветвей и границ с односторонним ветвлением на примере задач целочисленного линейного программирования (ЦЛП), целочисленного квадратичного программирования (ЦКП) и невыпуклого квадратичного программирования (НКП).

На k -м уровне в методе ветвей и границ текущее подмножество допустимых решений G_k по некоторым правилам ветвления подразделяется на подмножества G_{k+1}^- и G_{k+1}^+ , что порождает систему подмножеств, упорядоченных по включению. Эта связь представляется в виде дерева с корнем, отвечающим исходному допустимому множеству G , и любой непомеченной ветви можно поставить в соответствие алгоритм решения задачи на подмножестве G_k . Загружая процессоры в определенном порядке решением задач на подмножествах, получим асинхронное исполнение алгоритма ветвей и границ с односторонним ветвлением.

В результате получается некоторое сочетание алгоритмов с односторонним и одновременным ветвлением [1]. Использование системы параллельного программирования MPI существенно упрощает реализацию такого сочетания алгоритмов. Во время исполнения процессы обмениваются короткими сообщениями на фоне выполнения большого числа арифметических операций, что обеспечивает высокую эффективность распараллеливания как на вычислительных системах с общей памятью, так и на кластерах.

На кластере МВС-1000/М и на мультипроцессорной системе RM-600 E30 было решено большое количество задач ЦЛП и ЦКП, в параллельном и последовательном режимах, статистические данные по решению некоторых из этих задач приводятся далее. Распараллеливание алгоритмов ветвей и границ позволяет во многих задачах получить ускорение выше линейного. Программное обеспечение для задач ЦЛП и ЦКП выполнено на Фортране с использованием системы MPI, а для задач НКП разрабатывается в настоящее время.

1. Используемые алгоритмы

Задачи ЦЛП представляются в виде: минимизировать $f(x) = (c, x)$ при условиях

$$Ax = b, \quad (1)$$

$$\alpha \leq x \leq \beta, \quad (2)$$

$$x_j - \text{целые для } j \in J. \quad (3)$$

Здесь A – матрица размера $m \times n$, векторы $c, x, \alpha, \beta \in R^n$, $b \in R^m$. В задачах ЦКП целевая функция $f(x) = (Qx, x) + (c, x)$, Q – симметричная матрица размера $n \times n$ и $Q \geq 0$.

Решение исходных задач ЦЛП и ЦКП сводится к решению последовательности оценочных задач, в которых отсутствует условие (3), а вместо (2) используются условия $\alpha^i \leq x \leq \beta^i$, где векторы α^i и β^i формируются по правилам метода ветвей и границ. Оценочные задачи ЦЛП решаются симплекс-методом, а ЦКП – методом приведенного градиента [2] с применением для минимизации в подпространстве супербазисных переменных линейного метода сопряженных градиентов.

Для задач ЦЛП часто используются штрафные оценки [1]. Значение базисной переменной x_j для $j \in J$ в оптимальном базисе очередной задачи представим в виде $x_j = [x_j] + v_j$, $[x_j]$ – целевая часть x_j . Штраф за увеличение базисной переменной x_j на величину $1 - v_j$ обозначим через P_j^+ , а за уменьшение на v_j – через P_j^- .

Штрафы используются как для выбора переменной ветвления, так и для пометки вершин или ветвей дерева решений. После вычисления штрафов определяется $P_{\min} = \min_j \{\min\{P_j^-, P_j^+\}\}$ и $P_{\max} = \max_j \{\max\{P_j^-, P_j^+\}\}$. Пусть f^i – оптимальное значение целевой функции в i -й оценочной задаче, тогда, при $f^i + P_{\min} \geq r^i$, где r^i – текущий рекорд, пометается соответствующая вершина. Эта вершина не используется далее в качестве родительской. Аналогично, пометается некоторая ветвь, если $f^i + P^i \geq r^i$, где P^i – штраф P_j^+ или P_j^- . Для ветвления выбирается переменная x_j с индексом j , на котором достигается P_{\max} ($P_{\max} \neq 0$). Список оценочных задач пополняется информацией о двух новых оценочных задачах с модифицированными ограничениями на x_j : $x_j \leq [x_j^i]$ и $x_j \geq [x_j^i] + 1$. В качестве очередной оценочной задачи выбирается та, которой соответствует меньший штраф P_j^- или P_j^+ . Если $P_{\max} = 0$, то для выбора переменной ветвления рассматриваются величины $(1 - v_j)$ и v_j .

В задачах ЦКП также делается попытка воспользоваться штрафными оценками. Штрафы для квадратичной функции f заменим штрафами для линейной функции $g(x) = (\nabla f(x), x - x^*)$, где x^* – решение очередной оценочной задачи квадратичного программирования. В нелинейном случае при вычислении штрафов необходимо дополнительно учесть супербазисные переменные, если они присутствуют в оптимальном решении оценочной задачи.

Задачи НКП формулируются следующим образом: найти $\text{globalmax} f(x) = (Qx, x) + (c, x)$ на ограниченном многограннике (1), (2). На множестве G , заданном условиями (1), (2), строится линейная мажоранта M функции f , если Q – положительно – определенная матрица; и вогнутая квадратичная функция, если Q – знакопеременная матрица. Мажоранта $M(x)$ обладает следующими свойствами. Пусть $x^* \in G$ – точка, у которой хотя бы одна компонента x_i^* удовлетворяет условию $\alpha_i < x_i^* < \beta_i$. Рассмотрим два подмножества множества G :

$$G_i^- = \{x \in G : \alpha_i \leq x_i \leq x_i^*\}, \quad G_i^+ = \{x \in G : x_i^* \leq x_i \leq \beta_i\}.$$

На множествах G_i^-, G_i^+ определим мажоранты M^-, M^+ функции f . Тогда справедливы утверждения: $\forall x \in G_i^- M(x) \geq M^-(x)$, $\forall x \in G_i^+ M(x) \geq M^+(x)$ и $M^-(x^*) = M^+(x^*)$.

Воспользовавшись этими свойствами мажорант, можно построить алгоритм метода ветвей и границ с односторонним ветвлением для поиска глобального максимума функции f на множестве G . На k -м уровне метода множество допустимых решений G_k разбивается на два подмножества G_k^- и G_k^+ таким образом, чтобы мажоранты M_k, M_k^-, M_k^+ функции f на G_k, G_k^-, G_k^+ соответственно, удовлетворяли условиям: $M_k \geq M_k^-$ на G_k^- , $M_k \geq M_k^+$ на G_k^+ , $M_k^- = M_k^+$ на $G_k^- \cap G_k^+$. В качестве целевой функции оценочной задачи выбирается M_k^- или M_k^+ , поэтому на каждом уровне данного метода решается оценочная задача линейного программирования, если Q – положительно-определенная матрица, или квадратичного программирования, если Q – знакопеременная матрица.

Для выбора переменной ветвления здесь также используются штрафные оценки. Подробное описание алгоритма приведено в [3].

2. Организация параллельных вычислений

Для распараллеливания среди процессорных элементов выделяется элемент с нулевым номером, который, кроме исполнения алгоритма ветвей и границ, выполняет некоторые диспетчерские функции. По ходу решения задач на нулевом процессоре накапливается справочная информация для загрузки процессорных элементов.

На начальном этапе данные о задаче считываются с дисковой памяти нулевым процессором и пересылаются всем остальным. Далее на нулевом процессоре решается нулевая оценочная задача, а остальные процессоры находятся в ожидании.

На каждом из процессоров предусматривается подготовка данных для обеспечения возможности загрузки какого-либо из процессоров. В ходе выполнения алгоритма для некоторой непомеченной ветви запоминается информация в списке оценочных задач и подготавливаются данные для передачи какому-либо другому процессору. Для задач ЦЛП необходимо запомнить массивы, отвечающие спискам базисных и небазисных переменных, а для задач ЦКП и НКП, кроме этого, – массивы с перечнем индексов супербазисных переменных и значений супербазисных переменных.

Для загрузки любого процессора в его оперативную память, кроме выше перечисленных данных, пересылается некоторая часть списка оценочных задач, отвечающая уровню k , с которого начинается решение оценочных задач на данном процессоре. В переданном списке задач все ветви помечены, кроме той с которой начинается выполнение алгоритма. Для упорядочения загрузки процессорных элементов на нулевом процессоре хранится массив пар (i, k_i) , где i – номер процессора, а k_i – уровень подготовленной на i -м процессоре оценочной задачи. Если некоторый i -й процессор простаивает, то это помечается присвоением $k_i = -1$.

Рассмотрим случай, когда i -й процессор завершил выполнение алгоритма (достигнут нулевой уровень) или процессор изначально еще не загружен. Если номер процессорного элемента $i > 0$, то нулевому процессору i -й процессор посылает запрос на загрузку. После получения запроса на нулевом процессоре выбирается пара (j, k_j) с минимальным значением $k_j > 0$ и на j -й процессор отправляется сообщение о необходимости загрузки i -го

процессора. В результате на j -м процессоре ветвь уровня k_j помечается и отправляются данные для загрузки на i -й процессор. Если не существует $k_j > 0$, то i -й процесс находится в ожидании. При завершении исполнения алгоритма на нулевом процессоре аналогично определяется номер j -го процессора для загрузки.

Если на некотором процессоре получено новое значение рекорда, то это значение пересылается всем остальным процессорам. Полученное значение рекорда используется для обновления информации о помеченных и непомеченных ветвях в списке оценочных задач. Может оказаться, что на некотором из процессоров j помечается ветвь которая подготовлена для загрузки других процессоров. В этом случае на нулевой процессор посылается информация об изменении k_j . Возможно, также, что пометке подлежит ветвь, с которой начинался процесс решения на некотором из процессоров, тогда на этом процессоре завершается исполнение алгоритма и формируется запрос на загрузку.

Задача решена, если на всех процессорных элементах достигнут нулевой уровень.

3. Примеры решения задач

Расчеты производились на кластере МВС-1000/М с процессорами DECAlpha-212644, с тактовой частотой 833 МГц. Тестовые задачи взяты из [4]. В табл.1 отражены результаты решения задач ЦЛП в последовательном и параллельном (на десяти процессорах) режимах.

Таблица 1

N	m	m_e	n	n_i	it	S_{it}	t	t_p	t/t_p
1	91	–	104	58	7547308	2504731	634	31	20.45
2	91	–	104	58	7547308	2732125	634	44	14.41
3	97	96	1989	1989	3540594	2136474	1315	104	12.64
4	363	363	1298	1254	9850308	10108694	3983	584	6.82
5	1248	120	1224	720	> 120000000	126305890	133448	13560	9.82
6	1392	48	1224	408	19804205	17061570	20551	1887	10.89
7	2176	123	6000	6000	44597720	25133600	50398	3715	13.57

Примечание. m – число ограничений в матрице A , m_e из них на равенство, n – число переменных, среди которых n_i – целочисленные, it – общее число итераций в оценочных задачах в последовательном режиме, S_{it} – суммарное число итераций в оценочных задачах на всех процессорах в параллельном режиме, t – затраты времени на решение в последовательном режиме, t_p – затраты времени в параллельном режиме.

В строках 1 и 2 табл.1 приводится статистика по решению одной и той же задачи для разных запусков программы. Большие вариации в статистике объясняются тем, что здесь среднее число итераций на решение оценочной задачи меньше трех и процесс решения сильно зависит от загруженности коммуникационной сети. Решение задачи 5 в последовательном режиме прервано после выполнения заданного числа итераций, при этом решение получено, но не обоснована оптимальность.

В табл.2 приведены входные параметры задач ЦКП и результаты их решения в последовательном режиме. Столбец помеченный sr показывает общее число оценочных задач, а p_0 – случаев $P_{\max} = 0$.

Таблица 2

№ задачи	m	m_e	n	n_i	it	cp	p_0	t
1	106	—	122	60	354065	35765	18	67
2	576	128	505	28	32317905	71883	30294	45315
3	75	75	1899	1898	42003240	91423	10577	51143
4	68	—	151	150	1054772	53052	127	381
5	212	35	260	259	3848924	54826	2086	3057
6	4480	4367	10957	97	5500000	5411	—	83258
7	1192	132	840	48	32000000	150246	4166	85163
8	195	26	338	169	18496522	1324478	6604	4480
9	296	40	512	256	16239520	985215	15115	5751

Решение задач 6 и 7 прервано после выполнения заданного числа итераций.

В табл.3 результаты решения задач ЦКП в параллельном режиме на восьми процессорах.

Таблица 3

	Номер задачи								
	1	2	3	4	5	6	7	8	9
it_1	48243	3650064	7246301	123501	339065	2319167	15065957	2018111	1466794
it_2	47246	3770439	4379823	116647	324842	2500000	14285510	2005200	1490414
it_3	48071	3504662	4669789	121271	333464	2348354	14172405	2040903	1486702
it_4	48566	3947817	4523608	115781	315766	2361807	13030163	2039631	1423969
it_5	47621	3745081	3431145	113520	348888	2320268	14535137	1993103	1460954
it_6	46654	3707780	3473066	115535	309939	2326606	14784738	2019802	1449547
it_7	47678	3413254	3294001	115900	307952	2412781	13322902	2010772	1467747
it_8	45102	3532041	3281568	119760	302719	2317997	14299376	1963402	1529396
S_{it}	379181	29271138	34299301	941915	2582635	18906980	113496188	16090924	11775523
t_p	9.18	5187.	8851	45.49	293.4	40439.	38292	502.7	528.6
t/t_p	7.27	8.74	5.78	8.37	10.42	—	—	8.91	10.88

it_1, \dots, it_8 – обозначают общее число итераций решения оценочных задач на процессорных элементах.

Подробная информация о программах ЦКП и их тестирований приведена в [5].

Список литературы

- [1] Ковалев М.М. *Дискретная оптимизация (целочисленное программирование)*. Минск: Изд-во Белорус. ун-та, 1977.
- [2] Муртаф Б. *Современное линейное программирование. Теория и практика*. - М.: Мир, 1984.
- [3] Котельников Е.А. *Поиск глобального максимума квадратичной функции при линейных ограничениях*. // Сиб. журн. вычисл. математики / РАН. Сиб. отд-ние.- Новосибирск, 2004.-Т. 7, № 4.-С. 327-334.
- [4] <http://plato.la.asu.edu>
- [5] Забиняко Г.И., Котельников Е.А. *Параллельный алгоритм целочисленного квадратичного программирования*. // Вычислительные технологии. - 2004. - Т. 9, № 1. -С. 34-41.

EXAMPLES OF A PARALLELIZING THE PROCESS OF A SOLUTION IN SOME PROBLEMS OF OPTIMIZATION

G.I. Zabinyako, E.A. Kotel'nikov

Siberian Branch of the Russian Academy of Sciences

Institute Computational Mathematics and Mathematical Geophysics, Novosibirsk

e-mail: zabin@rav.sccc.ru

Abstract. In the article, the parallel algorithms the problem solving of integer linear, integer quadratic programming and nonconvex quadratic programming are considered. The examples of problem solving in sequential and parallel alternatives are given.

Key words: Integer linear programming, integer quadratic programming, nonconvex quadratic programming, system of parallel programming MPI.

РЕШЕНИЕ ОБРАТНОЙ НЕЛИНЕЙНОЙ ЗАДАЧИ ДОПОЛНИТЕЛЬНОСТИ

А.В. Зыкина

*Омский государственный технический университет, Омск
e-mail: zykina@omgtu.ru*

Аннотация. В работе исследуется решение обратной задачи для нелинейной задачи дополнителности. Предлагается алгоритм решения обратной задачи, приводится его теоретическое обоснование.

Ключевые слова: параметрическая нелинейная задача дополнителности, обратная задача.

Введение

Для многих математических задач можно построить обратные задачи. С точки зрения математического моделирования это чрезвычайно важный класс задач, поскольку он позволяет учитывать в модели заданные свойства искомых решений. Кроме того, важность этого класса задач обусловлена также и тем обстоятельством, что сложные содержательные задачи могут быть записаны как обратные задачи для известных, хорошо изученных классов задач.

К примеру, модель экономического равновесия Эрроу-Дебре является обратной (векторной) задачей оптимизации [1].

Обычно принимают следующее определение обратных задач: две задачи называются обратными друг к другу, если в постановку каждой из них входит решение другой. Из определения следует, что имеется некоторый произвол в том, какую из двух задач называть прямой, а какую – обратной. Обычно более простая или лучше изученная задача называется прямой [2].

Исследование обратных задач можно встретить у многих авторов (И.И. Еремин, Л.А. Истомина, В.П. Булатов и др.). В работе [3] обратная задача формулируется как система различных по природе задач. Так обратная задача выпуклого программирования определяется следующим образом:

для заданных скалярной функции $f(y, q)$ и векторных функций $G_0(y, q)$, $G_1(y, q)$ и $G_2(y, q)$ требуется определить пару векторов y^*, q^* , которая удовлетворяет системе, содержащей экстремальное включение, уравнения и (или) неравенства :

$$y^* \in \text{Argmin}\{f(y, q^*) | G_0(y, q^*) \leq 0, y \in Q_0\}, q^* \in Q_1,$$

$$G_1(y^*, q^*) = 0, G_2(y^*, q^*) \leq 0.$$

Другими словами, в параметрическом относительно q семействе задач выпуклого программирования требуется выбрать параметр $q = q^*$ и отвечающий ему оптимум $y = y^*$, такие, чтобы выполнялась система уравнений и (или) неравенств $G_1(y^*, q^*) = 0$, $G_2(y^*, q^*) \leq 0$.

В данной работе для нелинейной задачи дополнителности с параметрами приводится постановка обратной задачи, которая состоит в нахождении таких параметров для нелинейной задачи дополнителности, при которых заданные обратной задачей значения

переменных исходной нелинейной задачи дополнителъности являются ее решением.

Параметрическая нелинейная задача дополнителъности

Рассмотрим нелинейную задачу дополнителъности $NCP(P, q)$ вида:

$$\omega = P(y) - q, \quad \omega \geq 0, \quad y \geq 0, \quad y^T \omega = 0. \quad (1)$$

Здесь $w, y, q \in R^m$, $P: R^m \rightarrow R^m$.

В частном случае, когда отображение P линейно, то есть $P(y) = Py$, где P – квадратная матрица соответствующего размера, получаем линейную задачу дополнителъности $LCP(P, q)$.

В дальнейшем будем предполагать, что отображение P – сильно монотонно и удовлетворяет условию Липшица, а именно, для любых двух точек y^1 и y^2 выполняются неравенства

$$(y^2 - y^1)^T (P(y^2) - P(y^1)) \geq \beta \|y^2 - y^1\|^2, \quad \beta > 0, \\ \|(P(y^2) - P(y^1))\| \leq \nu \|y^2 - y^1\|, \quad \nu > 0.$$

Известно [4], что любое сильно монотонное отображение является строго монотонным, а если к тому же оно удовлетворяет условию Липшица, то такое отображение является обратно сильно монотонным:

$$(y^2 - y^1)^T (P(y^2) - P(y^1)) \geq \mu \|P(y^2) - P(y^1)\|^2, \quad \mu > 0.$$

Для линейного отображения $P(y) = Py$ условие сильной монотонности соответствует положительной определенности матрицы P .

При выполнении указанных предположений существует единственное решение y^* задачи дополнителъности (1) для любой правой части $q \in R^m$ [5].

Для последующих исследований нам потребуется вспомогательное утверждение. Пусть в $NCP(P, q)$ правая часть q является параметром, тогда будем говорить о параметрической нелинейной задаче дополнителъности $PNCP(P, q)$.

Теорема 1.

Решение $y = y(q)$ задачи $PNCP(P, q)$ с сильно монотонным отображением P непрерывно зависит от правой части q . Более того, для любых двух точек q^1 и q^2 соответствующие решения $y^1 = y(q^1)$ и $y^2 = y(q^2)$ задачи (1) связаны условиями Липшица

$$\|y^2 - y^1\| \leq \frac{1}{\beta} \|q^2 - q^1\|, \quad (2)$$

где $\beta > 0$ – константа сильной монотонности отображения P .

Для доказательства теоремы достаточно показать выполнение неравенства (2).

Рассмотрим параметрическую нелинейную задачу дополнителъности $PNCP(P, q)$ (1), в которой

$$q = Hx + g,$$

где H – матрица размерности $m \times n$, x – вектор размерности n , g – вектор размерности m , и вектор x выступает в качестве параметра. Тогда параметрическая нелинейная задача дополненности $PNCSP(P, q)$ запишется в виде $PNCSP(P, Hx + g)$

$$\omega = P(y) - Hx - g, \quad \omega \geq 0, \quad y \geq 0, \quad y^T \omega = 0. \quad (3)$$

Следствие.

Для параметрической (относительно параметра x) нелинейной задачи дополненности (3) оценка (2) принимает вид

$$\|y^2 - y^1\| \leq \frac{\|H\|}{\beta} \|x^2 - x^1\|,$$

где $y^1 = y(x^1)$, $y^2 = y(x^2)$ – соответствующие решения задачи (3), а под $\|H\|$ понимается спектральная норма матрицы H ($\|H\| = \sqrt{\Lambda}$, где Λ – максимальное собственное число матрицы HH^T).

Постановка обратной задачи дополненности

Определим обратную нелинейную задачу дополненности следующим образом: для заданных векторных функций $P(y)$, $G_1(y, q)$ и $G_2(y, q)$ требуется определить пару векторов y^*, q^* , которая удовлетворяет системе:

$$\begin{aligned} \omega = P(y) - q, \quad \omega \geq 0, \quad y \geq 0, \quad y^T \omega = 0, \quad q \in R^m, \\ G_1(y, q) = 0, \quad G_2(y, q) \leq 0. \end{aligned}$$

Другими словами, в параметрическом относительно q семействе нелинейных задач дополненности требуется выбрать параметр $q = q^*$ и отвечающее ему решение задачи дополненности $y = y^*$, такие, чтобы выполнялась система равенств и неравенств $G_1(y, q) = 0$, и $G_2(y, q) \leq 0$.

Для задачи $PNCSP(P, Hx + g)$ (3) обратную задачу дополненности определим так: найти такие параметры $x \in R^n$ для нелинейной задачи дополненности (3), при которых решение $y = y(x)$ задачи (3) удовлетворяет следующим условиям

$$H^T y(x) = 0. \quad (4)$$

Рассмотрим метод последовательных приближений для нахождения решения обратной задачи (4).

Пусть задана произвольная начальная точка x^0 . Последовательность $\{x^j\}$ строится в результате следующих шагов.

Для полученной точки x^j находим $y^j = y(x^j)$ решение задачи дополненности

$$w^j = P(y^j) - Hx^j - g, \quad w^j \geq 0, \quad y^j \geq 0, \quad (y^j)^T w^j = 0.$$

Если окажется, что $H^T y^j = 0$, то пара $(x^*, y^*) = (x^j, y^j)$ будет удовлетворять соотношениям (3), (4), то есть будет получено решение обратной задачи (4). В противном случае, выбрав некоторую величину шага $\alpha_j > 0$, полагаем

$$x^{j+1} = x^j - \alpha_j H^T y^j \quad (5)$$

и переходим к следующей итерации.

Сходимость метода решения обратной задачи дополненности

Исследуем сходимость метода (5) к решению (x^*, y^*) пары задач (3), (4). В идеале траекторию построения решения обратной задачи с помощью итерационной схемы (5) можно рассматривать как начинающуюся в точке $x^0 = x(0)$ траекторию решения $x(\alpha)$ дифференциальной системы

$$\dot{x} = -H^T y(x), \quad (6)$$

где $y = y(x)$ – решение нелинейной задачи дополненности (3).

Доказана следующая теорема.

Теорема 2.

Дифференциальная система (6) при любом начальном условии $x^0 = x(0)$ имеет единственное непрерывно дифференцируемое решение $x = x(\alpha)$, определенное для всех $\alpha \in [0, +\infty)$. При этом имеет место оценка

$$\|x(\alpha) - x^*\| \leq \|x^0 - x^*\|.$$

Сложность исследования дифференциальной системы (6) обусловлена несколькими факторами. Во-первых, оператор $-H^T y(x)$ не является потенциальным, то есть не является градиентом ни для какой функции, во-вторых, оператор $-H^T y(x)$ в общем случае не является монотонным, и в-третьих, правая часть дифференциальной системы (6) не задана аналитически, а вычисляется через решение нелинейной задачи дополненности (3).

Обсудим вопрос о строении траектории $x(\alpha)$ дифференциальной системы (6). Рассмотрим ее правую часть. Решение $y = y(x)$ линейной задачи дополненности (3) можно записать в виде

$$y(x) = \begin{pmatrix} y_B \\ y_N \end{pmatrix},$$

где $y_B > 0$ – вектор, составленный из положительных компонент решения $y = y(x)$ задачи (3), образующих базис

$$B = B(x) = \{i_1, \dots, i_J\}, J \leq m,$$

вектор $y_N = 0$ составлен из небазисных компонент решения $y = y(x)$. Множество векторов $x \in R^n$, для которых решение задачи (3) имеет базис B , обозначим через X_B . Тогда дифференциальная система (6) запишется соответственно в виде

$$\dot{x} = -H_B^T y_B, x \in X_B,$$

где H_B – соответствующая y_B вырезка из матрицы H .

Заметим, что случай $y(x(\alpha)) = 0$ означает, что решение задачи дополненности (3) тривиально. В дальнейшем поэтому будем считать, что $y(x(\alpha)) \neq 0$, и, следовательно,

базис B непустой.

Перейдем теперь к вопросу о сходимости траектории $x(\alpha)$ к некоторому решению x^* обратной задачи (4). Для этого рассмотрим функцию $\|H^T y(x)\|$ вдоль решения $x(\alpha)$ дифференциальной системы (6) при начальных условиях $x(0) = x^0$. Сильная монотонность и обратная сильная монотонность отображения P (в линейном случае положительная определенность матрицы P) обеспечивают монотонность функции $\|H^T y(x)\|$ вдоль траектории $x(\alpha)$ и сходимость $y(x(\alpha))$ и $x(\alpha)$ к паре y^* и x^* решения задач (3),(4).

Справедливы следующие теорема и следствие.

Теорема 3.

Вдоль траектории $x(\alpha)$ дифференциальной системы (6) величина $\|H^T y(x)\|$ является непрерывной убывающей функцией параметра α , причем при $\alpha_2 > \alpha_1 \geq 0$ выполняется следующее неравенство

$$\|H^T y(x(\alpha_2))\| \leq \|H^T y(x(\alpha_1))\| e^{-\gamma(\alpha_2 - \alpha_1)},$$

где $\gamma > 0$ — некоторая константа.

Следствие.

Вдоль траектории дифференциальной системы (6) существуют пределы

$$x^* = \lim_{\alpha \rightarrow +\infty} x(\alpha), \quad y^* = \lim_{\alpha \rightarrow +\infty} y(x(\alpha)),$$

причем x^* — решение обратной задачи (4), а y^* — соответствующее решение задачи дополненности (3).

Список литературы

- [1] *Приоритетные результаты в области математического программирования. Часть 1.* - Информационный бюллетень Ассоциации математического программирования №9. Екатеринбург: УрО РАН, 2001.
- [2] А.И. Голиков *Об одной постановке обратной задачи нелинейного программирования.* - Обратные задачи математического программирования. М.:ВЦ РАН, 1992.
- [3] А.С. Антипин *Обратная задача оптимизации: постановка задачи и подходы к ее решению.* - Обратные задачи математического программирования. М.:ВЦ РАН, 1992.
- [4] И.В. Коннов *Метод расщепления с линейным поиском для прямо-двойственных вариационных неравенств.* - Журнал вычислительной математики и математической физики, 2003, том 43, №4, с. 518-532.
- [5] Я.М. Берцанский, М.В. Мееров *Теория и методы решения задач дополненности* - Автоматика и телемеханика, 1983, №6, с.5-31.

DECISION OF THE INVERSE NONLINEAR COMPLEMENTARY PROBLEM

A.V. Zykina

Omsk State Technical University, Omsk
e-mail: zykina@omgtu.ru

Abstract. Decision of the inverse problem is researched in work for nonlinear complementary problem. The algorithm for the decision of the inverse problem is offered. Its theoretical motivation is received.

Key words: parametric nonlinear complementary problem, inverse problem.

ЧИСЛЕННОЕ ТЕСТИРОВАНИЕ МЕТОДОВ НУЛЕВОГО ПОРЯДКА В ЗАДАЧАХ НЕВЫПУКЛОГО ПРОГРАММИРОВАНИЯ¹

Д.В. Иванов, О.В. Хамисов

Институт систем энергетики им. Мелентьева СО РАН, Иркутск

Аннотация. В статье приводятся результаты численного тестирования известных методов нулевого порядка с модифицированным алгоритмом линейного поиска, который основан на известном методе парабол.

Ключевые слова: методы нулевого порядка, метод парабол.

Введение

Обычно методы нулевого порядка применяются к задачам, в которых оптимизируемая функция не имеет явного аналитического выражения. Как правило, для вычисления значений таких функций необходимо решать нетривиальные вспомогательные задачи вычислительной математики и оптимизации. В некоторых случаях предполагается, что вычисление одного значения целевой функции может занимать от нескольких минут до нескольких часов нерерывной работы компьютера. Цель данной статьи хотя и связана с методами оптимизации, основанными только на вычислении значений целевой функции, состоит в том, чтобы протестировать известные методы нулевого порядка на задачах глобальной оптимизации. Причинами для такого тестирования послужили следующие соображения. Для разработки эффективного метода глобальной оптимизации требуется выявление определенных свойств целевой функции (и ограничений). Например, определение хорошей оценки константы Липшица [3], [12] или представление функции в виде разности двух выпуклых функций [13]. Подобные вспомогательные задачи не всегда имеют единственное решение и не всегда легко разрешимы. Эффективность методов глобальной оптимизации часто зависит от возможности быстро найти хорошее локальное решение [11], что существенно может ускорить, например, работу методов ветвей и границ. Как показали численные эксперименты, некоторые методы нулевого порядка обладают возможностью не только находить хорошее локальное решение, а часто и сам глобальный оптимум.

Первоначально тестирование проводилось на одномерных многоэкстремальных задачах. В качестве метода минимизации использовался метод парабол. Оказалось, что даже для довольно сложных задач метод работает хорошо. Для многомерных задач снова применялся метод парабол в сочетании с покоординатным спуском и методом вращения координат. Проводилось сравнение с другими методами нулевого порядка. Решение многомерных задач оказалось более трудоемким, но и здесь удалось решить ряд довольно сложных примеров. Результаты тестирования описаны в последующих двух параграфах.

¹Работа поддержана грантом РФФИ N 03-01-00518

1. Одномерная минимизация. Метод квадратичной интерполяции.

Рассматривается следующая задача одномерной минимизации.

$$\begin{aligned} f(x) &\rightarrow \min, \\ x \in P \subset R, \quad P &= \{x \in R : \alpha \leq x \leq \beta\}. \end{aligned} \quad (1)$$

Для решения этой задачи использовался метод нулевого порядка, известный как метод квадратичной интерполяции или метод Пауэлла [7], или метод парабол [5]. На каждой итерации данного метода по трем заданным опорным точкам, лежащим на графике функции $f(x)$, строится квадратный трехчлен (парабола) и аналитически вычисляется его минимум. Полученная точка используется в качестве текущего приближения к точке минимума в задаче (1), а также в качестве новой опорной точки, замещающей наихудшую (в смысле значения $f(x)$) из предыдущих трех опорных точек. Далее процесс повторяется с новыми опорными точками. Приведем алгоритм метода.

Шаг 1. Выбрать $\varepsilon > 0$, положить $k = 0$. Выбрать точки $x_1^k, x_2^k, x_3^k \in [\alpha, \beta]$ такие, что $x_1^k < x_2^k < x_3^k$ и выполняется

$$f(x_1^k) \geq f(x_2^k) \leq f(x_3^k). \quad (2)$$

Положить $z = x_2^k$.

Шаг 2. Найти точку минимума параболы \bar{x}^k по формуле:

$$\begin{aligned} \bar{x}^k &= \frac{1}{2} \left(x_1^k + x_2^k - \frac{a_1^k}{a_2^k} \right), \\ a_1^k &= \frac{f(x_2^k) - f(x_1^k)}{x_2^k - x_1^k}, \quad a_2^k = \frac{1}{x_3^k - x_2^k} \left(\frac{f(x_3^k) - f(x_1^k)}{x_3^k - x_1^k} - a_1^k \right). \end{aligned}$$

Шаг 3. Проверка оптимальности. Если выполняется $|\bar{x}^k - z| < \varepsilon$, то процедура окончена, точка минимума $x^* = \bar{x}^k$. Иначе перейти к шагу 4.

Шаг 4. Выбрать $\hat{x} = \arg \min_{x=x_1^k, x_2^k, x_3^k} f(x)$. Положить $z = \bar{x}^k$. Обозначить точки $\hat{x}, x_2^k, \bar{x}^k$ в порядке возрастания, как x_1^k, x_2^k, x_3^k . Положить $k = k + 1$, перейти к шагу 2.

Нахождение опорных точек, необходимых для старта алгоритма, в случае многоэкстремальной функции, может оказаться нетривиальной задачей. Представляется логичным перед стартом метода парабол применять некую процедуру, которая, по возможности, находила бы несколько различных опорных троек, обеспечивая возможность мультистарта. Два способа реализации такой процедуры использовались при численном тестировании. Для каждого из них задается максимальное число вычислений целевой функции N . В первом варианте N точек равномерно располагаются на отрезке минимизации, затем берется три точки слева, начиная с первой, затем начиная со второй и т.д. и каждая такая тройка проверяется на условие (2). Во втором варианте отрезок минимизации делится пополам и точка деления вместе с концами отрезка образует первую тройку. Затем делится пополам каждая из полученных частей отрезка, в качестве троек выступают точки деления и концы тех частей отрезка, которые эти точки делят. Деление отрезка и проверка

получаемых троек точек продолжают, пока не будет превышено заданное число вычислений функции. В ходе численного эксперимента выяснилось, что наиболее эффективным является первый способ. Максимальное количество отрезков разбиения равнялось 10. Из них в среднем на 2-3 отрезках выполнялось условие (2). Затем, на каждом из отобранных отрезков стартовал метод парабол.

Данный алгоритм был протестирован на серии задач одномерной минимизации из [10]. Результаты представлены в Таблице 1. Тестирование проводилось на компьютере P-IV/2.4Gh/512Mb. Время решения каждой из задач пренебрежительно мало. Все задачи кроме первой были успешно решены. Для первой задачи не удалось найти стартовую тройку, удовлетворяющую условиям (2). Первые четыре столбца в таблице Таблицы 1 содержат данные задачи. В пятом столбце указано общее количество вычислений значений целевой функции - k_f . В шестом столбце приведены значения относительной погрешности в процентах - $\varepsilon\%$. Как следует из Таблицы 1 глобальный минимум находится за малое количество итераций. Достаточно заметить, что, например, что функция $-\sum_{k=1}^5 k \sin[(k+1)x+k]$ имеет 20 локальных минимумов на отрезке $[-10, 10]$. Глобальный минимум найден за 15 вычислений значений функции, что сопоставимо с работой метода золотого сечения.

Таблица 1.

функция	интервал	x^*	$f(x^*)$	k_f	$\varepsilon\%$
$\frac{1}{6}x^6 - \frac{52}{25}x^5 + \frac{39}{80}x^4$	$[-1.5, 11]$	—	—	10	—
$\sin x + \sin \frac{10x}{3}$	$[2.7, 7.5]$	5.146	-1.9	23	0
$-\sum_{k=1}^5 k \sin[(k+1)x+k]$	$[-10, 10]$	5.791	-12.031	15	0
$-(16x^2 - 24x + 5)e^{-x}$	$[1.9, 3.9]$	2.868	-3.85	14	0
$(3x - 1.4) \sin 18x$	$[0, 1.2]$	0.967	-1.489	23	0
$-(x + \sin x)e^{-x^2}$	$[-10, 10]$	0.68	-0.824	18	0
$-x \sin x$	$[0, 10]$	7.979	-7.917	18	0
$2 \cos x + \cos 2x$	$[-1.57, 6.28]$	2.094	-1.5	23	0
$\sin^3 x + \cos^3 x$	$[0, 6.28]$	4.712	-1	27	0
$-x^{2/3} + (x^2 - 1)^{1/3}$	$[.001, .99]$	0.707	-1.587	12	0
$-e^{-x} \sin(2\pi x)$	$[0, 4]$	0.225	-0.789	27	0
$\frac{x^2-5x+6}{x^2+1}$	$[-5, 5]$	2.415	-0.036	21	0
$2(x-3)^2 + e^{\frac{x^2}{2}}$	$[-3, 3]$	1.591	7.516	14	0
$x^6 - 15x^4 + 27x^2 + 250$	$[-4, 4]$	-3	-7	24	0
$\sin 3x - x - 1$	$[0, 6.5]$	5.873	-7.816	20	0

В заключении этого параграфа можно сказать, что метод парабол является достаточно эффективным методом одномерной многоэкстремальной оптимизации.

2. Численное тестирование методов нулевого порядка на многомерных задачах невыпуклой оптимизации.

В данном параграфе рассматривается 7 методов нулевого порядка: метод конфигураций или метод Хука-Дживса, симплексный метод или метод деформируемого многогранника или метод Нелдера-Мида, метод покоординатного спуска в сочетании методом парабол, метод вращения координат или метод Розенброка, метод вращения координат в сочетании с методом парабол, метод наилучшей пробы и метод Пауэлла в сочетании с методом парабол. Большинство из этих методов описаны в различных монографиях по методам оптимизации [1], [4], [6],[8],[9],[2] (см. также [5],[7]). На основании результатов предыдущего параграфа были сделаны следующие модификации стандартных методов нулевого порядка. В методах покоординатного спуска, вращения координат и Пауэлла для решения соответствующей задачи одномерной оптимизации использовался метод парабол. Тестовые задачи выбирались из набора стандартных тестовых задач, опубликованных в различных статьях и монографиях. Вычисления проводились на компьютере P-IV/2.4Gh/512Mb. Время решения в каждом случае не превысило 2 секунд. Ниже приводится список использованных тестовых задач. Многие целевые функции, имеют собственное название, чаще всего по имени автора, впервые ее предложившего в качестве тестовой. В статье сохранены эти обозначения, в тексте все задачи идут под соответствующими русскими названиями, а в таблицах используются их английские аналоги. Через $D(f(x))$ обозначены области задания тестовых функций.

1. Функция Бранина (Branin).

$$f(x) = \left(x_2 - \frac{5.1x_1^2}{4\pi^2} + \frac{5x_1}{\pi} - 6 \right)^2 + 10\left(1 - \frac{1}{8\pi}\right) \cos x_1 + 10,$$
$$D(f(x)) = [-5, 10] \times [0, 15].$$

Имеет два глобальных минимума в заданной области:

$$x^{1*} = (3.141593, 2.275), x^{2*} = (-3.141593, 2.275),$$
$$f^* = 0.397667.$$

2. Функция Треккани (Трессани).

$$f(x) = x_1^4 + 4x_1^3 + 4x_1^2 + x_2^2,$$
$$D(f(x)) = [-3, 3] \times [-3, 3].$$

Имеет два глобальных минимума в заданной области: $x^{1*} = (0, 0)$, $x^{2*} = (2, 0)$, $f^* = 0$.

3. Функция Шуберта (Shubert).

$$f(x) = \left(\sum_{i=1}^5 i \cos((i+1)x_1 + i) \right) \cdot \left(\sum_{i=1}^5 i \cos((i+1)x_2 + i) \right).$$
$$D(f(x)) = [-10, 10] \times [-10, 10].$$

Функция имеет 760 локальных минимумов на указанном параллелепипеде, из них 18 глобальных, $f^* = -186.730909$

4. Функция «трехгорбый верблюд» (3-hump camel).

$$f(x) = 2x_1^2 - 1.05x_1^4 + \frac{x_1^6}{6} - x_1x_2 + x_2^2,$$

$$D(f(x)) = [-3, 3] \times [-3, 3].$$

Имеет 3 минимума на $D(f)$, глобальный минимум единственный: $x^* = (0, 0)$, $f^* = 0$.

5. Функция «шестигорбый верблюд» (6-hump camel).

$$f(x) = 4x_1^2 - 2.1x_1^4 + \frac{x_1^6}{3} + x_1x_2 - 4x_2^2 + 4x_2^4,$$

$$D(f(x)) = [-3, 3] \times [-1.5, 1.5].$$

На заданной области имеет 6 минимумов, из них 2 глобальных:

$$x^{1*} = (-0.089842, 0.712656), \quad x^{2*} = (0.089842, -0.712656),$$

$$f^* = -1.031628.$$

6. Функция Розенброка (Rosenbrock).

$$f(x) = 100(x_1^2 - x_2)^2 + (x_1 - 1)^2,$$

$$D(f(x)) = [-5, 5] \times [-5, 5].$$

Глобальный минимум: $x^* = (1, 1)$, $f^* = 0$.

7. Первая функция Ге и Кина (Ge - Qin - 1).

$$f(x) = \frac{\pi}{n} \left(10 \sin^2(\pi x_1) + \sum_{i=1}^{n-1} (x_i - 1)^2 [1 + 10 \sin^2(\pi x_{i+1})] + (x_n - 1)^2 \right),$$

$$x_i \in [-10, 10], \quad i = \overline{1, n}$$

Функция имеет около 5^n локальных минимумов на $D(f)$, известный глобальный минимум $x_i = 1, \overline{1, n}$, $f^* = 0$.

8. Вторая функция Функция Ге и Кина (Ge-Qin - 2).

$$f(x) = \frac{\pi}{n} \left(10 \sin^2(\pi y_1) + \sum_{i=1}^{n-1} (y_i - 1)^2 [1 + 10 \sin^2(\pi y_{i+1})] + (y_n - 1)^2 \right),$$

$$y_i = \frac{x_i - 1}{4} + 1,$$

$$x_i \in [-10, 10], \quad i = \overline{1, n}$$

Функция имеет около 5^n локальных минимумов на $D(f)$, известный глобальный минимум $x_i = 1, \overline{1, n}$, $f^* = 0$.

9. Третья функция Ге и Кина (Ge-Qin – 3).

$$f(x) = \frac{1}{10} \sin^2(3\pi x_1) + \frac{1}{10} \sum_{i=1}^{n-1} (x_i - 1)^2 (1 + \sin^2(3\pi x_{i+1})) + \\ + \frac{1}{10} (x_n - 1)^2 (1 + \sin^2(2\pi x_n)), \\ x_i \in [-10, 10], i = \overline{1, n}$$

Функция имеет около 30^n локальных минимумов на $D(f)$, глобальный минимум единственный: $x_i = 1, \overline{1, n}, f^* = 0$.

Результаты тестирования приведены в таблицах 2-5. Обозначение Ge-Qin-1-5 соответствует задаче минимизации первой функции Ге и Кина с 5 переменными, Ge-Qin-1-10 - первой функции Ге и Кина с 50 переменными, Ge-Qin-1-100 - первой функции Ге и Кина со 100 переменными. Аналогичные смысл имеют обозначения, связанные со второй и третьей функциями Ге и Кина. Метод парабол и метод парабол с ВК в Таблице 4 означают метод покоординатного спуска и метод вращения координат в комбинации с методом парабол, применяемым для соответствующей одномерной минимизации. Максимальное количество вычислений значений целевой функции было равно 10000. Колонка $\bar{f} - f^*$ означает разность между полученным наилучшим (\bar{f}) и минимальным (f^*) значениями целевой функции. В колонке $\|\bar{x} - x^*\|$ указано расстояние от найденной точки \bar{x} до точки глобального минимума x^* . В колонке "число выч." приведено количество вычислений целевой функции для каждого метода и задачи. Символ X означает, что количество вычислений целевой функции превысило допустимый предел 10000. Жирным шрифтом выделены решенные задачи.

Таблица 2.

Задача	Метод наилучшей пробы			Метод вращения координат		
	$\bar{f} - f^*$	$\ \bar{x} - x^*\ $	число выч.	$\bar{f} - f^*$	$\ \bar{x} - x^*\ $	число выч.
Branin	0.0002	0.0006	133	2.6647	1.7352	15
Treccani	1.1E – 7	0	133	0	0	7
Shubert	0.0004	0.0004	133	0.0002	0.0002	187
3-hump camel	1.98E – 7	0.0004	127	0.2986	1.9537	167
6-hump camel	9.8E – 7	0.0004	133	0.0008	0.01	48
Rosenbrock	1.5058	4.147	109	0.9994	3.1621	16
Ge-Qin-1-5	153.8184	15.6442	85	28.2743	6.7082	21
Ge-Qin-1-50	78.5291	35.3497	115	78.4927	35.3341	3801
Ge-Qin-1-100	50.2595	39.9962	115	0	0	X
Ge-Qin-2-5	5.2313	11.5057	1723	1.98E – 5	0.0222	626
Ge-Qin-2-50	9.4896	39.3182	3433	11.978	54.927	1986
Ge-Qin-2-100	7.3456	34.1711	2563	3.1094	39.586	5001
Ge-Qin-3-5	12.626	11.0828	331	0.0989	0.9889	996
Ge-Qin-3-50	387.9169	35.2673	2749	117.207	34.1917	X
Ge-Qin-3-100	234.9288	39.9842	775	0.0989	0.9889	X

Таблица 3.

Задача	Метод конфигураций			Симплексный метод		
	$\bar{f} - f^*$	$\ \bar{x} - x^*\ $	число выч.	$\bar{f} - f^*$	$\ \bar{x} - x^*\ $	число выч.
Branin	0.0003	0.0094	74	0.0002	0.0011	59
Treccani	0	0	48	1.1E - 7	0.0001	52
Shubert	63.1625	0.868	75	0.0002	0.0002	58
3-hump camel	0	0	48	1.7E - 7	0.0004	61
6-hump camel	0	0	23	7E - 7	0.0005	52
Rosenbrock	0.2874	0.9507	373	1.5E - 7	0.0003	150
Ge-Qin-13-5	0	0	129	153.599	15.6162	126
Ge-Qin-13-50	0	0	1355	78.5104	35.3419	1745
Ge-Qin-13-100	0	0	2403	3.4007	10.1875	X
Ge-Qin-15-5	3.1416	8.9443	128	3.1096	8.8543	211
Ge-Qin-15-50	3.3041	28.6827	1342	11.5956	54.2295	X
Ge-Qin-15-100	3.1416	40	2402	0.0001	0.2523	X
Ge-Qin-19-5	0.5	2.2361	123	3.8163	6.1725	1221
Ge-Qin-19-50	5	7.0711	1303	122.4401	34.9737	7270
Ge-Qin-19-100	10	10	2302	0.8202	2.6932	X

Таблица 4.

Задача	Метод парабол			Метод парабол с ВК		
	$\bar{f} - f^*$	$\ \bar{x} - x^*\ $	число выч.	$\bar{f} - f^*$	$\ \bar{x} - x^*\ $	число выч.
Branin	0.0003	0.0141	462	0.0002	0.001	691
Treccani	5.1E - 8	1.9999	107	2.3E - 7	0.0002	335
Shubert	2.5E - 7	0.0002	157	2.48E - 7	0.0002	258
3-hump camel	0.2986	1.9531	109	0.2986	1.9532	161
6-hump camel	0	0	171	0	0	224
Rosenbrock	0.5339	2.1249	109	2.6E - 5	0.0095	2866
Ge-Qin-13-5	3.03E - 7	6.9E - 5	239	1.2E - 8	0.0001	476
Ge-Qin-13-50	3E - 8	7E - 5	2309	9E - 9	4E - 5	3722
Ge-Qin-13-100	1.5E - 8	7E - 5	4609	1E - 8	5.8E - 5	7205
Ge-Qin-15-5	0	0	262	0	0	632
Ge-Qin-15-50	0	0	2332	0	0	4898
Ge-Qin-15-100	0	0	4636	0	0	9410
Ge-Qin-19-5	0.1436	1.1883	257	0.1134	1.0585	1138
Ge-Qin-19-50	0.1436	1.1883	2327	0.1435	1.1884	3720
Ge-Qin-19-100	0.1436	1.1883	4627	0.1435	1.1885	7219

Таблица 5.

Задача	Метод Пауэлла		
	$\bar{f} - f^*$	$\ \bar{x} - x^*\ $	число выч.
Branin	0.0002	7.9E - 6	222
Treccani	5.1E - 8	1.9999	160
Shubert	2.5E - 7	0.0001	229
3-hump camel	0	0	171
6-hump camel	0	0	164
Rosenbrock	0	0	811
Ge-Qin-13-5	3.03E - 7	6.9E - 5	294
Ge-Qin-13-50	3E - 8	7E - 5	2365
Ge-Qin-13-100	1.5E - 8	7E - 5	4609
Ge-Qin-15-5	0	0	329
Ge-Qin-15-50	0	0	2399
Ge-Qin-15-100	0	0	4701
Ge-Qin-19-5	0.1436	1.1883	336
Ge-Qin-19-50	0.1436	1.1883	2396
Ge-Qin-19-100	0.1436	1.1883	4705

По результатам тестирования можно сказать следующее. Три метода имеют заметное преимущество по сравнению с остальными: метод парабол, метод парабол с вращением координат и метод Пауэлла. Несколько неожиданным является довольно удачное поведение метода наилучшей пробы на двумерных задачах относительно метода вращения координат, метода конфигураций и симплексного метода. Таким образом, применение метода парабол в методах нулевого порядка может дать существенный выигрыш при решении задач невыпуклой оптимизации.

Список литературы

1. Базара М., Шетти К. *Нелинейное программирование. Теория и алгоритмы*. — М.: Мир, 1982.
2. Васильев Ф.П. *Численные методы решения экстремальных задач*. — М.: Наука, 1980.
3. Жиглявский А.А., Жилинскас А.Г. *Методы поиска глобального экстремума*. М.: Наука, 1991
4. Ларичев О.И., Горвиц Г.Г. *Методы поиска локального экстремума овражных функций*. М.: Наука, 1990
5. Лесин В.В., Лисовец Ю.П. *Основы методов оптимизации*. — М: Издательство МАИ, 1998.

6. Мину М. *Математическое программирование. Теория и алгоритмы.* — М.: Наука, 1990.
7. Пантелеев А.В., Летова Т.А. *Методы оптимизации в примерах и задачах: Учебное пособие.* — М.: Высш. шк., 2002.
8. Поляк Б.Т. *Введение в оптимизацию.* — М.: Наука, 1983.
9. Сухарев А.Г., Тимохов А.В., Федоров В.В. *Курс методов оптимизации.* — М.: Наука, 1986.
10. Hansen P., Jaumard B. *Univariate Lipschitz Optimization.*// Handbook on Global Optimization;/ eds. Pardalos P., Horst R. — Kluwer Academic Publishers, 1995. — P. 217-289.
11. Horst R., Tuy H. *Global optimization. Deterministic approaches.* Berlin. Springer-Verlag, 1996
12. Pinter J. *Global optimization in action.* Dordrecht. Kluwer Academic Publishers, 1995
13. Tuy H. *D.C. optimization: Theory, methods and applications.* // Handbook of global optimization. Dordrecht. Kluwer Academic Publishers, 1995, pp.149-216

ДВУХЭТАПНАЯ ЗАДАЧА СТОХАСТИЧЕСКОГО ПРОГРАММИРОВАНИЯ

О. Н. Канева

Омский Государственный Технический Университет, Омск
e-mail: okaneva@yandex.ru

Аннотация. В статье предлагается новая постановка двухэтапной задачи стохастического программирования, которая отличается от классической постановки задачей второго этапа. Дана экономическая интерпретация, рассмотрены условия разрешимости.

Ключевые слова: стохастическое программирование, задача дополненности, стохастический квазиградиент.

Введение

Во многих практических задачах управления и планирования встречаются двухэтапные задачи стохастического программирования [1]. Эти задачи естественным образом описывают ситуации, когда решение должно быть принято до наблюдения реализаций некоторых случайных факторов. В дальнейшем нередко появляется возможность скорректировать предварительно принятое решение, учитывая при этом накопленную информацию о реализации случайных факторов таким образом, чтобы компенсировать возможные невязки. В соответствии с этим целесообразно рассматривать нахождение предварительного плана и процедуру компенсации определяемых им невязок как два этапа одной задачи. Стратегия, выбираемая на первом этапе, называется планом первого этапа, а стратегия корректировки этого плана - планом второго этапа.

Выбор оптимального плана в задачах подобного рода определяется тем, каким образом оцениваются невязки в условиях задачи и каким образом оценка невязки сопоставляется с затратами на реализацию предварительного плана первого этапа. Традиционно в качестве критерия оптимальности в двухэтапных задачах стохастического программирования рассматривают математическое ожидание суммарных затрат на первом и втором этапах.

Трудности, с которыми связан анализ двухэтапных задач, в значительной степени определяются необходимостью такого выбора предварительного плана первого этапа разрешимой задачи, который гарантировал бы существование компенсации последующих невязок для достаточно широкого множества значений случайных факторов.

В статье предложена новая постановка двухэтапной задачи стохастического программирования, которая отличается от остальных постановкой задачи второго этапа. Дано содержательное обоснование.

Для простоты оформления скалярных произведений в статье опускается знак транспонирования в первом сомножителе.

1. Общая постановка двухэтапной задачи стохастического программирования

Рассмотрим следующую задачу стохастического программирования:

$$\min_x M_\omega \{ \varphi(\omega, x) \}, \quad (1)$$

$$g_i(\omega, x) \leq b_i(\omega), i = 1, 2, \dots, m, \quad (2)$$

$$x \in X. \quad (3)$$

Здесь

ω — случайная величина, которая определяет состояние природы и является элементом некоторого вероятностного пространства (Ω, F, P) ;

$\varphi(\omega, x)$ — скалярная случайная функция, описывающая затраты, связанные с реализацией плана $x = (x_1, \dots, x_n)$ в состоянии природы ω ;

$g_i(\omega, x), i = 1, \dots, m$ — случайные функции, которые вместе со случайным m -мерным вектором $b(\omega)$ описывают ограничения задачи;

X — некоторое множество из R^n .

Заметим, что множество X содержит ограничения, не зависящие от ω , то есть детерминированные. Тогда предварительным планом первого этапа является некоторое допустимое решение, удовлетворяющее условиям (3), то есть решение, не учитывающее состояние природы ω .

После того, как ω становится известным, в ограничениях (2) могут возникнуть невязки, которые ликвидируются на втором этапе выбором вектора коррекции $y = (y_1, \dots, y_r)$ из следующих условий:

$$g_i(\omega, x) - b_i(\omega) \leq h_i(\omega, y), i = 1, \dots, m, \quad (4)$$

$$y \in Y, \quad (5)$$

где $h_i(\omega, y), i = 1, \dots, m$ — случайные функции, а Y — некоторое множество из R^r . Ограничение (4) характеризует возможность скорректировать за счет $y \in Y$ невязки, возникающие при реализации плана x первого этапа.

Кроме того, за нарушение условий (2) устанавливается штраф, зависящий от координат вектора y , компенсирующего невязку. Штраф задается величиной

$$\psi(\omega, y). \quad (6)$$

В этом случае задача второго этапа при принятом выше x и известном значении ω состоит в минимизации затрат (6) при условиях (4),(5). Тогда двухэтапная задача стохастического программирования запишется в следующем виде:

$$\min_x M_\omega \{ \varphi(\omega, x) + \min_y [\psi(\omega, y) | g(\omega, x) - b(\omega) \leq h(\omega, y), y \in Y] \},$$

$$x \in X.$$

Здесь $g(\omega, x), h(\omega, y)$ — m -мерные вектора, компонентами которых являются функции $g_i(\omega, x), h_i(\omega, y)$ соответственно.

2. Новая постановка двухэтапной задачи стохастического программирования

Для задачи (1) – (3) запишем ограничения задачи второго этапа в следующем виде:

$$g(\omega, x) - b(\omega) \leq P(\omega)y, y \geq 0, \quad (7)$$

$$y(g(\omega, x) - b(\omega)) = yP(\omega)y. \quad (8)$$

Задача (7),(8) является линейной задачей дополнителности относительно переменных y при фиксированных x и ω . Если неравенства в (7) понимать покомпонентно как сопряженные неравенства, то условия (8) понимаются как классические условия дополняющей нежесткости: в каждой паре сопряженных неравенств есть хотя бы одно равенство.

Для задачи (7),(8) известно [3], что:

если $P(\omega) \geq 0$ при любом ω , то для каждого вектора $g(\omega, x) - b(\omega)$ при любом x и ω существует по крайней мере одно решение тогда и только тогда, когда все диагональные элементы $P(\omega)$ положительны;

если матрица $P(\omega)$ положительно полуопределена и система (7) совместна, то для каждого вектора $g(\omega, x) - b(\omega)$ при любом x и ω существует по крайней мере одно решение;

если матрица $P(\omega)$ имеет положительные главные миноры, то при любом векторе $g(\omega, x) - b(\omega)$, а значит при любых x и ω существует единственное решение.

В терминах планирования производства задача (7), (8) имеет содержательную интерпретацию, аналогичную интерпретации задачи (4), (5), более того, условие (8) выводит ограничения (2) исходной задачи на предел разрешимости.

Штраф за реализацию плана компенсации y будем задавать в виде функции

$$\psi(\omega, x, y) = y(g(\omega, x) - b(\omega)). \quad (9)$$

Тогда задача второго этапа при принятом выше x и известном значении ω состоит в минимизации затрат (9) при условиях (7), (8).

Лемма. Задача второго этапа (7) – (9) при каждом x и ω имеет решение $y = y(\omega, x)$, в случаях:

- 1) если $P(\omega) \geq 0$ и все диагональные элементы $P(\omega)$ положительны;
- 2) если $P(\omega)$ положительно полуопределена и система (7) совместна;
- 3) если $P(\omega)$ имеет положительные главные миноры.

Доказательство.

Допустимая область, определяемая условиями (7), (8) не пуста при любых x и ω . Это обеспечивается каждым из условий 1), 2), 3) леммы.

Целевая функция (9) — ограничена на допустимой области (7), (8). Действительно, по условию (8) целевая функция (9) совпадает с функцией $yP(\omega)y$. Тогда из любого условия 1), 2), 3) леммы для любых $y \geq 0$ получаем условие ограниченности целевой функции $\psi(\omega, x, y) \geq 0$.

Лемма доказана.

3. Решение задачи второго этапа

Задача второго этапа (7) – (9) при фиксированных x и ω в общем случае является задачей нелинейного программирования. Однако специфика ограничений (7), (8) позволяет записать эту задачу в несколько другой форме.

При фиксированных x и ω задача дополнителности (7), (8), в случае ее разрешимости, эквивалентна следующей задаче квадратичного программирования

$$\min_y \{y(P(\omega)y - (g(\omega, x) - b(\omega)))\},$$

$$g(\omega, x) - b(\omega) \leq P(\omega)y, y \geq 0.$$

Тогда задача второго этапа (7) – (9) является многокритериальной задачей последовательной оптимизации. В этом случае решение задачи (7) – (9) можно записать в виде паретовской свертки двух критериев [4]:

$$\begin{aligned} \min_y \{ & \alpha_1 y(g(\omega, x) - b(\omega)) + \alpha_2 y(P(\omega)y - (g(\omega, x) - b(\omega))) \}, \\ & g(\omega, x) - b(\omega) \leq P(\omega)y, y \geq 0, \\ & \alpha_1 > 0, \alpha_2 > 0. \end{aligned} \quad (10)$$

Зафиксируем в задаче (10) параметры ω , $\alpha_1 > 0$ и $\alpha_2 > 0$, и выпишем функцию Лагранжа :

$$L(y, u) = \alpha_1 y q(\omega, x) + \alpha_2 y(P(\omega)y - q(\omega, x)) - u(P(\omega)y - q(\omega, x)), \quad (11)$$

где $q(\omega, x) = g(\omega, x) - b(\omega)$.

Теорема. Пусть, при каждом ω , функции $g_i(\omega, x)$, $i = 1, 2, \dots, m$ выпуклы и $P(\omega)$ — неособенная положительно полуопределенная матрица. Тогда, при каждом x и ω , существует седловая точка $(y(\omega, x), u(\omega, x))$ функции (11) при $y \geq 0$, $u \geq 0$ и верно следующее соотношение

$$u(\omega, x) = (\alpha_1 + \alpha_2)y(\omega, x) = (\alpha_1 + \alpha_2)P^{-1}q(\omega, x). \quad (12)$$

Доказательство.

Согласно лемме, при сделанных предположениях, задача второго этапа (7) – (9) при фиксированных x и ω имеет, по крайней мере, одно решение, значит разрешима и задача (10).

Из разрешимости задачи (10) следует существование седловой точки $(y(\omega, x), u(\omega, x))$. Для нахождения этой точки выпишем двойственную по Лагранжу задачу

$$\max_u \{ \theta(u) | u \geq 0 \},$$

где $\theta(u) = \inf_y \{ L(y, u) | y \geq 0 \}$.

Заметим, что при фиксированном u функция Лагранжа выпукла и достигает минимума в точке, удовлетворяющей равенству

$$y = \frac{1}{2\alpha_2}u - \frac{1}{2\alpha_2}(\alpha_1 - \alpha_2)P^{-1}q(\omega, x). \quad (13)$$

Подставляя полученное выражение в $\theta(u)$ получим альтернативную форму двойственной задачи

$$\begin{aligned} \max_u \{ & -\frac{1}{4\alpha_2}uPu + \frac{1}{2\alpha_2}(\alpha_1 + \alpha_2)uq - \frac{1}{4\alpha_2}(\alpha_1 - \alpha_2)^2qPq \}, \\ & u \geq 0. \end{aligned}$$

Так как $\frac{1}{4\alpha_2} \geq 0$, то максимизируемая функция вогнута и точка максимума находится из соотношения

$$-\frac{1}{2\alpha_2}Pu + \frac{1}{2\alpha_2}(\alpha_1 + \alpha_2)q = 0.$$

В результате получаем

$$u = (\alpha_1 + \alpha_2)P^{-1}q. \quad (14)$$

Подставляя выражение (14) в (13) получаем требуемое соотношение (12).

Теорема доказана.

Следствие. Точка $(y(\omega, x), u(\omega, x), \nu)$, компоненты которой при $y \geq 0, u \geq 0$ и $\nu > 0$ связаны соотношением

$$u(\omega, x) = (1 + \nu)y(\omega, x) = (1 + \nu)P^{-1}q(\omega, x), \quad (15)$$

является седловой точкой функции Лагранжа задачи (7) – (9).

Доказательство.

Выпишем функцию Лагранжа для задачи (7) – (9)

$$\bar{L}(y, u, \nu) = yq(\omega, x) - u(P(\omega)y - q(\omega, x)) + \nu y(P(\omega)y - q(\omega, x)), \quad (16)$$

где $q(\omega, x) = g(\omega, x) - b(\omega)$.

При $\alpha_1 = 1$ и $\alpha_2 = \nu$ функции (11) и (16) совпадают при каждом x и ω . Отсюда следует, что точка $(y(\omega, x), u(\omega, x), \nu)$ является седловой для (16) и выполняется соотношение (15). Следствие доказано.

4. Алгоритм решения двухэтапной задачи стохастического программирования

Запишем двухэтапную задачу стохастического программирования в виде

$$\min_x \{F(x) | x \in X\}, \quad (17)$$

где $F(x) = M_\omega \{f^0(x, y, \omega)\}$, а

$$f^0(\omega, x, y) = \varphi(\omega, x) + \min_y [yq(\omega, x) | q(\omega, x) \leq P(\omega)y, yq(\omega, x) = yP(\omega)y, y \geq 0].$$

Тогда $f^0(\omega, x, y(\omega, x)) = \varphi(\omega, x) + q(\omega, x)P^{-1}(\omega)q(\omega, x)$.

Для решения задачи (17) используем метод [2], который строит последовательность точек $\{x^s(\omega)\}_{s=0,1,\dots}$ согласно формуле

$$x^{s+1} = \pi_X(x^s - \rho_s \gamma_s \xi^s), \quad s = 0, 1, \dots \quad (18)$$

Здесь x^0 — произвольная точка, $\pi_X(x)$ — оператор проектирования, ρ_s — величина шага, γ_s — нормирующий множитель, $\xi^s(\omega)$ — случайный вектор, удовлетворяющий условию:

$$F(x) - F(x^s) \geq (M_\omega(\xi^s | x^s), x - x^s) \quad (19)$$

для любой точки $x \in X$. Такой вектор называется стохастическим квазиградиентом, а процедура (18) — методом проектирования стохастических квазиградиентов.

Если $\varphi(\omega, x)$ и $g_i(\omega, x)$, $i = 1, \dots, m$ — выпуклые функции, то $f^0(\omega, x, y(\omega, x))$ выпукла по переменной x при каждом ω . Тогда для того, чтобы решить задачу (17) с выпуклым и замкнутым множеством X , следует взять

$$\xi^s = \varphi_x(\omega^s, x^s) + 2q(\omega^s, x^s)P^{-1}q_x(\omega^s, x^s), \quad (20)$$

где ω^s , $s = 0, 1, \dots$, — независимые реализации состояний ω . Можно показать, что вектор (20) удовлетворяет условию (19).

Согласно (20) вектор ξ^s метода (18) в данном случае вычисляется следующим образом:

- 1) после s -й итерации имеется предварительный план первого этапа x^s ;
- 2) наблюдается значение случайной величины ω^s в соответствии с заданным распределением или путем "проигрывания" определенных сценариев;
- 3) вычисляется направление для итерационного процесса по формуле (20).

Список литературы

- [1] Д.Б. Юдин *Математические методы управления в условиях неполной информации*. М.: Сов. радио, 1974, 400 с.
- [2] Ю.М. Ермольев *Методы стохастического программирования*. М.: Наука, 1976, 240 с.
- [3] Я.М. Берщанский, М.В. Мееров *Теория и методы решения задач дополнителности*. - Автоматика и телемеханика, 1983, №6, с. 5-31.
- [4] В.В. Подиновский, В.М. Гаврилов *Оптимизация по последовательно применяемым критериям*. М.: Сов. радио, 1975, 192 с.

THE TWO-STAGED PROBLEM OF STOCHASTIC PROGRAMMING

O.N. Kaneva

Omsk State Technical University, Omsk
e-mail: okaneva@yandex.ru

Abstract. The new formulation of two-staged problem of stochastic programming, which differs from the classical formulation by the problem of the second stage, is offered in the article. Economic interpretation of such formulation is given and conditions of solvability are considered there.

Key words: stochastic programming, complementary problem, stochastic quasigradient.

РЕШЕНИЕ ОДНОЙ ЗАДАЧИ ВЫБОРА ИЗДЕЛИЙ МЕТОДОМ ВЕТВЕЙ И ГРАНИЦ ¹

Р.М. Ларин

Институт математики СО РАН, Новосибирск
e-mail: orlab@math.nsc.ru

Е.В. Хмель

Новосибирский госуниверситет, Новосибирск
e-mail: he-814@gorodok.net

Аннотация. Рассматривается частично целочисленная задача линейного программирования, для которой предлагается новый способ получения оценок в методе ветвей и границ. Способ основан на переходе к двойственной задаче по непрерывным переменным и на известном соотношении между минимаксом и максимином. Приводится численный пример, иллюстрирующий особенности такого подхода.

Ключевые слова: оптимизация, программирование, линейное, целочисленное, метод, ветви, границы, минимакс, двойственность.

Введение

Задача выбора типов изделий из заданного множества при ограничениях на объёмы производства и спроса и при ограниченных ресурсах потребителей может быть сформулирована как частично целочисленная задача линейного программирования [1]. Она относится к задачам, для которых к настоящему времени не удалось построить эффективных алгоритмов. Обычно применяется метод ветвей и границ [3]. Оценки находятся как решение той же задачи без условия целочисленности. При этом возникают трудности с нахождением начального рекорда. Кроме того, оценки, определяющие рекорды, могут быть достаточно грубыми. В связи с этим представляют интерес другие методы построения оценок. Один из них рассматривается в данной работе.

1. Постановка задачи

Введём обозначения:

$I = \{1, 2, \dots, n\}$ - множество номеров видов продукции (изделий, товаров);

$J = \{1, 2, \dots, m\}$ - множество номеров потребителей;

c_i^0 - затраты производителя на разработку i -го вида продукции;

c_{ij} - доход производителя от реализации единицы i -го вида продукции j -му потребителю;

a_i - ограничение на объём производства i -го вида продукции;

d_{ij} - затраты j -го потребителя на приобретение единицы i -го вида продукции;

D_j - суммарные средства j -го потребителя;

¹Работа выполнена при поддержке гранта РФФИ № 02-01-00977

b_j - ограничение на объём спроса j -го потребителя;

$x = (x_1, x_2, \dots, x_n)$ - переменные выбора: $x_i = 1$ - если i -й товар производится, $x_i = 0$ - если i -й товар не производится;

y_{ij} - объём производства i -го товара для j -го потребителя, $y = (y_{ij})_{i \in I, j \in J}$.

Пусть $X = \{0, 1\}^n$,

$$Y(x) = \{ y \mid \sum_{j \in J} y_{ij} \leq a_i x_i, \sum_{i \in I} y_{ij} \leq b_j, \sum_{i \in I} d_{ij} y_{ij} \leq D_j, y_{ij} \geq 0 (i \in I, j \in J) \}.$$

В этих обозначениях постановка задачи выглядит следующим образом:

$$S(x, y) = \sum_{i \in I} (-c_i^0 x_i + \sum_{j \in J} c_{ij} y_{ij}) \rightarrow \max_{x \in X} \max_{y \in Y(x)}. \quad (1)$$

Все заданные параметры неотрицательны. При этом естественно считать, что все $a_i > 0$ и $c_{ij} \leq d_{ij}$ (в затраты покупателя включена цена товара).

Для изложения метода решения воспользуемся более простой постановкой задачи, считая, что имеется только один обобщённый потребитель. Пусть параметры c_{ij} и d_{ij} зависят только от i . Обозначив $y_i = \sum_{j \in J} y_{ij}$, $b = \sum_{j \in J} b_j$, $D = \sum_{j \in J} D_j$, рассмотрим следующую задачу:

$$S(x, y) = \sum_{i \in I} (-c_i^0 x_i + c_i y_i) \rightarrow \max_{x \in X} \max_{y \in Y(x)}, \quad (2)$$

где

$$Y(x) = \{ y \mid 0 \leq y_i \leq a_i x_i (i \in I), \sum_{i \in I} y_i \leq b, \sum_{i \in I} d_i y_i \leq D \}.$$

2. Преобразование задачи

Используя линейность и непрерывную зависимость целевой функции S от y , перейдём к двойственной задаче с помощью двойственных переменных z'_i, v, w , считая x параметром. Двойственная задача примет такой вид:

$$- \sum_{i \in I} c_i^0 x_i + \sum_{i \in I} a_i x_i z'_i + bv + Dw \rightarrow \min_{(z'_i), v, w}$$

при условиях $z'_i + v + d_i w \geq c_i, z'_i \geq 0 (i \in I), v \geq 0, w \geq 0$.

Нетрудно показать, что эту задачу можно записать так:

$$\Phi(x, u) = \sum_{i \in I} \{ a_i \max(0, c_i - v - d_i w) - c_i^0 \} x_i + bv + Dw \rightarrow \min_{u \geq 0}, \quad (3)$$

где $u = (v, w)$. При этом очевидно равенство

$$\max_{x \in X} \max_{y \in Y(x)} S(x, y) = \max_{x \in X} \min_{u \geq 0} \Phi(x, u). \quad (4)$$

3. Некоторые свойства задачи

Лемма 1. Область значений w можно ограничить сверху единицей.

Доказательство: Так как $c_i \leq d_i$, то при $w \geq 1$ имеем равенство $\Phi = bv + Dw$. Значит, в оптимальном решении $w \leq 1$.

Лемма 2. В оптимальном решении $v \leq v_{max} = \max_{i \in I} c_i$.

Доказательство: Пусть $v \geq v_{max}$, тогда $\Phi = bv + Dw$ и в рассматриваемой области минимум достигается при $v = v_{max}$.

Обозначим $X' = \{x \mid 0 \leq x_i \leq 1 (i \in I)\}$, $U = \{u = (v, w)^T \mid 0 \leq v \leq v_{max}, 0 \leq w \leq 1\}$.

Следствие 1. $\min_{u \geq 0} \Phi(x, u) = \min_{u \in U} \Phi(x, u)$.

Следствие 2. $\max_{x \in X'} \min_{u \in U} \Phi(x, u) = \min_{u \in U} \max_{x \in X'} \Phi(x, u)$.

Утверждение следует, например, из Теоремы 5.2 [2]: 1) функция $\Phi(x, u)$ непрерывна в ограниченной замкнутой области $X' \times U$; 2) функция $\Phi(x, u)$ выпукла по u при фиксированном векторе x и линейна, т.е. вогнута, по x при фиксированном векторе u .

Пусть $f_i(u) = a_i \max(0, c_i - v - d_i w) - c_i^0$. Тогда $\Phi(x, u) = \sum_{i \in I} f_i(u)x_i + bv + Dw$

Теорема 1. Функция $\max_{x \in X} \Phi(x, u)$ выпукла по u .

Доказательство: Функция $f_i(u)$ выпукла как максимум из линейных функций. Так как $x \geq 0$, то выпукла функция $\max_{x \in X} (\sum_{i \in I} f_i(u)x_i)$. Следовательно, выпукла и функция $\max_{x \in X} \Phi(x, u)$.

Теорема 2. Функция $\max_{x \in X} \Phi(x, u)$ кусочно-линейна по u .

Доказательство: Условиями $f_i(u) = 0 (i \in I)$ область определения функции разбивается на подобласти. При этом, если $f_i(u) > 0$, то $x_i = 1$, а если $f_i(u) < 0$, то $x_i = 0$. Значит, если $f_i(u) > 0$, то $f_i(u)x_i = a_i(c_i - v - d_i w) - c_i^0$, а при $f_i(u) < 0$ это произведение равно нулю. Доказательство очевидно.

Теорема 3. Имеет место следующая цепочка неравенств:

$$\begin{aligned} \min_{u \geq 0} \Phi(\bar{x}, u) &\leq^1 \max_{x \in X} \min_{u \geq 0} \Phi(x, u) \leq^2 \max_{x \in X'} \min_{u \geq 0} \Phi(x, u) =^3 \max_{x \in X'} \min_{u \in U} \Phi(x, u) =^4 \\ &=^4 \min_{u \in U} \max_{x \in X'} \Phi(x, u) =^5 \min_{u \in U} \max_{x \in X} \Phi(x, u). \end{aligned}$$

Доказательство: Неравенства (1) и (2) очевидны. Равенства (3) и (4) выполняются в силу Следствий 1 и 2. Равенство (5) справедливо в силу линейности функции $\Phi(x, u)$ по x при фиксированном u .

Замечание. Построены примеры задач, для которых имеет место равенство $\max_{x \in X} \min_{u \geq 0} \Phi(x, u) = \min_{u \geq 0} \max_{x \in X} \Phi(x, u)$.

4. Метод ветвей и границ для решения задачи $\max_{x \in X} \min_{u \geq 0} \Phi(x, u)$

Обозначим через X_k подмножество множества X , соответствующее вершине с номером k ($k = 0, 1, \dots$): $X_k = \{x \mid x = (\alpha_1^k, \dots, \alpha_t^k, x_{t+1}, \dots, x_n), x_i \in \{0, 1\} (i = \overline{t+1}, \overline{n})\}$, где $\alpha_i^k \in \{0, 1\} (i = \overline{1}, t)$ - заданные значения. При ветвлении полагаем на одной ветви $x_{t+1} = 1$, на другой - $x_{t+1} = 0$. При $t = 0$ имеем исходную вершину дерева ветвлений, при этом $k = 0$, а при $t = n$ - какую-то конечную вершину дерева.

Верхней оценкой в k -ой вершине является величина $\Phi_k^* = \Phi(x^k, u^k) = \min_{u \in U} \max_{x \in X_k} \Phi(x, u)$, а нижней - $\varphi(x^k) = \Phi(x^k, u(x^k)) = \min_{u \geq 0} \Phi(x^k, u)$. Для нахождения верхней оценки необходимо найти минимум по u кусочно-линейной выпуклой функции $\max_{x \in X_k} \Phi(x, u)$. Эта задача решается методом, аналогичным симплекс-методу, при этом область определения разбивается на подобласти прямыми $c_i - v - d_i w = 0$ для $i = \overline{1}, t$ и

прямыми $c_i - v - d_i w - c_i^0/a_i = 0$ для $i = \overline{t+1, n}$. Так как часть компонент вектора x может быть известна, то это, естественно, упрощает вычисление верхней оценки. Нижняя оценка находится точно так же, но при этом вектор x уже полностью задан, и поэтому область разбивается прямыми $c_i - v - d_i w = 0$, $i = \overline{1, n}$.

Обозначим $R = \varphi(x^*)$, где x^* - рекорд. Множество X_k отбрасывается, если 1) $\Phi_k^* \leq R$ или 2) $\Phi_k^* = \varphi(x^k)$. Если во втором случае $\varphi(x^k) > R$, то полагаем $x^* = x^k$, $R = \varphi(x^*)$.

Алгоритм заканчивает работу, если все множества X_k отброшены. Имеющийся к этому моменту рекорд x^* - оптимальное решение.

5. Некоторые особенности метода

Для решения исходной задачи можно применить классический метод ветвей и границ с верхней оценкой в k -ой вершине $S_k^* = S(\tilde{x}^k, \tilde{y}^k) = \max_{x \in X'_k} \max_{y \in Y(x)} S(x, y)$, где $X'_k = \{x \mid x = (\alpha_1^k, \dots, \alpha_t^k, x_{t+1}, \dots, x_n), 0 \leq x_i \leq 1 (i = \overline{t+1, n})\}$, а $\alpha_i^k \in \{0, 1\} (i = \overline{1, t})$ - заданные значения.

В этой вершине имеем цепочку равенств:

$$S_k^* = \max_{x \in X'_k} \max_{y \in Y(x)} S(x, y) = \max_{x \in X'_k} \min_{u \in U} \Phi(x, u) = \min_{u \in U} \max_{x \in X'_k} \Phi(x, u) = \min_{u \in U} \max_{x \in X_k} \Phi(x, u) = \Phi_k^*.$$

То есть верхние оценки в классическом методе и в методе, использующем соотношения двойственности, совпадают. Но существует принципиальное различие: вектор $\tilde{x}^k \in X'_k$ может быть нецелочисленным, а решение $x^k \in X_k$ - всегда целочисленное. Это означает, что в случае классического метода можно довольно долго не иметь рекорда, а если он уже есть, то меняться может редко. В то время как в предлагаемом методе нетривиальный рекорд определяется уже в начальной вершине, и он может меняться в любой вершине. Этот факт позволяет существенно ускорить нахождение оптимального решения. Проиллюстрируем это на примере.

Пусть

$$\begin{aligned} a_1 = 2, \quad c_1 = 2, \quad d_1 = 3, \quad c_1^0 = 1, \quad b = 3, \\ a_2 = 2, \quad c_2 = 3, \quad d_2 = 5, \quad c_2^0 = 2, \quad D = 5, \\ a_3 = 2, \quad c_3 = 6, \quad d_3 = 8, \quad c_3^0 = 8. \end{aligned}$$

Применение классического метода ветвей и границ.

В нулевой вершине $S_0^* = \frac{55}{16}$, при этом $\tilde{x}^0 = (0, 0, \frac{5}{16})$, $\tilde{y}^0 = (0, 0, \frac{5}{8})$. Ветвим, рассматривая при $k = 1$ множество точек $X'_1 = \{x \mid x_1 = 1, 0 \leq x_2 \leq 1, 0 \leq x_3 \leq 1\}$. Имеем $S_1^* = \frac{39}{16}$, $\tilde{x}^1 = \tilde{x}^0$. При $k = 2$ имеем $X'_2 = \{x \mid x_1 = 1, x_2 = 1, 0 \leq x_3 \leq 1\}$, $S_2^* = \frac{7}{16}$, $\tilde{x}^2 = \tilde{x}^1$. Наконец, при $k = 3$ множество $X'_3 = \{(1, 1, 1)\}$, и имеем первый рекорд $x^* = (1, 1, 1)$, $R = S_3^* = \frac{1}{4}$. Далее последовательно получаем:

$$\begin{aligned} k = 4, \quad X'_4 = \{(1, 1, 0)\}, \quad x^* = \tilde{x}^4 = (1, 1, 0), \quad R = S_4^* = \frac{7}{3}; \\ k = 5, \quad \alpha_1^5 = 1, \quad \alpha_2^5 = 0, \quad \tilde{x}^5 = \tilde{x}^0, \quad S_5^* = \frac{39}{16}; \\ k = 6, \quad X'_6 = \{(1, 0, 1)\}, \quad S_6^* = \frac{7}{4}; \\ k = 7, \quad X'_7 = \{(1, 0, 0)\}, \quad S_7^* = \frac{7}{3}; \\ k = 8, \quad \alpha_1^8 = 0, \quad \tilde{x}^8 = \tilde{x}^0, \quad S_8^* = \frac{55}{16}; \\ k = 9, \quad \alpha_1^9 = 0, \quad \alpha_2^9 = 1, \quad S_9^* = \frac{23}{16} < R; \text{ вершина 9 вычёркивается;} \\ k = 10, \quad \alpha_1^{10} = 0, \quad \alpha_2^{10} = 0, \quad \tilde{x}^{10} = \tilde{x}^0, \quad S_{10}^* = \frac{55}{16}; \end{aligned}$$

$k = 11, X'_{11} = \{(0, 0, 1)\}, x^* = \tilde{x}^{11} = (0, 0, 1), R = S^*_{11} = \frac{11}{4};$

$k = 12, X'_{12} = \{(0, 0, 0)\}, S^*_{12} = 0.$

Все вершины просмотрены. Оптимальное решение получено в 11-ой вершине: $S^* = \frac{11}{4}, x^* = (0, 0, 1), y^* = (0, 0, \frac{5}{8})$. Чтобы найти решение, рассмотрели 13 вершин.

Применение метода ветвей и границ к двойственной задаче.

При $k = 0$ имеем $\Phi_0^* = \frac{55}{16}, u^0 = (0, \frac{11}{16}), x^0 = (0, 0, x_3)$, где $x_3 \in \{0, 1\}$. Нетривиальный рекорд равен $x^* = x^0 = (0, 0, 1)$, при этом $R = \varphi(x^0) = \frac{11}{4}, u(x^0) = (0, \frac{3}{4})$. Таким образом, уже в нулевой вершине имеем нетривиальный рекорд.

Далее последовательно получаем:

$k = 1, \alpha_1^1 = 1, \Phi_1^* = \frac{39}{16} < R$; вершина 1 вычёркивается;

$k = 2, \alpha_1^2 = 0, \Phi_2^* = \Phi_0^* > R$;

$k = 3, \alpha_1^3 = 0, \alpha_2^3 = 1, \Phi_3^* = \frac{7}{16} < R$; вершина 3 вычёркивается;

$k = 4, \alpha_1^4 = 0, \alpha_2^4 = 0, \Phi_4^* = \Phi_0^* > R$.

В пятой вершине множество X_5 содержит единственную точку, совпадающую с рекордом, а в шестой вершине $X_6 = \{(0, 0, 0)\}$ и $\Phi_6^* = 0$. Чтобы найти решение, рассмотрено шесть вершин. Оптимальное решение $x^* = x^0 = (0, 0, 1)$, при этом $\Phi^* = \frac{11}{4}$. Это решение, естественно, совпадает с полученным ранее.

6. Решение задачи (1)

Аналогично можно решать общую задачу (1). Двойственная задача может быть представлена так:

$$\Phi(x, v, w) = \sum_{i \in I} \{a_i \max(0, \max_{j \in J} (c_{ij} - v_j - d_{ij}w_j) - c_i^0)\} x_i + \sum_{j \in J} (b_j v_j + D_j w_j) \rightarrow \max_{x \in X} \min_{v, w \geq 0} .$$

Здесь $v = (v_1, \dots, v_m), w = (w_1, \dots, w_m)$.

Все основные свойства упрощённой задачи, рассмотренные в разделе (3), имеют место и в этой задаче. Но для нахождения оценок требуется на каждом шаге решать m задач, аналогичных задаче нахождения оценки, описанной выше, так как теперь надо найти решение в каждой из областей $U_{j_0} = \{(v, w) \mid c_{ij_0} - v_{j_0} - d_{ij_0}w_{j_0} \geq c_{ij} - v_j - d_{ij}w_j (j \in J, j \neq j_0)\}$.

Список литературы

- [1] В. Л. Береснев, Э. Х. Гимади, В. Т. Дементьев *Экстремальные задачи стандартизации*. - Новосибирск: Наука, 1978, 333 с.
- [2] В. Ф. Демьянов, В. Н. Малозёмов *Введение в минимакс*. - М.: Наука, 1972, 368 с.
- [3] М. М. Ковалёв *Дискретная оптимизация* - Минск: Изд-во БГУ, 1977, 192 с.

РЕГУЛЯРИЗОВАННЫЙ ПРОЕКЦИОННЫЙ ДВУХШАГОВЫЙ МЕТОД ВТОРОГО ПОРЯДКА С ПЕРЕМЕННОЙ МЕТРИКОЙ ДЛЯ ЗАДАЧ МИНИМИЗАЦИИ С ОГРАНИЧЕНИЯМИ

В.Г. Малинов

*Ульяновский государственный университет, Ульяновск
e-mail: vgtalinov@mail.ru*

Аннотация. В статье предлагается и исследуется метод регуляризации с переменной метрикой для решения задач минимизации с функциональными ограничениями при неточных исходных данных в гильбертовом пространстве, основанный на двухшаговом проекционном обобщенном двухпараметрическом методе с переменной метрикой второго порядка в сочетании с методом штрафных функций. Обосновываются достаточные условия сходимости метода.

Ключевые слова: минимизация с ограничениями, регуляризация, двухшаговый метод переменной метрики второго порядка.

1. Постановка задачи

Рассмотрим нелинейную задачу минимизации

$$f(\mathbf{x}) \longrightarrow \inf, \mathbf{x} \in Q \subset H, \quad (1)$$

$$G = \{\mathbf{x} \in Q : g_i(\mathbf{x}) \leq 0, i \in \{1, \dots, m\}; g_i(\mathbf{x}) = 0, i \in \{m+1, \dots, t\}\}, \quad (2)$$

где Q – выпуклое замкнутое множество; гильбертово пространство H нормировано скалярным произведением, $\forall \mathbf{x} \in H \|\mathbf{x}\| = (\mathbf{x}, \mathbf{x})^{1/2}$; функции $f(\mathbf{x}), g_i(\mathbf{x}), i \in \{1, \dots, t\}$, определены, выпуклы и дважды дифференцируемы по Фреше на H . Предполагаем, что $f(\mathbf{x})$ имеет "овражные" поверхности уровней;

$$\inf f(\mathbf{x}) = f_* > -\infty, \mathbf{x} \in Q; \quad G_* = \{\mathbf{x} \in G : f(\mathbf{x}) = f_*\} \neq \emptyset; \quad (3)$$

градиенты функций $f(\mathbf{x})$ и $g_i(\mathbf{x})$ имеют возмущения. Первое предположение подсказывает предпочтительность двух- и многошаговых методов для решения задачи, а последнее говорит о неустойчивости задачи (1), (2) и необходимости для её решения методов регуляризации.

2. Метод решения задачи

Для решения этой задачи в последнее десятилетие на основе подхода из работы [1] разработаны и исследованы непрерывные методы в пространствах с переменной метрикой [2], [3]. Итеративный двухшаговый метод переменной метрики (МПМ) первого порядка на основе этого подхода предложен и исследован в работе [4], а итеративный МПМ второго порядка – в [5]. Здесь, применяя метод регуляризации Тихонова, поставим в соответствие задаче (1), (2) последовательность задач минимизации функционала Тихонова

$$T_k(\mathbf{x}) = f(\mathbf{x}) + \theta_k P(\mathbf{x}) + \tau_k \|\mathbf{x}\|^2/2,$$

где $P(\mathbf{x}) = \sum_{i=1}^t [g_i(\mathbf{x})]^2 \quad \forall \mathbf{x} \in H$ – выпуклая штрафная функция. В силу условий для функций $f(\mathbf{x})$ и $g_i(\mathbf{x})$, функции $P(\mathbf{x})$ и $T_k(\mathbf{x})$ дважды дифференцируемы по Фреше на множестве Q ; существует невырожденный $\forall \mathbf{x} \in H$ положительно определенный оператор второй производной $\mathbf{T}''_k(\mathbf{x})$, который для краткости будем обозначать $\mathbf{T}''(\mathbf{x})$.

Наряду с имеющимся, введем в H новое скалярное произведение $(\mathbf{T}''(\mathbf{x})\mathbf{u}, \mathbf{u})$, где $\mathbf{T}''(\mathbf{x}) : H \rightarrow H$ при каждом фиксированном $\mathbf{x} \in H$ служит положительно определенным самосопряженным линейным оператором, изменяющим метрику пространства. Проекция $\mathbf{w} = P_Q^{\mathbf{T}''(\mathbf{x})}(\mathbf{v})$ точки $\mathbf{v} \in H$ на множество Q в новой метрике существует и является единственным решением задачи $g(\mathbf{u}) = (\mathbf{T}''(\mathbf{x})(\mathbf{u} - \mathbf{v}), \mathbf{u} - \mathbf{v}) \rightarrow \inf, \quad \mathbf{u} \in Q$, в силу выпуклости множества Q и сильной выпуклости функции $g(\mathbf{u})$. Критерием проекции $\mathbf{w} \in Q$ точки $\mathbf{v} \in H$ в новой метрике служит неравенство

$$(\mathbf{T}''(\mathbf{x})(\mathbf{w} - \mathbf{v}), \mathbf{u} - \mathbf{w}) \geq 0, \quad \mathbf{u} \in Q. \quad (4)$$

В построенном пространстве для решения задачи (1), (2) рассмотрим метод регуляризации на основе двухшагового обобщенного двухпараметрического двухэтапного проекционного МПМ:

$$\begin{aligned} \mathbf{y}^k &= \mathbf{x}^k - \mathbf{x}^{k-1}, \quad \mathbf{z}^k = \mathbf{x}^k + \alpha_k \mathbf{y}^k; \\ \mathbf{x}^{k+1} &= P_Q^{\mathbf{T}''(\mathbf{z}^k)} [\mathbf{z}^k - \gamma_k \mathbf{T}''(\mathbf{z}^k)^{-1} \mathbf{t}'_k(\mathbf{z}^k)], \quad k = 0, 1, 2, \dots, \end{aligned} \quad (5)$$

где $\forall \mathbf{x}^0 \in H$ – начальная точка; $\mathbf{x}^{-1} = \mathbf{x}^0$; $\mathbf{T}''(\mathbf{z})^{-1}$ – оператор, обратный к $\mathbf{T}''(\mathbf{x})$ в точке \mathbf{z}^k ; в приближениях

$$\mathbf{t}'_k(\mathbf{z}^k) = \mathbf{f}'_k(\mathbf{z}^k) + \theta_k \mathbf{P}'_k(\mathbf{z}^k) + \tau_k \mathbf{z}^k, \quad k \geq 0 \quad (6)$$

в точке \mathbf{z}^k точного градиента $\mathbf{T}'_k(\mathbf{x}) = \mathbf{f}'(\mathbf{x}) + \theta_k P'(\mathbf{x}) + \tau_k \mathbf{x}$ функции Тихонова $T_k(\mathbf{x})$ приближенные градиенты $\mathbf{f}'_k(\mathbf{z}^k)$, $P'_k(\mathbf{z}^k)$ таковы, что

$$\max \{ \|\mathbf{f}'_k(\mathbf{x}) - \mathbf{f}'(\mathbf{x})\|; \|\mathbf{P}'_k(\mathbf{x}) - \mathbf{P}'(\mathbf{x})\| \} \leq \delta_k (1 + \|\mathbf{x}\|) \quad \forall \mathbf{x} \in Q, \quad k \geq 0; \quad (7)$$

$\alpha_k, \gamma_k, \theta_k, \tau_k, \delta_k$ – параметры метода (5), (6).

3. Исследование сходимости метода (5), (6)

Нормальным решением задачи (1), (2) называют его решение с минимальной нормой. Рассмотрим достаточные условия сильной сходимости итерационной последовательности $\{\mathbf{x}^k\}$ метода (5), (6) к нормальному решению задачи.

Теорема 1. Пусть выполнены следующие условия:

- 1) множество $Q \subset H$ выпукло и замкнуто;
- 2) функции $f(\mathbf{x})$, $g_i(\mathbf{x})$ выпуклы, дважды дифференцируемы по Фреше на H , поэтому градиенты функций $f(\mathbf{x})$, $P(\mathbf{x})$ липшицевы, $\exists L = \text{const} > 0$,

$$\max \{ \|\mathbf{f}'(\mathbf{x}) - \mathbf{f}'(\mathbf{u})\|; \|\mathbf{P}'(\mathbf{x}) - \mathbf{P}'(\mathbf{u})\| \} \leq L \|\mathbf{x} - \mathbf{u}\| \quad \forall \mathbf{u}, \mathbf{x} \in H; \quad (8)$$

оператор $\mathbf{T}''(\mathbf{x})$ таков, что

$$m \|\mathbf{u}\|^2 \leq (\mathbf{T}''(\mathbf{x})\mathbf{u}, \mathbf{u}) \leq M \|\mathbf{u}\|^2, \quad 0 < m \leq M, \quad \mathbf{u}, \mathbf{x} \in H; \quad (9)$$

- 3) для приближений $\mathbf{f}'_k(\mathbf{x}), \mathbf{P}'_k(\mathbf{x})$ точных градиентов $\mathbf{f}'(\mathbf{x}), \mathbf{P}'(\mathbf{x})$ имеет место (7);
 4) существуют выпуклые функции $\varphi_1(\mathbf{x}), \varphi_2(\mathbf{x}) \in C^{1,1}(Q)$ такие, что

$$\varphi'_1(\mathbf{x}) = \mathbf{T}''(\mathbf{x})^{-1}\mathbf{f}'(\mathbf{x}), \quad \varphi'_2(\mathbf{x}) = \mathbf{T}''(\mathbf{x})^{-1}\mathbf{P}'(\mathbf{x}) \quad \forall \mathbf{x} \in Q;$$

- 5) функция Лагранжа задачи (1), (2) $L(\mathbf{x}, \lambda) = f(\mathbf{x}) + \sum_{i=1}^t \lambda_i g_i(\mathbf{x}), \quad \mathbf{x} \in Q,$
 $\lambda \in \Lambda = \{\lambda \in E^t : \lambda_i \geq 0, i \in \{1 : t\}\}$ имеет седловую точку $(\mathbf{x}^*, \lambda^*) \in Q \times \Lambda,$
 $\forall \mathbf{x} \in Q, \lambda \in \Lambda, m. e.$

$$L(\mathbf{x}^*, \lambda) \leq L(\mathbf{x}^*, \lambda^*) \leq L(\mathbf{x}, \lambda^*);$$

- 6) параметры метода (5), (6) $\alpha_k, \gamma_k, \tau_k, \theta_k, \delta_k$ – таковы, что:

$$\alpha_k \geq \alpha_{k+1} \geq 0, \gamma_k \geq \gamma_{k+1} > 0, \delta_k \geq 0, \theta_{k+1} \geq \theta_k > 0, \tau_k \geq \tau_{k+1} > 0, \quad k \geq 0;$$

$$5m\alpha_k < m - 10\tau_k - 5\delta_k(1 + \theta_k), (8 - 15\alpha_k)M^2 \leq 60m(1 - \alpha_k)\gamma_k\tau_k; \quad (10)$$

$$10\tau_k + 5\delta_k(1 + \theta_k) < m, \quad k \geq 0;$$

$$\theta_{k+1} - \theta_k \rightarrow 0, \quad \tau_k - \tau_{k+1} \rightarrow 0, \quad k \rightarrow \infty;$$

$$(|\theta_k - \theta_{k+1}| + |\tau_k - \tau_{k+1}|)^2 [\gamma_{k-1}\tau_{k-1}]^{-1} \tau_k^{-2} \rightarrow 0, \quad k \rightarrow \infty,$$

$$\delta_k + \tau_k + \gamma_k^2 [\delta_k(1 + \theta_k) + \tau_k] (\gamma_{k-1}\tau_{k-1})^{-1} \rightarrow 0, \quad k \rightarrow \infty.$$

Тогда последовательность $\{\mathbf{x}^k\}$, определяемая методом (5), (6), (10), из любой начальной точки $\mathbf{x}^0 \in H$ равномерно относительно выбора приближенных градиентов $\mathbf{f}'_k(\mathbf{x}), \mathbf{P}'_k(\mathbf{x})$ в (7) сходится к точке \mathbf{x}^* ,

$$\|\mathbf{x}^k - \mathbf{x}^*\| \rightarrow 0, \quad k \rightarrow \infty, \quad (11)$$

где $\|\mathbf{x}^*\| = \inf \|\mathbf{x}\|, \mathbf{x} \in G_*, \mathbf{x}^* \in G_*$ – нормальное решение задачи (1), (2).

Доказательство. Прежде всего отметим, что в нём существенно используется четвертое условие теоремы и эквивалентность метрик в гильбертовом пространстве H . Далее заметим, что в силу предположений теоремы 1 условия (3) выполнены, множество минимумов G_* выпукло и замкнуто, нормальное решение задачи (1), (2) существует. Ввиду сильной выпуклости функции Тихонова на H существует единственная точка $\mathbf{v}^r \in Q$ такая, что (см. [1]–[4])

$$T_k(\mathbf{v}^r) = \inf T_k(\mathbf{x}), \quad \mathbf{x} \in Q, \quad r \geq 0,$$

$$\|\mathbf{v}^k - \mathbf{x}^*\| \rightarrow 0, \quad k \rightarrow \infty, \quad (12)$$

$$(\mathbf{T}'_k(\mathbf{v}^k), \mathbf{u} - \mathbf{v}^k) \geq 0, \quad \mathbf{u} \in Q \quad (13)$$

$$\|\mathbf{v}^{k+1} - \mathbf{v}^k\| \leq C\tau_k^{-1} (|\tau_k - \tau_{k+1}|; |\theta_k - \theta_{k+1}|), \quad k \geq 0, \quad (14)$$

$$C = \sup_{k \geq 0} \max \{ \|\mathbf{u}^k\|; \|\mathbf{f}'(\mathbf{v}^k)\|; \|\mathbf{P}'(\mathbf{v}^k)\| \}. \quad (14')$$

Обосновывая (11), оценим правую часть неравенства

$$\|\mathbf{x}^k - \mathbf{x}^*\| \leq \|\mathbf{x}^k - \mathbf{x}^{k+1}\| + \|\mathbf{x}^{k+1} - \mathbf{v}^k\| + \|\mathbf{v}^k - \mathbf{x}^*\|, \quad k \geq 0. \quad (15)$$

Учитывая (12), в (15) следует показать выполнение соотношений

$$\|\mathbf{x}^k - \mathbf{x}^{k+1}\| \rightarrow 0, \quad \|\mathbf{v}^k - \mathbf{x}^{k+1}\| \rightarrow 0, \quad k \rightarrow \infty. \quad (16)$$

Из характеристического свойства оператора проектирования ([6], с. 72) и из (5) получаем вариационное неравенство (далее индекс k у параметров метода для краткости почти везде опускаем)

$$\left(\mathbf{x}^{k+1} - \mathbf{z}^k + \gamma \mathbf{T}''(\mathbf{z}^k)^{-1} \mathbf{t}'_k(\mathbf{z}^k), \mathbf{u} - \mathbf{x}^{k+1} \right) \quad \forall \quad \mathbf{u} \in Q, \quad k \geq 0. \quad (17)$$

Заметим, что неравенство (13) в силу свойства (4) оператора $\mathbf{T}''(\mathbf{x})$ новой метрики в H эквивалентно равенству

$$\mathbf{v}^k = P_Q^{\mathbf{T}''(\mathbf{v}^k)} \left[\mathbf{v}^k - \gamma \mathbf{T}''(\mathbf{v}^k)^{-1} \mathbf{t}'^k(\mathbf{v}^k) \right], \quad \gamma > 0, \quad k \geq 0. \quad (18)$$

С другой стороны, согласно свойству оператора проектирования ([6], с.72) в первоначальной метрике пространства H , из (18) следует неравенство

$$\gamma \left(\mathbf{T}''(\mathbf{v}^k)^{-1} \mathbf{T}'_k(\mathbf{v}^k), \mathbf{u} - \mathbf{v}^k \right) \geq 0, \quad \mathbf{u} \in Q, \quad k \geq 0.$$

Положим здесь $\mathbf{u} = \mathbf{x}^{k+1}$, затем сложим с (17), положив в нём $\mathbf{u} = \mathbf{v}^k$:

$$\left(\mathbf{x}^{k+1} - \mathbf{z}^k + \gamma \left[\mathbf{T}''(\mathbf{z}^k)^{-1} \mathbf{t}'_k(\mathbf{z}^k) - \mathbf{T}''(\mathbf{v}^k)^{-1} \mathbf{T}'_k(\mathbf{v}^k) \right], \mathbf{v}^k - \mathbf{x}^{k+1} \right) \geq 0 \quad (19)$$

$\forall k \geq 0$. Подставив выражения для точного градиента и приближенного градиента (6) функции Тихонова, представим (19) в виде

$$\begin{aligned} & \left(\mathbf{x}^{k+1} - \mathbf{z}^k, \mathbf{v}^k - \mathbf{x}^{k+1} \right) \leq \gamma \left\{ \left(\mathbf{T}''(\mathbf{z}^k)^{-1} (\mathbf{f}'_k(\mathbf{z}^k) - \mathbf{f}'(\mathbf{z}^k)), \mathbf{v}^k - \mathbf{x}^{k+1} \right) + \right. \\ & \quad + \left(\mathbf{T}''(\mathbf{z}^k)^{-1} (\mathbf{f}'(\mathbf{z}^k) - \mathbf{T}''(\mathbf{v}^k)^{-1} (\mathbf{f}'(\mathbf{v}^k) + \tau \mathbf{T}''(\mathbf{z}^k)^{-1} (\mathbf{z}^k - \mathbf{v}^k))), \mathbf{v}^k - \mathbf{z}^k \right) + \\ & \quad + \theta \left(\mathbf{T}''(\mathbf{z}^k)^{-1} [\mathbf{P}'_k(\mathbf{z}^k) - \mathbf{P}'(\mathbf{z}^k)] + \mathbf{T}''(\mathbf{z}^k)^{-1} \mathbf{P}'(\mathbf{z}^k) - \mathbf{P}'(\mathbf{v}^k) \mathbf{v}^k, \mathbf{v}^k - \mathbf{x}^{k+1} \right) + \\ & \quad \left. + \tau \left(\mathbf{T}''(\mathbf{z}^k)^{-1} (\mathbf{z}^k - \mathbf{v}^k), \mathbf{z}^k - \mathbf{x}^{k+1} \right) + \tau \left(\left[\mathbf{T}''(\mathbf{z}^k)^{-1} - \mathbf{T}''(\mathbf{v}^k)^{-1} \right] \mathbf{v}^k, \mathbf{v}^k - \mathbf{x}^{k+1} \right) \right\}, k \geq 0. \end{aligned}$$

Преобразуем это неравенство: левую часть – с помощью свойств скалярного произведения, правую – пользуясь (7)–(10), вторым, третьим, четвертым условиями теоремы 1, а также (14') и неравенством [7]

$$(m/M^2) \|\mathbf{u}\|^2 \leq \left(\mathbf{T}''(\mathbf{x})^{-1} \mathbf{u}, \mathbf{u} \right) \leq \|\mathbf{u}\|^2/m, \quad \forall \quad \mathbf{u}, \mathbf{x} \in Q, \quad (20)$$

где M и m из (9); тогда получим

$$\begin{aligned} & \|\mathbf{x}^{k+1} - \mathbf{v}^k\| + \left(\mathbf{v}^k - \mathbf{z}^k, \mathbf{x}^{k+1} - \mathbf{v}^k \right) + \gamma \tau \left(\mathbf{T}''(\mathbf{z}^k)^{-1} (\mathbf{z}^k - \mathbf{v}^k), \mathbf{z}^k - \mathbf{v}^k \right) \leq \\ & \leq \gamma \delta (1 + \theta) (1 + \|\mathbf{z}^k\|) \|\mathbf{x}^{k+1} - \mathbf{v}^k\|/m + \gamma (\varphi'_1(\mathbf{z}^k) - \varphi'_1(\mathbf{v}^k), \mathbf{v}^k - \mathbf{x}^{k+1}) + \\ & \quad + \gamma \theta (\varphi'_2(\mathbf{z}^k) - \varphi'_2(\mathbf{v}^k), \mathbf{v}^k - \mathbf{x}^{k+1}) + \gamma \tau \|\mathbf{z}^k - \mathbf{v}^k\| \|\mathbf{z}^k - \mathbf{x}^{k+1}\|/m + \\ & \quad + 2\gamma \tau (C \|\mathbf{v}^k - \mathbf{x}^k\| + C \|\mathbf{x}^k - \mathbf{x}^{k+1}\|) /m, k \geq 0. \end{aligned} \quad (21)$$

Далее, оценив слагаемые в левой и правой части (21) с помощью (20), неравенств

$$2 |ab| \leq \varepsilon a^2 + \varepsilon^{-1} b^2, \quad (a+b)^2 \leq (1+\varepsilon)a^2 + (1+\varepsilon^{-1})b^2, \quad \forall \quad a, b, \varepsilon > 0,$$

$$(\mathbf{g}'(\mathbf{u}) - \mathbf{g}'(\mathbf{v}), \mathbf{v} - \mathbf{w}) \leq L\|\mathbf{v} - \mathbf{w}\|^2/4 \quad \forall \mathbf{u}, \mathbf{v}, \mathbf{w} \in Q$$

(см. [1]), придём к неравенству

$$\begin{aligned} & \|\mathbf{x}^{k+1} - \mathbf{v}^k\|^2 - a_1 \|\mathbf{x}^k - \mathbf{v}^k\|^2 + a_2 \|\mathbf{x}^k - \mathbf{x}^{k+1}\|^2 \leq \\ & \leq a_3 \|\mathbf{x}^k - \mathbf{v}^k\|^2 + a_4 \|\mathbf{y}^k\|^2 - \alpha \|\mathbf{x}^{k-1} - \mathbf{v}^k\|^2 + a_5, \quad k \geq 0, \end{aligned} \quad (22)$$

где $a_1 = 2/5 - 2m(1 - \alpha)\gamma\tau M^2$;

$$a_2 = 1 - \alpha - 0.5\gamma \{ [Lm(1 + \alpha) + 2\delta](1 + \theta) + 2\tau(1 + \alpha) \} / m - [2\tau + \delta(1 + \theta)] / m;$$

$$a_3 = 3/5 + \alpha + 2[\tau + \delta(1 + \theta)] / m + \gamma[(3 + \alpha)\delta(1 + \theta) + (1 + \alpha)\tau] / m;$$

$$a_4 = 2\alpha + 2m(\alpha - \alpha^2)\gamma\tau/M^2 + (\alpha + \alpha^2)\gamma[(Lm + 2\delta)(1 + \theta) + 4\tau] / (2m);$$

$$a_5 = C_1\gamma^2[\delta(1 + \theta) + \tau];$$

$$C_1 = \max \{ (C + 1)^2 / m; 4C^2 / m \}.$$

Неравенство (22) представим в виде

$$u_k \leq (1 - s_k)u_{k-1} + w_k + a_5, \quad k - 1 \geq N, \quad (23)$$

где $s_k = \gamma_{k-1}\tau_{k-1}$, левая часть u_k равна левой части (21), а w_k равно разности между правой частью (21) без a_5 и первым слагаемым в правой части (23).

Равномерная сходимость последовательности $\{u_k\}$ относительно выбора приближенных градиентов в (7) доказывается аналогично проведенному в работе [8]. Из равномерной сходимости последовательности $\{u_k\}$ следует равномерная сходимость в (16). Из (16), (12) и (15) следует равномерная сходимость в (11). Теорема 1 доказана.

Правило останова метода (5), (6) и регуляризирующий оператор строятся так же, как это сделано в работах [4]–[8].

Список литературы

- [1] Антипин А.С. *Методы нелинейного программирования, основанные на прямой и двойственной модификации функции Лагранжа*. – Препринт. М.: ВНИИ системных исследований, 1979. – 73 с.
- [2] Антипин А.С., Васильев Ф.П. *О непрерывном методе минимизации в пространствах с переменной метрикой*// Известия вузов. Математика. – 1995. – № 12(403). – С. 3–9.
- [3] Амочкина Т.В., Антипин А.С., Васильев Ф.П. *Регуляризованный непрерывный метод минимизации с переменной метрикой при неточно заданных исходных данных* // Вестник МГУ. – Сер. 15. – Вычисл. матем. и киберн. – 1997. – № 3. – С. 9–12.
- [4] Малинов В.Г. *Проекционный двухшаговый обобщенный двухпараметрический метод минимизации первого порядка с переменной метрикой* // Ученые записки Ульяновского гос. университета. Серия "Фундаментальные проблемы математики и механики". – Вып.1(13). – Ульяновск. Изд-во УлГУ, 2003. – С. 127–138.

- [5] Malinov V.G. *Projection two-step Variable Metric Methods* // The 4th Moscow International Conference on Operations Research (ORM2004). Moscow. September 21-24, 2004. Proceedings. – Moscow. MAKS Press, 2004. – P. 135–137.
- [6] Васильев Ф.П. *Методы решения экстремальных задач*. – М.: Наука, 1981. – 400 с.
- [7] Пшеничный Б.Н., Данилин Ю.М. *Численные методы в экстремальных задачах*. – М.: Наука, 1975. – 320 с.
- [8] Малинов В.Г. *О регуляризованном двухшаговом проекционном методе для задач минимизации с ограничениями* // ЖВМ и МФ. – 2000. – Т. 40. – № 1. – С. 65–71.

REGULARIZED PROJECTION TWO-STEP SECOND ORDER VARIABLE METRIC CONSTRAINED MINIMIZATION METHOD

V.G. Malinov

Ulyanovsk State University, Ulyanovsk
e-mail: vgmalinov@mail.ru

Abstract. A regularization method in Hilbert space is proposed for problems of constrained minimization with inaccurate initial data, based on the iterative projection two-step second order variable metric method in conjunction with the Tikhonov function method. Sufficient conditions for the convergence proved.

Key words: constrained minimization, regularization, two-step second order variable metric method.

О НАИМЕНЬШИХ ЗНАЧЕНИЯХ ПОЛИНОМОВ

М.А.Новиков

Институт динамики систем и теории управления СО РАН

e-mail: nta@icc.ru

Аннотация. В статье обсуждается вопрос о наименьших значениях полиномов, принимающих значения во всей допустимой области определения. Для изложения материала выбраны два независимых аргумента. В качестве метода избрано построение формального ряда с понижающимися показателями степеней.

Ключевые слова: экстремум, наименьшее значение, полином, решение системы уравнений, параметрическое решение.

Введение

Исследование функций кроме локальных экстремумов предполагает выявление наибольших и наименьших значений не только в замкнутой области, но и во всей области определения. Это, в частности, имеет важное значение в задаче полиномиальной оптимизации [1].

С ростом числа переменных анализ функций усложняется, поэтому ограничимся только двумя переменными, а в качестве исследуемой функции выберем полином. Для определенности будем проводить исследование наименьших значений функции $f(x, y)$.

Ставится задача определения наименьших значений полиномиальных функций

$$f(x, y) = F_m(x, y) + F_{m-1}(x, y) + \dots + F_1(x, y), \quad (1)$$

где $F_k(x, y)$ – однородные формы порядка k ($1 \leq k \leq m$); $m < \infty$; $x, y \in R$. Целью исследования является нахождение условий и вычислительного метода для нахождения наименьших значений полиномов в бесконечно удаленных точках.

Схожие постановки приводятся, например, в задачах энергетики [2], где преимущественно исследование сводилось к определению области допустимых значений функций.

1. Об условиях существования граничных значений в бесконечно удаленных точках

При решении задачи о нахождении наименьших значений функции во всей вещественной плоскости во всяком случае нужно определить все локальные экстремумы. Необходимыми условиями локального экстремума [3] являются:

$$f'_x(x, y) = 0, \quad f'_y(x, y) = 0. \quad (1.1)$$

На решениях уравнений (1.1) можно составить неоднородную форму

$$\Phi(x, y) = x f'_x(x, y) + y f'_y(x, y) = 0,$$

которое по теореме Эйлера об однородных функциях примет вид:

$$\Phi(x, y) = m F_m(x, y) + (m - 1) F_{m-1}(x, y) + \dots + F_1(x, y) = 0. \quad (1.2)$$

При отсутствии ограничений на ООФ нужен анализ функции в бесконечно удаленных точках $\{x_0, y_0\}$, когда хотя бы одна из переменных обращается в бесконечность. Множество таких точек с их окрестностями описывается неравенством

$$x_0^2 + y_0^2 > A^2, \quad (A \rightarrow \infty). \quad (1.3)$$

Различаются три вида таких бесконечно удаленных точек:

- 1) $x_0 = a$ ($|a| < \infty$); $|y_0| = \infty$;
- 2) $|x_0| = \infty$; $y_0 = b$ ($|b| < \infty$);
- 3) $|x_0| = \infty$; $|y_0| = \infty$.

Не уменьшая общности считаем в первом случае $a = 0$, что всегда можно достичь подстановкой $x = z + a$ в функцию (1) с последующей заменой z на x без учета свободного члена. Во втором случае так же можно полагать $b = 0$.

Будем исходить из предположения, что в бесконечно удаленной точке $\{x_0, y_0\}$ функция (1) принимает предельное значение $\lim_{x \rightarrow x_0, y \rightarrow y_0} f(x, y) = B$, которым является точная нижняя грань (т.н.г.) функции $f(x, y)$. Конечно, должно быть $B < 0$, иначе наименьшим значением окажется $f(0, 0) = 0$.

Установим некоторые свойства таких функций в бесконечно удаленных точках. Можно показать, что в бесконечно удаленных точках перечисленных трех видов выполняются необходимые условия:

$$\lim_{x \rightarrow x_0, y \rightarrow y_0} f'_x(x, y) = 0; \quad \lim_{x \rightarrow x_0, y \rightarrow y_0} f'_y(x, y) = 0. \quad (1.1')$$

Вместе с уравнениями (1.1') справедливо и уравнение

$$\lim_{x \rightarrow x_0, y \rightarrow y_0} \Phi(x, y) = 0. \quad (1.2')$$

Два уравнения (1.1') и одно (1.2') составляют три зависимые уравнения, из которых независимыми можно выбрать любые два.

Форма старшего порядка оказывает существенное влияние на существование точных нижних граней полиномиальных функций в бесконечно удаленных точках. Очевидно, при знакопеременной старшей форме $F_m(x, y)$ функция (1) не имеет своей нижней грани (в том числе и верхней). В этом случае областью значений функции $f(x, y)$ является $(-\infty, +\infty)$.

Определенные заключения можно сделать и при знакоопределенной форме $F_m(x, y)$, когда не существует точной нижней грани (соответственно и верхней) функции $f(x, y)$ в бесконечно удаленных точках. В этом случае наименьшее значение полинома возможно только в точках локального минимума.

Очевидно, точки локального экстремума, являющиеся решениями системы (1.1), включаются в решения системы (1.1'), доставляющие наименьшие значения полиному (1). Поэтому все решения локальных экстремумов и наименьших значений полиномов будем искать из системы (1.1).

2. Построение решения задачи

Полиномиальные системы уравнений обычно решают построением параметрических кривых [4, 5, 6, 7] в виде степенных рядов с центром в начале координат и с положительными степенями по параметру. В нашей задаче центром является бесконечно удаленная точка. По типу бесконечно удаленных точек возможны 3 вида таких точек центра:

- 1) $(\infty; a)$ или $(-\infty; a)$, где a — конечная величина,
- 2) $(b; \infty)$ или $(b; -\infty)$, где b — конечная величина,
- 3) один из возможных вариантов: $(\infty; \infty)$, $(\infty; -\infty)$, $(-\infty; \infty)$, $(-\infty; -\infty)$.

Их можно привести к центру в начале координат соответствующим степенным преобразованием, которое для каждого вида указанного центра будет различным. При этом в новых переменных многоугольники Ньютона [7] перестроятся.

Для уменьшения анализа центров разных видов будем искать решения системы (1.1), используя многоугольники Ньютона в плоскости OXY . Соответствующие параметрические разложения будут строиться в порядке понижения степени параметра.

Сразу отметим, что все точки многоугольника Ньютона для функции $f'_x(x, y)$ сдвигом на 1 вправо совмещаются с точками многоугольника функции $f(x, y)$, точно так же все точки многоугольника для функции $f'_y(x, y)$ так же сдвигом на 1 вверх совмещаются с точками многоугольника функции $f(x, y)$.

Общее решение системы (1.1) предполагает качественные и количественные условия на вершины многоугольников $f'_x(x, y)$ и $f'_y(x, y)$.

Первые состоят в параллельности верхних (правых или левых) частей опорных линий многоугольников Ньютона $f'_x(x, y)$ и $f'_y(x, y)$ (иначе система (1.1) не совместна и не может иметь решений для центра в бесконечно удаленной точке).

Количественные условия определяются в начале процесса построения решений системы (1.1). Пусть решение системы (1.1) представимо в параметрическом виде [4, 5, 6, 7]:

$$x = \sum_{j=L}^{-\infty} a_j t^j, \quad y = \delta t^M, \quad (2.1)$$

где a_j — вещественные коэффициенты, определяемые последовательно, начиная с больших степеней, в процессе построения ветвей решений; δ может принимать значение (-1) для четных M и отрицательных y при $f(-x, y) = f(x, y)$, в остальных случаях $\delta = 1$; значение L определяется опорной линией многоугольников Ньютона функций $f'_x(x, y)$ и $f'_y(x, y)$, и вычисляется на начальном этапе построения параметрического решения. Параметру t будем придавать достаточно большие значения, при этом функция $x(t)$ может допускать как конечные, так и бесконечные значения в зависимости от величины L .

Система уравнений (1.1) определяет вначале величину a_L и по возможности δ , затем последовательно a_{L-1}, a_{L-2}, \dots . При этом можно установить, что решения задач (1.1) и (1.1') в параметрическом виде (2.1) существуют только при условиях:

- 1) верхние части опорных линий многоугольников Ньютона $f'_x(x, y)$ и $f'_y(x, y)$ параллельны;
- 2) продолжение верхней опорной линии $f(x, y)$ проходит через начало координат, либо верхняя опорная линия $f(x, y)$ проходит через вершины показателей степеней обобщенно-однородной функции, мономы которой удовлетворяют системе

$$\begin{cases} f'_x(a_L t^L, \delta t^M) = 0, \\ f'_y(a_L t^L, \delta t^M) = 0. \end{cases}$$

Построение решения (2.1) заключается в последовательном нахождении коэффициентов a_j системы:

$$\begin{cases} f_1(a, M, L, Q_1^{(j)}, t) = 0, \\ f_2(a, M, L, Q_2^{(j)}, t) = 0 \end{cases} \quad (2.2)$$

степеней $Q_1^{(j)}$ функции $f'_x(x(t), y(t))$ и $Q_2^{(j)}$ функции $f'_y(x(t), y(t))$.

В системе (2.2) могут получаться решения в виде бесконечных рядов, при этом $f(x(t), y(t))$ принимает конечные предельные значения, являющееся наименьшим значением $f(x, y)$.

Вместе с тем, система (2.2) может привести к отсутствию определенного вещественного решения a_j . Если при этом показатели степеней $Q_1^{(j)} + L \leq 0$, $Q_2^{(j)} + M \leq 0$, то выполняется система:

$$\begin{cases} f'_1(a, M, L, Q_1^{(j)}, t) \rightarrow 0, \\ f'_2(a, M, L, Q_2^{(j)}, t) \rightarrow 0 \end{cases} \quad (2.2')$$

Можно легко установить $Q_1^{(j)} + L = Q_2^{(j)} + M$, поэтому достаточно проверки только одного условия $Q_1^{(j)} + L \leq 0$. Точно так же (прежде всего для бесконечно удаленных точек второго вида) возможна параметризация:

$$y = \sum_{j=L}^{-\infty} a_j t^j, \quad x = \delta t^{\tilde{M}}, \quad (2.1')$$

где вещественные a_j так же подлежат определению.

Приведем пример, где описывается общая схема для нахождения т.н.г. полинома:

$$f(x, y) = \frac{1}{2} x^2 (y^4 + 2y^2 + 3) + x(-y^7 - y^6 - 3y^4 + y^3 - 4y^2 + 6y - \frac{9}{2}) + \frac{1}{2} y^{10} + y^9 - \frac{1}{2} y^8 + y^7 + \frac{3}{2} y^6 - 2y^5 + y^4 - \frac{7}{2} y^3 + \frac{21}{2} y^2 - 9y.$$

Система (1.1) в этом примере примет вид:

$$\begin{cases} f'_x = x(y^4 + 2y^2 + 3) - y^7 - y^6 - 3y^4 + y^3 - 4y^2 + 6y - \frac{9}{2} = 0, \\ f'_y = 2x^2(y^3 + y) + x(-7y^6 - 6y^5 - 12y^3 + 3y^2 - 8y + 6) + \\ + 5y^9 + 9y^8 - 4y^7 + 7y^6 + 9y^5 - 10y^4 - \frac{21}{2} y^2 + 21y - 9 = 0. \end{cases} \quad (2.3)$$

Легко посчитать, локальными точками экстремума здесь будут:

1. $x = \frac{7}{6} - \frac{11}{36}\sqrt{6}$; $y = \frac{\sqrt{6}}{6}$,
2. $x = \frac{7}{6} + \frac{11}{36}\sqrt{6}$; $y = \frac{-\sqrt{6}}{6}$,
3. $x = 3/2$; $y = 0$.

Как показывает анализ, первая и вторая точки являются стационарными, не доставляющими минимума, а в третьей точке достигается минимум, равный $(-27/8)$.

Теперь вычислим наименьшее значение. В принятых обозначениях по описанной схеме полагаем $M = 1$, $L = 3$. Сразу вычисляем $Q_1 = 7$; $Q_2 = 9$. Внутренние точки многоугольника Ньютона расположены ниже опорной линии. Поэтому параметризация (2.1) для нашей задачи будет следующей:

$$x = a_3 t^3 + \dots, \quad y = \delta t.$$

При подстановке последней параметризации в (2.3) составим систему вида (2.2):

$$\begin{cases} a_3 - \delta = 0, \\ 2a_3^2 \delta - 7a_3 + 5\delta = 0. \end{cases}$$

Ее решением является единственно $a_3 = \delta$. Следующий коэффициент a_2 определится из системы

$$\begin{cases} a_2 - 1 = 0, \\ -3 a_2 + 9 - 6 \delta = 0. \end{cases}$$

Отсюда получим: $\delta = 1$; $a_2 = 1$. Коэффициент $a_1 = -2$ находится из системы

$$\begin{cases} a_1 + 2 = 0, \\ -3 a_1 - 6 = 0. \end{cases}$$

Так же коэффициент $a_0 = 1$ удовлетворяет системе

$$\begin{cases} a_0 - 1 = 0, \\ -3 a_0 + 3 = 0. \end{cases}$$

Последующие коэффициенты с отрицательными степенями параметра a_{-j} ($j = 1, 2, \dots$) обозначим b_j . Дальнейшие вычисления приводят к значениям: $b_1 = 0$, $b_2 = -1$; $b_4 = 7/2$; $b_{2i+1} = 0$ ($i = 1, 2, \dots$), $b_6 = -4$; $b_8 = -5/2$; $b_{10} = 17$. При вычислении коэффициента b_{12} получается система:

$$\begin{cases} b_{12} + 19 = 0, \\ -3 b_{12} - \frac{159}{2} = 0 \end{cases}$$

соответственно (-8) и (-6) порядков параметра t . В этом случае последняя система не совместна, и построение ряда в прекращается. Применительно к введенным обозначениям получим: $Q_1^{(12)} + L = -8 + 3 = -5$, $Q_2^{(12)} = -6$.

Следовательно, при $|t| \rightarrow \infty$ существует предельное значение:

$$\lim_{t \rightarrow \infty} f(t^3 + t^2 - 2t + 1 - \frac{1}{t^2} + \frac{7}{2t^4} + \dots, t) = -3.5.$$

Простым перебором можно установить, что точной нижней гранью исследуемой функции получается $f_{min}(x, y) = -3.5$, являющееся наименьшим числом среди локальных минимумов и вычисленного наименьшего значения.

Список литературы

- [1] Утешев А.Ю., Черкасов Т.М. *К задаче полиномиальной оптимизации.*- М.: ДАН РАН, 1998, Т. 361, N. 2, с. 168-170
- [2] Зоркальцев В.И., Попов Л.Д. *Современные методы оптимизации и их приложения к моделям энергетики.* - Новосибирск: Наука, 2003, 248 с.
- [3] Фихтенгольц Г.М. *Курс дифференциального и интегрального исчисления.* Т. 1. - М.:Наука, 1966, 608 с.
- [4] Уокер Р. *Алгебраические кривые.* -М.: ИЛ. 1962, 236 с.
- [5] Брюно А.Д. *Локальный метод нелинейного анализа дифференциальных уравнений.* - М.: Наука, 1979, 255 с.

- [6] Солеев А. *Выделение ветвей аналитической кривой и многогранники Ньютона*. - М.: ДАН СССР, 1983, Т. 268, N. 6, с. 1305-1307
- [7] Брюно А.Д. *Степенная геометрия в алгебраических и дифференциальных уравнениях*. - М.: Наука, 1998, 288 с.

ON THE LEAST VALUES OF POLINOMIAL

M.A. Novickov

Institute of Systems Dynamics and Control Theory, Siberian Branch of the Russian Academy of Sciences, 134 Lermontov st., Irkutsk, 664033, Russia, E-mail: nma@icc.ru

Abstract The paper discusses the problem of the least values of polinomial, with acquire the values within the total admissible domain of definition. Two independent frgments have been shossen for the purposes of representing the material. In the capacity of the method costructing of a formal series with reducing the indicators of power has been shosen.

Key words polinom, extremum, least value, solving of the equations, parametric solving.

ИТЕРАЦИОННЫЙ МЕТОД УДЗАВЫ ДЛЯ ПОЛУКОЭРЦИТИВНОЙ ЗАДАЧИ СИНЬОРИНИ¹

Р.В. Намм, С.А. Сачков, Г. Ву

ХГТУ, Хабаровск

e-mail: sachkoff@mail.khstu.ru

Аннотация. Для полуконформной задачи Синьорини рассматривается итерационный метод Удзавы с модифицированным функционалом Лагранжа. Показано, что данный метод сходится к седловой точке при соответствующей регулярности решений вспомогательных задач.

Ключевые слова: вариационные неравенства, коэрцитивность, итеративный метод Удзавы.

Введение

Построение и исследование методов решения эллиптических вариационных неравенств в механике основывается обычно на предположении строгой коэрцитивности функционалов, минимизируемых на исходном гильбертовом пространстве. Однако, для ряда практически важных вариационных неравенств строгая коэрцитивность достигается только на подпространствах конечной размерности исходного пространства.

Приближенные методы решения такого рода полуконформных вариационных неравенств требуют специального исследования для обеспечения устойчивости вычислительного процесса [1, 2, 3].

Применение в полуконформных задачах классического функционала Лагранжа не гарантирует сходимость известных методов поиска седловых точек. В данной работе применительно к полуконформной задаче Синьорини исследуется итеративный метод Удзавы, основанный на модификации классического функционала Лагранжа. Будет показано, что данный метод сходится к седловой точке при соответствующей регулярности решений вспомогательных задач.

1. Метод Удзавы с модифицированным функционалом Лагранжа

Рассмотрим задачу минимизации [4, 5]:

$$\begin{cases} J(v) = \frac{1}{2} \int_{\Omega} |\nabla v|^2 d\Omega - \int_{\Omega} f v d\Omega & -min, \\ v \in G = \{w \in W_2^1(\Omega); \gamma w \geq 0 & \text{п.в. на } \Gamma\}, \end{cases} \quad (1)$$

где $\Omega \subset R^n (n = 2, 3)$ – ограниченная область с достаточно гладкой границей Γ , $f \in L_2(\Omega)$ – заданная функция, и $\gamma v \in W_2^{\frac{1}{2}}(\Gamma)$ – след функции $v \in W_2^1(\Omega)$ на Γ .

Так как функционал $J(v)$ не является строго коэрцитивным на G , задача (1) может не иметь решения. Однако, если

$$\int_{\Omega} f d\Omega < 0 \quad (2)$$

¹Работа поддержана грантом РФФИ (проект 04-01-97003)

то $J(v) \rightarrow +\infty$ при $\|v\|_{W_2^1(\Omega)} \rightarrow \infty$ ($v \in G$), и поэтому задача разрешима [4, 5]. Более того условие (2), обеспечивает единственность решения. В дальнейшем считаем данное условие выполненным.

Рассмотрим классический функционал Лагранжа:

$$L(v, \ell) = J(v) - \int \ell \gamma v d\Gamma = \frac{1}{2} \int_{\Omega} |\nabla v|^2 d\Omega - \int_{\Omega} f v d\Omega - \int_{\Gamma} \ell \gamma v d\Gamma$$

$$\forall (v, \ell) \in W_2^1(\Omega) \times L_2(\Gamma).$$

Пусть $(L_2(\Gamma))^+$ – конус неотрицательных функций, интегрируемых с квадратом на Γ .

Определение 1. Точка $(v^*, \ell^*) \in W_2^1(\Omega) \times (L_2(\Gamma))^+$ называется седловой точкой функционала $L(v, \ell)$, если выполнено неравенство:

$$L(v^*, \ell) \leq L(v^*, \ell^*) \leq L(v, \ell^*) \quad \forall (v, \ell) \in W_2^1(\Omega) \times (L_2(\Gamma))^+.$$

В монографии [6] показано, что если решение u^* задачи Синьорини в коэрцитивном случае принадлежит пространству $W_2^2(\Omega)$, то функционал Лагранжа имеет единственную седловую точку $(u^*, \frac{\partial u^*}{\partial n})$, т.е. $v^* = u^*$ п.в. в Ω и $\ell^* = \frac{\partial u^*}{\partial n}$ п.в. на Γ , где n – единичный вектор внешней нормали к Γ .

Аналогичный результат справедлив и для полукоэрцитивной задачи (1). Можно последовательно показать, что

$$1) L(u^*, \ell) \leq L(u^*, \frac{\partial u^*}{\partial n}) \quad \forall \ell \in (L_2(\Gamma))^+$$

$$2) L(u^*, \frac{\partial u^*}{\partial n}) \leq L(v, \frac{\partial u^*}{\partial n}) \quad \forall v \in W_2^1(\Omega).$$

Неравенство верно, т.к. $-\Delta u^* = f$ в Ω .

3) Единственность седловой точки.

Действительно, пусть наоборот $(u_1, \frac{\partial u_1}{\partial n}), (u_2, \frac{\partial u_2}{\partial n})$ – две седловые точки $L(v, \ell)$. Легко показать, что $u_1 - u_2 = const$ и, таким образом, $\frac{\partial u_1}{\partial n} = \frac{\partial u_2}{\partial n}$. Учитывая, что $L(u_1, \frac{\partial u_1}{\partial n}) = L(u_2, \frac{\partial u_1}{\partial n})$ и условие (2), получим требуемый результат.

Тем не менее, применение метода Удзавы для поиска седловой точки с использованием функционала $L(v, \ell)$ в полукоэрцитивном случае невозможно, т.к. сходимость итерационного процесса обеспечивается согласованием длины шага сдвига по двойственной переменной ℓ с константой положительной определенности квадратичной формы минимизируемого функционала. Но в полукоэрцитивной задаче (1) квадратичная форма $a(v, v) = \int_{\Omega} |\nabla v|^2 d\Omega$ лишь неотрицательно определена.

Для преодоления этого затруднения рассмотрим некоторую модификацию функционала Лагранжа.

Определим модифицированный функционал Лагранжа $M(v, \ell)$ на пространстве $W_2^1(\Omega) \times L_2(\Gamma)$ следующим образом:

$$M(v, \ell) = J(v) + \frac{1}{2r} \int_{\Gamma} \left(((\ell - r\gamma v)^+)^2 - \ell^2 \right) d\Gamma,$$

где $r > 0$ – const, символ w^+ означает $\max\{0, w\}$, т.е. $(\ell - r\gamma v)^+ = \max\{0, \ell - r\gamma v\}$.

Можно показать, что функционал $M(v, \ell)$ является выпуклым по v при фиксированном ℓ и вогнутым по ℓ при фиксированном v , и, более того, модифицированный функционал Лагранжа дифференцируем по Гато по двум переменным v, ℓ и при этом для производных $\nabla_v M(v, \ell), \nabla_{\ell} M(v, \ell)$ имеют место равенства:

$$(\nabla_v M(v, \ell), h) = a(v, h) - (f, h) - \int_{\Gamma} ((\ell - r\gamma v) + h) d\Gamma \quad \forall h \in W_2^1(\Omega),$$

$$(\nabla_\ell M(v, \ell), \theta) = \frac{1}{r} \int_\Gamma ((\ell - r\gamma v)^+ - \ell) \theta d\Gamma \quad \forall \theta \in L_2(\Gamma),$$

где $a(v, h) = \int \nabla v \cdot \nabla h d\Omega$, $(f, h) = \int_\Omega f h d\Omega$.

Рассмотрим для задачи (1) метод Удзавы с модифицированным функционалом Лагранжа $M(v, \ell)$.

Зададим произвольно начальную точку $\ell^0 \in W_2^{\frac{1}{2}}(\Gamma)$, которая является следом на Γ некоторой функции $u^0 \in W_2^1(\Omega)$. Алгоритм выглядит следующим образом:

1) на k -ом шаге ($k = 0, 1, \dots$), определяем

$$u^{k+1} = \arg \min_{v \in W_2^1(\Omega)} M(v, \ell^k) \quad (3)$$

2) находим ℓ^{k+1} по формуле:

$$\ell^{k+1} = (\ell^k - r\gamma u^{k+1})^+ \quad (4)$$

Здесь r означает параметр сдвига по двойственной переменной

Ниже будет показано, что точки $u^k, k = 1, 2, \dots$, которые определены по формуле (3), существуют, и, более того, при предположении, что $u^k \in W_2^1(\Omega)$, единственны.

Для исследования метода (3), (4) рассмотрим некоторую абстрактную схему, подобную схеме предложенной в [7, 8] для конечномерных задач оптимизации.

На пространстве $W_2^1(\Omega) \times L_2(\Gamma) \times L_2(\Gamma)$ определим функционал:

$$K(v, \ell, m) = \begin{cases} J(v) + \int_\Gamma \ell m d\Gamma + \frac{r}{2} \int_\Gamma m^2 d\Gamma, & \text{если } -\gamma v \leq m \\ +\infty & \text{в противном случае.} \end{cases}$$

Имеем:

$$\begin{aligned} \inf_m K(v, \ell, m) &= \inf_{-\gamma v \leq m} \left\{ J(v) + \int_\Gamma \left(\ell m + \frac{r}{2} m^2 \right) d\Gamma \right\} = \\ &= J(v) + \frac{1}{2r} \inf_{-\gamma v \leq m} \int_\Gamma ((\ell + rm)^2 - \ell^2) d\Gamma = \\ &= J(v) + \frac{1}{2} \int_\Gamma \left(((\ell - r\gamma v)^+)^2 - \ell^2 \right) d\Gamma = \\ &= M(v, \ell) \end{aligned} \quad (5)$$

Таким образом $M(v, \ell) = \inf_m K(v, \ell, m)$.

Введем функционал

$$\underline{M}(\ell) = \inf_v M(v, \ell) = \inf_v \inf_m K(v, \ell, m)$$

Известно, что

$$\inf_v \inf_m K(v, \ell, m) = \inf_m \inf_v K(v, \ell, m).$$

Поэтому

$$\inf_m \inf_v K(v, \ell, m) = \inf_m \inf_v \left\{ J(v) + \int_\Gamma \ell m d\Gamma + \frac{r}{2} \int_\Gamma m^2 d\Gamma \right\} =$$

$$\begin{aligned}
&= \inf_m \left\{ \inf_{-\gamma v \leq m} J(v) + \int_{\Gamma} \ell m d\Gamma + \frac{r}{2} \int_{\Gamma} m^2 d\Gamma \right\} = \\
&= \inf_m \left\{ \chi(m) + \int_{\Gamma} \ell m d\Gamma + \frac{r}{2} \int_{\Gamma} m^2 d\Gamma \right\},
\end{aligned}$$

где $\chi(m) = \inf_{-\gamma v \leq m} J(v)$.

Легко показать, что $\chi(m)$ – выпуклая функция [7, 8] и при условии (2) функция $\chi(m) \neq -\infty$ для $\forall m \in L_2(\Gamma)$ [5].

Таким образом функционал $\underline{M}(\ell)$ может быть представлен двумя способами.

$$\underline{M}(\ell) = \inf_v \left\{ J(v) + \frac{1}{2r} \int_{\Gamma} \left(((\ell - r\gamma v)^+)^2 - \ell^2 \right) d\Gamma \right\} \quad (6)$$

$$\underline{M}(\ell) = \inf_m \left\{ \chi(m) + \int_{\Gamma} \ell m d\Gamma + \frac{r}{2} \int_{\Gamma} m^2 d\Gamma \right\} \quad (7)$$

Теорема 1. Задача (6) разрешима при произвольном фиксированном $\ell \in (L_2(\Gamma))^+$.

Теорема 2. Пусть точки $u^k = \arg \min_{v \in W_2^1(\Omega)} M(v, \ell^{k-1})$, $k = 1, 2, \dots$ удовлетворяют условиям

(A) $u^k \in W_2^2(\Omega)$

(B) $\|u^k\|_{W_2^2(\Omega)} \leq C$, $C > 0$ – const. Тогда последовательность $\{(u_k, \ell^{k-1})\}$, полученная по методу (3), (4) сходится в $W_2^1(\Omega) \times L_2(\Gamma)$ к седловой точке $(u^*, \frac{\partial u^*}{\partial n})$ при любом выборе начальной точки $\ell^0 \in W_2^{\frac{1}{2}}(\Gamma)$ и любом фиксированном параметре $r > 0$.

Список литературы

- [1] Намм Р.В., Сачков С.А. Об устойчивом методе решения задачи Мосолова и Мясникова с трением на границе, основанном на схеме двойственности. // Сиб. журн. вычисл. математики. РАН, Сиб. отделение. – Новосибирск, 2002. – Т. 5, № 4. – с. 351-365.
- [2] Золотухин А.Я., Намм Р.В., Пачина А.В. Приближенное решение полукоэрцитивной задачи Синьборини с неоднородным граничным условием. // Журн. вычисл. математики и мат. физики. РАН. – Москва, 2003. – Т.43, № 3. – С. 388-398.
- [3] Вихтенко Э.М., Намм Р.В. О методе решения полукоэрцитивных вариационных неравенств, основанном на методе итеративной проксимальной регуляризации. // Известия высших учебных заведений. Серия: Математика. – Казань: Изд-во Казанск. гос. ун. 2004. – № 1. – С. 31-35.
- [4] Главачек И., Гаслингер Я., Нечас И., Ловишек Я. Решение вариационных неравенств в механике. М.: Мир, 1986. – 270 с.
- [5] Дюво Г., Лионс Ж.Л. Неравенства в физике и механике. – М.: Наука, 1980. – 480 с.
- [6] Glowinski R. Numerical Methods for Nonlinear Variational Problems. New York: Springer, 1984. – 381 p.

- [7] Гроссман К., Каплан А.А. Нелинейное программирование на основе безусловной минимизации. – Новосибирск: Наука, 1981. – 278 с.
- [8] Гольштейн Е.Г., Третьяков Н.В. Модифицированные функции Лагранжа. М.: Наука, 1989. – 400 с.

ITERATIONAL UDZAWA'S METHOD FOR SEMI-COERSIVE SIGNORINI'S PROBLEM

R.V. Namm, S.A. Sachkoff, G. Woo

680035 Russia, Khabarovsk, Tikhookeanskaya st., 136, KhSTU

e-mail: sachkoff@mail.khstu.ru

Abstract. Iterational Udzawa's method is observed for semi-coersive Signorini's problem. It is shown that the method converges to the saddle point in condition of regularity of auxiliary problems solutions.

О ПРИМЕНЕНИИ МЕТОДОВ КВАДРАТИЧНОГО ПРОГРАММИРОВАНИЯ К НАХОЖДЕНИЮ ДОПУСТИМОЙ ТОЧКИ СИСТЕМ КВАДРАТИЧНЫХ НЕРАВЕНСТВ ¹

Е.В. Таирова

*Иркутский государственный университет путей сообщения
e-mail: tairova_l@mail.ru*

Аннотация. Рассматривается задача нахождения допустимой точки системы невыпуклых квадратичных неравенств. Описывается редукция исходной задачи к задаче вогнутого квадратичного программирования. Приводятся результаты численных экспериментов.

Ключевые слова: невыпуклое квадратичное программирование, допустимая точка, вогнутое программирование, редукция.

Введение

Квадратичная оптимизация представляет собой наиболее важный раздел нелинейного программирования. Это связано прежде всего с многочисленными приложениями задач квадратичного программирования в различных областях. Например, задачи планирования производства и теории расписаний [1], моделирование сложных технических систем, дизайн, управление, задачи размещения производства, задачи экономико-математического моделирования, связанные с экономией от масштаба [2], задачи нахождения равновесных решений в рыночных моделях функционирования электроэнергетических систем (ЭЭС)[3,4]. Некоторые интересные комбинаторные задачи также могут быть помещены в рамки квадратичного программирования [5].

Если говорить о задачах невыпуклого квадратичного программирования с точки зрения математической сложности, то данные задачи являются NP-трудными [6] и, следовательно, представляют определенный интерес среди оптимизационных задач и требуют новых сложных техник их решения.

Следовательно, задачи квадратичного программирования важны и интересны как с математической, так и с прикладной точек зрения.

В последние несколько лет, наряду с задачами оптимизации квадратичных функций при линейных или выпуклых квадратичных ограничениях, для которых с успехом применяются методы локального поиска [7,8], в квадратичной оптимизации исследуются задачи и с невыпуклыми квадратичными функциями. Для решения таких задач существуют различные методы глобальной оптимизации. Но, в виду сложности алгоритмов глобальной оптимизации, их применение требует существенных вычислительных усилий [9].

В статье рассматривается задача нахождения допустимой точки системы невыпуклых квадратичных неравенств. Данная задача играет важную роль в качестве вспомогательной при решении более общих задач глобальной оптимизации и имеет множество практических приложений. Описывается редукция этой задачи к задаче вогнутого программирования, для решения которой применяются методы локального поиска. В

¹Работа поддержана грантом РФФИ N 03-01-00518

настоящее время программное обеспечение для методов локального поиска разработано лучше, чем программное обеспечение для решения задач глобальной оптимизации. И применение методов локального поиска существенно проще с программистской точки зрения, по сравнению с алгоритмами глобальной оптимизации. Приводятся результаты численных экспериментов.

1. Редукция к задаче вогнутого квадратичного программирования

Рассматривается задача нахождения допустимой точки системы невыпуклых квадратичных неравенств

$$g_i(x) \leq 0, i = 1, \dots, m, \quad (1.1)$$

$$x \in X, \quad (1.2)$$

где $g_i(x) = x^T Q^i x + x^T c^i + r_i$, Q^i - некоторые $n \times n$ симметричные матрицы, $c^i \in R^n$, $r_i \in R$, X - выпуклый компакт. Для решения поставленной задачи в общем случае применяются методы глобальной оптимизации.

Рассмотрим способ сведения задачи (1.1)-(1.2) к задаче вогнутого программирования. Сопоставим задаче (1.1)-(1.2) следующую задачу:

$$\max_{i=1, \dots, m} g_i(x) \rightarrow \min, \quad (1.3)$$

$$x \in X. \quad (1.4)$$

Используя свойство диагонального доминирования [10], можно определить константу $N > 0$ такую, что матрицы $Q^i + N \cdot I$, $i = 1, \dots, m$ положительно определены, I - единичная матрица и следовательно, можно представить целевую функцию (1.3) в виде разности двух выпуклых функций, а задачу (1.3)-(1.4) переписать в следующем виде

$$\max_{i=1, \dots, m} \{\bar{g}_i(x)\} - N \|x\|^2,$$

$$x \in X,$$

где $\bar{g}_i(x) = g_i(x) + N \|x\|^2$ - выпуклые квадратичные функции.

Вводя новую переменную t , перепишем последнюю задачу в следующем эквивалентном виде

$$t - N \|x\|^2 \rightarrow \min,$$

$$\max_{i=1, \dots, m} \{\bar{g}_i(x)\} \leq t,$$

$$x \in X.$$

В результате приходим к задаче (1.5)-(1.7)

$$t - N \|x\|^2 \rightarrow \min, \quad (1.5)$$

$$\bar{g}_i(x) \leq t, i = 1, \dots, m, \quad (1.6)$$

$$x \in X. \quad (1.7)$$

При нахождении допустимой точки целевая функция $t - N \|x\|^2 \leq 0$.

Задача (1.5)-(1.7) является задачей вогнутого программирования и для ее решения применяются методы локального поиска [2].

1.1. Нахождение локального решения

В данной статье для нахождения хорошего локального решения задачи (1.5)-(1.7) применяется следующий подход.

Поскольку целевая функция вогнутая, то ее минимум достигается на границе допустимой области (1.6),(1.7).

Рассматриваемый подход предполагает решение $2n + 1$ задачи выпуклого программирования и одну задачу локальной квадратичной оптимизации. Сначала решается задача

$$t \rightarrow \min,$$

при условиях (1.6)-(1.7). Обозначим через V^0 ее решение. Затем $2n$ задач следующего вида:

$$\pm x_j \rightarrow \min, j = 1, \dots, n,$$

при условиях (1.6), (1.7) и пусть $V^i, i = 1, \dots, 2n$ - оптимальные решения этих задач.

Из всех найденных решений $V^i, i = 0, \dots, 2n$ выбирается

$$\bar{V} = \arg \min_{0 \leq i \leq 2n} \{t^i - \|V^i\|^2\}$$

. Неположительные значения $t^i - \|V^i\|^2$ соответствуют допустимым точкам системы невыпуклых квадратичных неравенств (1.1), (1.2). Если все значения $t^i - \|V^i\|^2$ положительны, то найденное решение \bar{V} является начальным приближением для задачи (1.5)-(1.7), которая решается методами локального спуска.

Пример.

Рассматривается система невыпуклых квадратичных неравенств следующего вида:

$$g_1(x, y) = -xy + 1 \leq 0, \quad (1.8)$$

$$g_2(x, y) = x^2 + y^2 - 5 \leq 0, \quad (1.9)$$

$$-5 \leq x \leq 5, \quad (1.10)$$

$$-5 \leq y \leq 5. \quad (1.11)$$

Функция $g_2(x, y)$ выпуклая. Для того, чтобы функцию $g_1(x, y)$ привести к выпуклому виду достаточно принять $N = 1$.

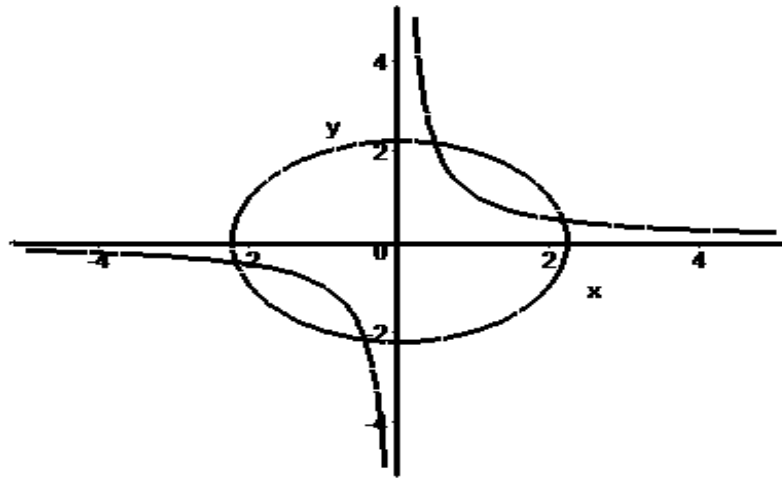


Рис.1. Допустимая область системы (1.8)-(1.11)

Тогда, в соответствии с изложенной выше методикой, приходим к следующей задаче вогнутого программирования.

$$\begin{aligned}
 t - x^2 - y^2 &\rightarrow \min, \\
 x^2 - xy + y^2 + 1 &\leq t, \\
 2x^2 + 2y^2 - 5 &\leq t, \\
 -5 &\leq x \leq 5, \\
 -5 &\leq y \leq 5, \\
 -1000 &\leq t \leq 1000.
 \end{aligned}$$

Для решения этой задачи применялись методы локального поиска и система алгебраического моделирования GAMS [11]. Результаты решения вспомогательных задач выпуклого программирования представлены в следующей таблице.

Таблица 1.1 Результаты решения вспомогательных задач.

Оптимизируемый параметр	x	y	t	$F = t - x^2 - y^2$
$\min t$	0	0	1	1
$\max x$	5	0	45	20
$\min x$	-5	5	1000	950
$\max y$	5	5	1000	950
$\min y$	5	-5	1000	950

Из решения приведенных пяти задач была выбрана стартовая точка $t = 1$ и в результате получено решение рассматриваемой задачи вогнутого программирования $x^* = -1.414$, $y^* = -1.414$, $F^* = -1$, которое удовлетворяет исходной системе неравенств (1.8)-(1.11).

2. Результаты численных экспериментов

Для проверки рассматриваемой методики моделировались тестовые задачи для системы невыпуклых квадратичных неравенств следующего вида:

$$g_i(x) = x^T Q^i x + x^T c^i + r_i \leq 0, i = 1, \dots, m, \quad (2.1)$$

$$x \in P = \{x : \alpha_j \leq x_j \leq \beta_j, j = 1, \dots, n\}, \quad (2.2)$$

где $g_i(x)$ - невыпуклые квадратичные функции, α_j, β_j - некоторые числа.

Тестовые задачи формировались согласно следующей процедуре:

1. Выбиралась точка $x^0 \in P$.
2. Матрицы Q^i и векторы c^i формировались случайным образом.
3. Задавалось число $\Delta < 0$ так, чтобы выполнялось соотношение $r_i = \Delta - (x^0)^T Q^{iT} x^0 - (x^0)^T c^i$, тогда x^0 будет допустимой точкой системы (2.1), (2.2) поскольку $g_i(x^0) = \Delta < 0$.

Расчеты проводились на PC Celeron/1200 Mhg/ 256 Mb с использованием системы алгебраического моделирования GAMS [11]. Решались случайным образом сгенерированные тестовые задачи с числом переменных от двух до пятидесяти. С ростом размерности наблюдалось уменьшение числа найденных допустимых точек, что видно из следующей таблицы.

Таблица 2.1 Результаты решения тестовых задач

Число переменных	Число допустимых точек	Время
2	4	00:00.04
5	4	00:00.07
10	4	00:00.25
15	3	00:00.30
20	1	00:01.00
25	1	00:13.00
30	1	00:15.00
35	1	00:45.00
50	1	01:30.00

Уменьшение числа найденных допустимых точек связано со сложностью решаемых задач, которая еще более возрастает с ростом размерности. Тем не менее, применяемая методика позволяет без использования методов глобальной оптимизации найти по крайней мере одну допустимую точку исходной системы невыпуклых квадратичных неравенств.

Предлагаемый подход может быть распространен и на решение общих задач невыпуклого квадратичного программирования.

СПИСОК ЛИТЕРАТУРЫ

- [1] Ириков В.А., Тренев В.Н., Распределенные системы принятия решений (теория и приложения).-М.: Наука, 1999 г.
- [2] Horst R., Pardalos P.M., Thoai N., Introduction to Global Optimization.-Dordrecht: Kluwer Academic Publishers, 1995, 320 p.
- [3] Гамм А.З., Таирова Е.В., Хамисов О.В., Поиск равновесных точек в моделях рыночных механизмов ЭЭС.// Известия РАН. Энергетика, 2000 г., N 6, с.57-63.
- [4] Хамисов О.В., Таирова Е.В., О коалиционном поведении в рыночных моделях электроэнергетических систем.// Известия РАН. Энергетика, 2002, N 40-47.
- [5] Fiodas C.A., Visweswaran V., Quadratic optimization.// Handbook of Global Optimization, Kluwer Academic Publishers, 1995, pp.217-270.
- [6] Pardalos P., Vavasis S., Quadratic programming with one negative eigenvalue is NP-hard //J. Global Optimization.-1991, V.1, pp.15-22.
- [7] Булатов В.П., Алексеева Т.Л., Численные методы поиска всех вещественных корней систем алгебраических уравнений и их приложения// Труды XI Международной Байкальской школы-семинара "Методы оптимизации и их приложения", Иркутск, Байкал, 5-12 июля 1998 г., с. 21-31.
- [8] Булатов В.П., Ганхуяг Д., Поиск вещественных корней систем нелинейных алгебраических уравнений и необходимое условие оптимальности.// Современные методы оптимизации и их приложения к моделям энергетики.-Новосибирск: Наука, 2003, с.56-69.
- [9] Horst R, Tuy H., Global Optimization (Deterministic Approaches).-Berlin, New York, Springer-Verlag, 1993.
- [10] Гантмахер Ф.Р., Теория матриц.-М.: Наука, 1988, 548 с.
- [11] Brook A., Kendrick D., Meeraus A., GAMS. Release 2.25. A user's guide.// GAMS Development Corporation, 1996.

УСТОЙЧИВАЯ АППРОКСИМАЦИЯ ДАННЫХ ЛИНЕЙНЫМИ РЕГРЕССИОННЫМИ МОДЕЛЯМИ

А.Н. Тырсин

Челябинский государственный университет, Челябинск
e-mail: at2001@yandex.ru

Аннотация. В статье предложен новый метод устойчивого построения линейных регрессионных моделей – обобщенный метод наименьших модулей. Приводится теоретическое обоснование метода. Рассматривается экспериментальная апробация на тестовых данных.

Ключевые слова: оптимизация, аппроксимация, робастность, регрессия, модель, многоэкстремальная, вогнутость, математическая.

Введение

Многие оптимизационные задачи сводятся к аппроксимации исходных данных некоторой моделью [1]. Одной из наиболее распространенных моделей является множественная линейная регрессия [2]. Отметим, что требование линейности не столь существенно, поскольку в большинстве практических задач путем соответствующей замены переменных модель удастся линеаризовать. Множественная линейная регрессия имеет вид

$$\mathbf{y} = \mathbf{X}\mathbf{a} + \boldsymbol{\epsilon}, \quad (1)$$

где $\mathbf{y}^T = (y_1, \dots, y_n)$ – вектор значений зависимой переменной; $\mathbf{X} = \{x_{ij}\}$, $i = 1, 2, \dots, n$, $j = 0, 1, \dots, m$ – матрица значений объясняющих переменных; $\boldsymbol{\epsilon}^T = (\epsilon_1, \dots, \epsilon_n)$ – вектор случайных ошибок; $\mathbf{a}^T = (a_0, a_1, \dots, a_n)$ – вектор коэффициентов регрессии.

Среди прикладных статистических методов построения регрессионных зависимостей (1) наиболее распространен метод наименьших квадратов (МНК). Он позволяет получить глубокие статистические результаты в предположении, что случайные ошибки ϵ_i распределены по нормальному (гауссовскому) закону. Однако в ряде случаев возникают ситуации, когда предположение о гауссовой модели случайных ошибок не выполняется. При этом использование МНК может привести к значительным ошибкам. С целью обеспечения устойчивости оценок относительно отклонений случайных ошибок от гауссовой модели разработан ряд робастных методов: метод наименьших модулей (МНМ) [3], метод минимакса [4], ранговые [5] и знаковый [6] методы. Термин "робастность" означает устойчивость или слабочувствительность статистической процедуры к незначительным отклонениям данных от начальных предположений [4]. Данные методы основаны на более общих предположениях относительно случайных ошибок чем МНК.

1. Постановка задачи

Наиболее распространенной моделью исследования устойчивости является модель засорения Тьюки-Хьюбера

$$F(x) = (1 - \gamma)F_\epsilon(x) + \gamma H(x),$$

где $F_\epsilon(x)$ – функция распределения случайных ошибок ϵ_i , обладающая "хорошими" свойствами (как правило, нормальностью); $H(x)$ – функция распределения засорений, имеющих вид выбросов, как по уровню, так и по дисперсии; γ – вероятность появления выброса.

В теоретических исследованиях обычно в качестве засорения используют случайные величины, у которых математическое ожидание стремится к бесконечности или не существует. Кроме того, обычно предполагается известной степень засорения g , что на практике, как правило, невозможно. Представляется актуальным обеспечение устойчивости оценок параметров модели (1) не для гипотетического случая неограниченности величины засорения, а разработка робастных процедур, удовлетворяющих следующим требованиям [7]:

- они должны быть ориентированны на ограниченное засорение, находящееся "на грани", между "интуитивно возможным" и "интуитивно невозможным";
- степень засорения должна находиться в достаточно широких пределах;
- скорость сходимости выборочных оценок к асимптотическим результатам должна быть достаточно высокой, чтобы обеспечить их практическое применение.

2. Метод решения

Рассмотрим следующую задачу построения регрессионной модели с помощью обобщенного метода наименьших модулей (ОМНМ) [8]:

$$(\hat{a}_0, \dots, \hat{a}_m) = \arg \min \sum_{i=1}^n \left| y_i - a_0 - \sum_{j=1}^m a_j x_{ij} \right|^\lambda, \quad 0 < \lambda < 1. \quad (2)$$

Данное оценивание имеет преимущество перед МНМ и знаковым методом в случае, когда выбросы в ошибках разного знака имеют различные вероятности. Как известно, целевая функция задачи (2) имеет множество локальных минимумов. Предложенный в [3] метод вариационно-взвешенных квадратических приближений позволяет найти с заданной погрешностью один из локальных минимумов, ближайший к начальному приближению $\mathbf{a}^{(0)}$ вектора \mathbf{a} . Глобальный минимум не находится. Более того, идея данного подхода заключается в итерационном приближении решения к некоторому единственному экстремуму, который по умолчанию является глобальным. Этим объясняется, что данный подход нашел применение лишь для реализации метода наименьших модулей.

Теорема 1. Пусть имеется выборка наблюдений $(x_{i1}, \dots, x_{im}, y_i)$, $i = 1, 2, \dots, n$, и задана функция

$$Q(a_0, \dots, a_m) = \sum_{i=1}^n \left| y_i - a_0 - \sum_{j=1}^m a_j x_{ij} \right|^\lambda, \quad 0 < \lambda < 1. \quad (3)$$

Локальными минимумами функции (3) являются все узловые точки пересечения гиперплоскостей

$$a_0 = y_i - \sum_{j=1}^m a_j x_{ij}, \quad (i = 1, \dots, n). \quad (4)$$

Согласно теореме 1 поиск решения задачи минимизации (2) сводится к перебору узловых точек.

Оценка (2) обладает рядом полезных свойств. Во-первых, в отличие от МНМ, большие ошибки здесь уменьшаются. Во-вторых, данная оценка, в отличие от МНМ-оценки

и знаковой оценки, гораздо менее чувствительна к несимметричным выбросам. Однако, одновременно наблюдается рост дисперсии оценок по сравнению с методом наименьших квадратов. Это можно объяснить тем, что малые ошибки в задаче (2) увеличиваются по сравнению с МНК и МНМ.

Задачу (2) можно обобщить на следующий случай. Очевидно, что функция $f(x) = |x|^\lambda$ при $0 < \lambda < 1$ является вогнутой. Оказывается, что использование вместо нее любой монотонно возрастающей, дважды непрерывно-дифференцируемой на положительной полуоси и равной нулю при $x = 0$ вогнутой функции, сохраняет верной теорему 1.

Теорема 2. Пусть имеется выборка наблюдений $(x_{i1}, \dots, x_{im}, y_i)$, $i = 1, 2, \dots, n$, и задана функция

$$Q(a_0, \dots, a_m) = \sum_{i=1}^n g \left(\left| y_i - a_0 - \sum_{j=1}^m a_j x_{ij} \right| \right), \quad (5)$$

где $g(x)$ – некоторая монотонно возрастающая, всюду дважды непрерывно-дифференцируемая на положительной полуоси функция (при $x > 0$), причем $g(0) = 0$, $g''(x) < 0$.

Локальными минимумами функции (5) являются все узловые точки пересечения гиперплоскостей (4).

В классе удовлетворяющих условиям теоремы 2 функций представляет интерес

$$g(x) = \arctan |x|. \quad (6)$$

При этом решается следующая задача

$$(\hat{a}_0, \dots, \hat{a}_m) = \arg \min \sum_{i=1}^n \arctan \left(\left| y_i - a_0 - \sum_{j=1}^m a_j x_{ij} \right| \right). \quad (7)$$

Функция (6) обладает следующими особенностями. Во-первых, она асимптотически ограничена. Иными словами, оценки (7), в отличие от (2) устойчивы к наличию в измеренных данных сверхбольших выбросов. Во-вторых, при малых погрешностях измерений дисперсия оценок (2) будет значительно превосходить дисперсию оценок (7). В-третьих, на основе функции (7) можно построить квазиоптимальные оценки, сочетающие устойчивость по отношению к выбросам с низкой дисперсией при малых погрешностях. Это можно объяснить тем, что график функции

$$g(x) = \arctan (|x|^\tau) \quad (8)$$

для τ вблизи 3 при малом значении аргумента похож на параболу, соответствующую МНК-оценкам.

Квазиоптимальность оценок вызвана тем, что при функции (10) теряет свойство вогнутости, и в общем случае может иметь локальные минимумы в стационарных точках.

3. Эксперимент

Проведем сравнительный анализ ОМНМ-, МНМ-, МНК-оценок. В данной статье ограничимся случаем конечных выборок приемлемого объема с точки зрения практики. Рассмотрим следующую модель парной линейной регрессии

$$y_i = \alpha + \epsilon_i, \quad (i = 1, \dots, n),$$

где случайные ошибки ϵ_i имеют распределение $F(x) = (1 - \gamma)N(0, \sigma^2) + \gamma N(\mu, \sigma_1^2)$, представляющее собой смесь двух нормальных распределений. Для определенности зададим: $\alpha = 0$; $\sigma = 1$; $\mu = 2$; $\sigma_1 = 3$; $n = 200$; $M = 4000$ – число испытаний. Случайные ошибки моделируем с помощью датчика случайных чисел. Результаты расчетов (абсолютные погрешности Δa и дисперсии s_a^2 оценки a параметра α представлены на рис. 1 и 2. Здесь обозначено: 1 – МНК-оценки; 2 – МНМ-оценки; 3-5 – ОМНМ-оценки (2) с $\lambda = 0,75$, $\lambda = 0,5$ и $\lambda = 0,25$; 6-8 – ОМНМ-оценка (7) с $\tau = 1$, , а также квазиоптимальные ОМНМ-оценки с $\tau = 3$, $\tau = 5$.

Выводы

1. Предложен новый метод устойчивого оценивания линейных моделей – обобщенный метод наименьших модулей.

2. ОМНМ-оценки имеют гораздо более высокую устойчивость при одностороннем засорении по сравнению с МНМ-оценками. Одновременно, они сохраняют устойчивость и при симметричном засорении.

3. В ОМНМ предпосылки относительно случайных ошибок становятся еще более общими по сравнению с другими робастными методами. Фактически требуется лишь, чтобы случайные ошибки были независимы. Равная вероятность появления выбросов, требуемая в МНМ и знаковом методе здесь не обязательна.

Список литературы

- [1] П.-Ж. Лоран *Аппроксимация и оптимизация*. М.: Мир, 1975, 496 с.
- [2] С.А. Айвазян, В.С. Мхитарян *Прикладная статистика и основы эконометрики*. М.: ЮНИТИ, 1998, 1022 с.
- [3] В.И. Мудров, В.Л. Кушко *Методы обработки измерений. Квазиправдоподобные оценки*. М.: Радио и связь, 1983, 304 с.
- [4] П. Хьюбер *Робастность в статистике*. М.: Мир, 1984, 304 с.
- [5] Т. Хеттманспергер *Статистические выводы, основанные на рангах*. М.: Финансы и статистика, 1987, 334 с.
- [6] М.В. Болдин , Г.И. Симонова , Ю.Н. Тюрин *Знаковый статистический анализ линейных моделей*. М.: Наука. Физматлит, 1997, 288 с.
- [7] А.И. Орлов *Эконометрика*. М.: Экзамен, 2002, 576 с.
- [8] А.Н. Тырсин, Д.С. Воронина *Устойчивый метод построения линейных регрессионных моделей при несимметричном распределении выбросов*. – Обозрение прикладной и промышленной математики, 2004, т. 11, вып. 4, с. 940-941.

ROBUST APPROXIMATION OF EXPERIMENTAL DATA OF THE LINEAR REGRESSION MODELS

A.N. Tyrsin

Chelyabinsk State University, Chelyabinsk
e-mail: at2001@yandex.ru

Abstract. In article the new method of steady construction linear regression models – the generalized method of the least modules is offered. It is resulted theoretical justification of a method. Experimental approbation on the test data is considered.

Key words: optimization, approximation, robustness, regression, model, multiextremal, concave, mathematical.

ГЛОБАЛЬНАЯ ОПТИМИЗАЦИЯ

НОВЫЙ СПОСОБ УЧЕТА ОГРАНИЧЕНИЙ В УСЛОВНЫХ ЗАДАЧАХ ГЛОБАЛЬНОЙ ОПТИМИЗАЦИИ ¹

К.А. Баркалов

*Нижегородский государственный университет им.Н.И.Лобачевского, Нижний Новгород
e-mail: barkalov@hotmail.ru*

Аннотация. Работа посвящена дальнейшему развитию нового подхода к решению многоэкстремальных задач с невыпуклыми ограничениями. Характерной чертой этого подхода, не использующего идей метода штрафных функций, является отдельный учет каждого ограничения задачи. При этом последовательно осуществляемая в точке каждой итерации проверка выполнимости ограничений прерывается при обнаружении первого нарушения. Новое предложение состоит в том, что на каждой итерации адаптивно определяется свой порядок проверки ограничений. Это позволяет начинать проверку или с ограничения, для которого более вероятно нарушение в выбранной точке очередной итерации, или с более простого ограничения, переходя в конце к более сложным. Тем самым форсируется завершение итерации при меньших вычислительных затратах. Даны достаточные условия сходимости метода. Приведены результаты сравнения алгоритмов с фиксированным и с адаптивным порядками осуществления проверок. Сравнение проведено путем численного решения обоими методами многих сотен случайно генерируемых многоэкстремальных тестовых задач с невыпуклыми ограничениями.

Ключевые слова: глобальная оптимизация, многомерные задачи, невыпуклые ограничения.

Введение

Статья продолжает разработку нового подхода к минимизации многоэкстремальных функций при невыпуклых ограничениях, развитого в [1]-[6] и получившего название *индексного метода* глобальной оптимизации.

Подход основан на отдельном учете каждого ограничения задачи и не связан с использованием штрафных функций. При этом решение многомерных задач сводится к решению эквивалентных им одномерных задач. Соответствующая редукция основана на использовании разверток единичного отрезка вещественной оси на гиперкуб. Роль таких разверток играют непрерывные однозначные отображения типа кривой Пеано, называемые также кривыми, заполняющими пространство. Численные методы, позволяющие эффективно использовать аппарат таких отображений, детально разработаны в [1], [5] и [6].

В соответствии с правилами индексного метода каждая итерация, называемая испытанием в соответствующей точке области поиска, включает последовательную проверку выполнимости ограничений задачи в этой точке. При этом обнаружение первого нарушенного ограничения прерывает испытание и инициирует переход к точке следующей итерации. Если при этом ввести нумерацию ограничений в порядке возрастания вычислительных затрат на их проверку (т.е. определить, что сначала проверяются более простые ограничения), то это может существенно снизить трудоемкость решения задачи в целом. Сказанное актуально для задач, в которых оценка функционалов требует заметных вычислительных ресурсов, связанных с необходимостью численного моделирования пове-

¹Работа выполнена при финансовой поддержке РФФИ (код гранта 01-01-00587).

дения систем, характеризуемых этими функционалами (как это имеет место, например, для многих задач оптимального проектирования).

Постановка задачи условной глобальной оптимизации

Рассмотрим одномерную задачу условной глобальной оптимизации вида

$$\varphi^* = \varphi(y^*) = \min \{ \varphi(y) : y \in D, g_j(y) \leq 0, 1 \leq j \leq m \} \quad (1)$$

$$D = \{ y \in R^N : -0.5 \leq y_i \leq 0.5, 1 \leq i \leq N \}$$

в предположении, что целевая функция φ (в дальнейшем обозначаемая также g_{m+1} , т.е. $g_{m+1}(y) = \varphi(y)$) и левые части ограничений $g_j, 1 \leq j \leq m$, являются определенными в области D липшицевыми функциями с соответствующими константами $L_j, 1 \leq j \leq m+1$. В общем случае все эти функции могут быть многоэкстремальными. Также предполагается, что ограничение, имеющее номер j , выполняется во всех точках области

$$Q_j = \{ y \in D, g_j(y) \leq 0 \}, 1 \leq j \leq m,$$

которую будем называть допустимой областью для этого ограничения. При этом допустимая область Q исходной задачи (1) определяется условием $Q = Q_1 \cap \dots \cap Q_m$

Известно, что с помощью кривых типа развертки Пеано $y(x)$, однозначно отображающих отрезок $[0, 1]$ на N -мерный гиперкуб D

$$D = \{ y \in R^N : -0.5 \leq y_i \leq 0.5, 1 \leq i \leq N \} = \{ y(x) : 0 \leq x \leq 1 \},$$

задачу (1) можно редуцировать к одномерной задаче

$$\varphi^* = \varphi(y(x^*)) = \min \{ \varphi(y(x)) : x \in [0, 1], g_j(y(x)) \leq 0, 1 \leq j \leq m \}$$

Редукция такого типа позволяет использовать для решения многомерной задачи (1) идеи индексного подхода к решению задач условной глобальной оптимизации.

Испытание в точке $y \in D$ состоит в последовательном вычислении величин $g_1(y), \dots, g_\nu(y)$, где значения индекса $\nu \leq m$ определяются условиями

$$y \in Q_j, 1 \leq j < \nu, y \notin Q_\nu,$$

При этом выявление первого нарушенного ограничения прерывает испытание в точке y .

В случае, когда точка y допустима, т.е. когда $y \in Q$, испытание включает вычисление значений всех функционалов задачи. При этом значение индекса принимается равным величине $\nu = m + 1$.

В связи с изложенным, пару $\nu = \nu(y), z = g_\nu(y)$, где индекс ν заведомо лежит в границах $1 \leq \nu \leq m + 1$, будем называть *результатом испытания* в точке y .

Описание нового подхода к учету ограничений

Одной из важных черт описываемого алгоритма является использование при его разработке введенного в [5] понятия ϵ -резервированного решения, которое можно интерпретировать как некоторый аналог условия регулярности в классических задачах нелинейного программирования.

Определение. Точка y_ϵ называется ϵ -резервированным решением задачи (1), если выполнено условие

$$\varphi(y_\epsilon) = \min \{ \varphi(y) : y \in D, g_j(y) \leq -\epsilon_j, 1 \leq j \leq m \}$$

Положительные значения резервов $\epsilon_j, 1 \leq j \leq m$, вводимые (как параметры) в схему алгоритма, обеспечивают ускорение сходимости к решению. При этом априори неизвестные значения этих параметров могут адаптивно оцениваться в процессе решения задачи (см. [7]).

Также для многих прикладных задач естественно допустить, что ограничение, нарушенное в некоторой точке области поиска, не будет выполняться и в некоторой окрестности этой точки. В таком случае (для сокращения числа проверок) может оказаться полезным начинать испытания в точках этой окрестности с анализа выполнимости указанного ограничения. В работе [8] предложен модифицированный индексный метод, допускающий адаптивное изменение порядка проверки ограничений с учетом информации, получаемой в ходе поиска решения.

Модификация. Нумерацию ограничений и целевой функции, введенную в формулировке задачи (1) и определенную последовательностью целых чисел $H = \{1, 2, \dots, m, m+1\}$ будем рассматривать как *базовую*. Примем, что каждой конкретной точке y из области поиска D может быть сопоставлен свой порядок проверки ограничений задачи. При этом порядок такой проверки, соответствующий точке y_i , будем задавать как некоторую перестановку $H(y_i)$ базовых номеров, т.е.

$$H(y_i) = \{j_{i1}, \dots, j_{im}, j_{i,m+1}\}, j_{i,m+1} = m+1.$$

Далее на каждой итерации метода определяется новый порядок проверки ограничений $H(y_i)$ таким образом, чтобы минимизировать вычислительные затраты на проведение испытания.

Введение адаптивного порядка проверки выполнения ограничений можно рассматривать как продолжение тенденции к более полному использованию информации, получаемой в процессе решения задачи.

Список литературы

- [1] Стронгин Р.Г. *Численные методы в многоэкстремальных задачах*. М.: Наука, 1978.
- [2] Стронгин Р.Г., Маркин Д.Л. *Минимизация многоэкстремальных функций при невыпуклых ограничениях* Кибернетика, 1986, №4, с.63-69.
- [3] Маркин Д.Л., Стронгин Р.Г. *Метод решения многоэкстремальных задач с невыпуклыми ограничениями, использующий априорную информацию об оценках оптимума*. Ж. вычисл. матем. и матем. физ, 1987, т.27, №1, с.33-39.
- [4] Стронгин Р.Г. *Поиск глобального оптимума*. М.: Знание, 1990.
- [5] Стронгин Р.Г. *Параллельная многоэкстремальная оптимизация с использованием множества разверток*. Ж. вычисл. матем. и матем. физ, 1991, т.31, №8, с.1173-1185.

- [6] Strongin R.G., Sergeyev Ya.D. *Global optimization with non-convex constraints. Sequential and parallel algorithms.* Kluwer Acad. Publ., 2000.
- [7] Стронгин Р.Г., Баркалов К.А. *О сходимости индексного алгоритма в задачах условной оптимизации с ϵ -резервированными решениями.* Математические вопросы кибернетики, 1999, с.273-288.
- [8] Стронгин Р.Г., Баркалов К.А. *Метод глобальной оптимизации с адаптивным порядком проверки ограничений* Ж. вычисл. матем. и матем. физ, 2002, Т.42, №9, С.1338-1350.

A NEW METHOD FOR CONSTRAINTS ACCOUNTING IN CONSTRAINED GLOBAL OPTIMIZATION PROBLEMS

K.A.Barkalov

Nizhni Novgorod State University
e-mail: barkalov@hotmail.ru

Abstract. A new approach to solving multimodal problems with nonconvex constraints is proposed. A specific feature of this approach, which does not rely on the penalty function method, is a separate consideration of every constraint. The sequential checking for constraints performed at every iteration point is stopped as soon as the first violation is discovered. A new proposition is that the order of checking for constraints is adaptively determined at each iteration step. This makes it possible to begin checking for constraints with the one is most likely to be violated at the current point. Thus, the iteration step can be completed at a lower cost. Sufficient conditions for convergence of the method are formulated. The results obtained by comparing algorithms with fixed and adaptive order of checking for constraints are presented. The comparison was carried out by applying both methods to compute hundreds of randomly generated multimodal test problems with nonconvex constraints.

Key words: multiextremal optimization, multidimensional optimization, nonconvex constraints, global optimization technique.

A GLOBAL METHOD FOR SOLVING THE CONVEX QUADRATIC MAXIMIZATION PROBLEM

B.Barsbold, Ch.Altannar and R.Enkhbat

*The School of Mathematics and Computer Science
National University of Mongolia, Mongolia
e-mail: renkhat@ses.edu.mn*

Abstract. In this paper we consider the problem of maximizing convex quadratic function over a polyhedral set. We simplified an algorithm for solving the problem given in [1] and improved the upper bound of the approximation error proposing a new technique for construction of approximation of the level set.

Key words: approximation, simple set, concave programming, conjugate directions.

Introduction

Consider the quadratic convex maximization problem.

$$\max_{x \in D} f(x) = \max_{x \in D} [\langle Cx, x \rangle + \langle d, x \rangle] \quad (1)$$

where C is a positive semidefinite symmetric $n \times n$ matrix, D is a nonempty closed subset of R^n and d is an n vector. The work in [1] has proposed a method for solving Problem (1). Their method is based on the global optimality conditions by Strekalovsky ([2]) and uses approximations sets on the level set of the objective function and the eigenvectors of matrix C .

The following global optimality conditions for Problem (1) were first formulated by Strekalovsky in [2, 3]

Theorem 1. ([2, 3]) *Let $z \in D$ be such that $\nabla f(z) \neq 0$. Then z is a solution of problem (1) if and only if*

$$\langle \nabla f(y), x - y \rangle \leq 0 \text{ for all } y \in E_{f(z)}(f) \text{ and } x \in D, \quad (2)$$

where $E_c(f) = \{y \in R^n : f(y) = c\}$.

1. First Order Approximation of the Level Set

To construct a numerical method for solving Problem (1) based on the optimality conditions (2), we assume that D is a polytope and C is a symmetric positive definite $n \times n$ matrix; i.e.,

$$\max_{Ax \leq b} f(x) = \max_{Ax \leq b} [\langle Cx, x \rangle + \langle d, x \rangle], \quad (3)$$

where A is an $r \times n$ matrix and $b \in R^r$, $d \in R^n$.

Definition 1. The set $E_{f(z)}(f)$ defined by

$$E_{f(z)}(f) = \{y \in R^n : f(y) = f(z)\}$$

is called the level set of f at z .

Note that checking the optimality conditions (2) for Problem (3) requires to solve the linear programming problem

$$\max_{Ax \leq b} \langle \nabla f(y), x - y \rangle$$

for each $y \in E_{f(z)}(f)$. This is a hard problem. Thus, we need to find an appropriate approximation set such that one could check the optimality conditions at a finite number of points.

Definition 2. The set A_z^m defined by

$$A_z^m = \{y^1, y^2, \dots, y^m : y^i \in E_{f(z)}(f), i = 1, 2, \dots, m\} \quad (4)$$

is called the approximation set to the level set $E_{f(z)}(f)$ at the point z .

When we construct A_z^m , we specify the elements $y^i \in R^n$, $z \in D$, and the integer m . Assume that A_z^m for each $y^i \in A_z^m$, $i = 1, 2, \dots, m$, solve the auxiliary problems

$$\max_{x \in D} \langle \nabla f(y^i), x \rangle.$$

Let u^i , $i = 1, 2, \dots, m$ be the solutions of these problems, which always exist due to compactness of D :

$$\langle \nabla f(y^i), u^i \rangle = \max_{x \in D} \langle \nabla f(y^i), x \rangle. \quad (5)$$

Define θ_m as follows:

$$\theta_m = \max_{i=1,2,\dots,m} \langle \nabla f(y^i), u^i - y^i \rangle. \quad (6)$$

There are some properties of A_z^m and θ_m .

Lemma 1. ([1]) If there is a point $y^i \in A_z^m$ for $z \in D$ such that $\langle \nabla f(y^i), u^i - y^i \rangle > 0$, where $u^i \in D$ satisfies $\langle \nabla f(y^i), u^i \rangle = \max_{x \in D} \langle \nabla f(y^i), x \rangle$, then

$$f(u^i) > f(z)$$

holds.

For a positive definite and symmetric matrix C , we can define the scalar product of the vectors x and $y \in R^n$ by $\langle x, y \rangle_C = \langle Cx, y \rangle$ and the norm of the vector x by $\|x\|_C = \sqrt{\langle Cx, x \rangle}$. Let x^* be a minimum of the function f over R^n ; i.e., $f(x^*) = \min_{x \in R^n} f(x)$. Then we can rewrite Problem (3) in its equivalent form as follows.

$$\max_{x \in D} \|x - x^*\|_C^2 + f(x^*) = \max_{x \in D} \langle C(x - x^*), x - x^* \rangle + f(x^*). \quad (7)$$

The gradient of the function f can be given as

$$\nabla f(x) = 2C(x - x^*). \quad (8)$$

C -orthonormal directions constitute an orthonormal basis in space R^n . Consider C -orthonormal directions l^1, \dots, l^n . For a given $z \in D$, we define the elements $y^1, \dots, y^{2n} \in R^n$ as follows:

$$\begin{aligned} y^j &= x^* + \alpha l^j, \\ y^{n+j} &= x^* - \alpha l^j, \quad j = 1, 2, \dots, n, \end{aligned} \quad (9)$$

where $\alpha = \sqrt{f(z) - f(x^*)}$.

Let u^1, \dots, u^n and u^{n+1}, \dots, u^{2n} are solutions to the problems

$$\max_{x \in D} \langle Cl^j, x \rangle \text{ and } \max_{x \in D} \langle -Cl^j, x \rangle \text{ for } j = 1, \dots, n,$$

respectively. Introduce γ_j and γ_{n+j} :

$$\begin{aligned} \gamma_j &= \max_{x \in D} \langle Cl^j, x - x^* \rangle = \langle Cl^j, u^j - x^* \rangle \\ \gamma_{n+j} &= \max_{x \in D} \langle -Cl^j, x - x^* \rangle = \langle -Cl^j, u^{n+j} - x^* \rangle \end{aligned} \quad (10)$$

for $j = 1, \dots, n$. Define the set $R(D)$ and $T(z)$ as follows:

$$\begin{aligned} R(D) &= \left\{ x \in R^n : x = x^* + \sum_{j=1}^n \beta_j l^j, \quad -\gamma_{n+j} \leq \beta_j \leq \gamma_j, \quad j = 1, \dots, n \right\}, \\ T(z) &= \left\{ x \in R^n : x = x^* + \sum_{j=1}^n \mu_j (y^j - x^*), \quad |\mu_j| \leq 1, \quad j = 1, \dots, n \right\}. \end{aligned}$$

Lemma 2.

- a. $R(D) = \{x \in R^n : \langle \nabla f(y^j), x - x^* \rangle \leq \langle \nabla f(y^j), u^j - x^* \rangle, \quad j = 1, \dots, 2n\}$,
- b. $T(z) = \{x \in R^n : \langle \nabla f(y^j), x - x^* \rangle \leq 2(f(z) - f(x^*)), \quad j = 1, \dots, 2n\}$.

Lemma 3. $D \subseteq R(D)$.

Lemma 4. Assume that an approximation set $\{y_1, \dots, y_{2n}\}$ is constructed at a point $z \in D$ by (9). Then $R(D) \subseteq T(z)$ if and only if $\theta_{2n} \leq 0$.

Let define a function $sign(t)$ as

$$sign(t) = \begin{cases} 1 & \text{if } t \geq 0 \\ -1 & \text{if } t < 0 \end{cases}$$

Lemma 5. If $\beta_j = sign(\gamma_j - \gamma_{n+j})max\{\gamma_j, \gamma_{n+j}\}$ for $j = 1, \dots, n$ and $x' = x^* + \sum_{j=1}^n \beta_j l^j$, then x' is a solution to the problem

$$\max_{x \in R(D)} [\langle Cx, x \rangle + \langle d, x \rangle]. \quad (11)$$

2. Algorithm

Step 1 Find an unconstrained minimum x^* and C -orthonormal directions l^1, \dots, l^n using the conjugate directions method.

Step 2 Solve the problems

$$\max_{x \in D} \langle Cl^j, x \rangle \text{ and } \max_{x \in D} \langle -Cl^j, x \rangle$$

for $j = 1, \dots, n$ to obtain the solutions u^j for $j = 1, 2, \dots, 2n$, i.e.

$$\langle \nabla f(y^j), u^j \rangle = \max_{x \in D} \langle \nabla f(y^j), x \rangle.$$

Step 3 Choose a point $z \in D$ such that $f(z) \neq f(x^*)$.

Step 4 Construct an approximation set A_z^{2n} at the point z by (9).

Step 5 Find a number $i \in \{1, 2, \dots, 2n\}$ such that

$$\begin{aligned} \theta_{2n} &= 2\alpha(\langle Cl^i, u^i - x^* \rangle - \alpha) \\ &= 2\alpha \max\left\{ \max_{1 \leq j \leq n} \langle Cl^j, u^j - x^* \rangle - \alpha; \max_{1 \leq j \leq n} -(\langle Cl^{n+j}, u^{n+j} - x^* \rangle - \alpha) \right\}, \end{aligned}$$

where $\alpha = \sqrt{f(z) - f(x^*)}$

Step 6 Find a local maximizer employing the conditional gradient method starting with an initial approximation point u^i and denote it by z .

Step 7 z is an ϵ global maximizer, where $\epsilon = f(x') - f(z)$ and x' is a solution to the problem (3), i.e.

$$x' = x^* + \sum_{j=1}^n \beta_j l^j$$

$$\beta_j = \text{sign}(\gamma_j - \gamma_{n+j}) \max\{\gamma_j, \gamma_{n+j}\} \text{ for } j = 1, \dots, n$$

■

Theorem 2. Assume that the approximation set $\{y_1, \dots, y_{2n}\}$ at a point $z \in D$ is constructed by (9). If $\theta_{2n} \leq 0$ then z is an approximate solution to Problem (3) and an approximation error δ satisfies the following inequality

$$\delta \leq f(x') - f(z), \quad (12)$$

where x' is a solution to Problem (3).

Proof. Let x^M is a solution to problem (3).

$$f(x^M) = \max_{x \in D} f(x).$$

From Lemma 3, it follows

$$f(x^M) \leq f(x').$$

Therefore,

$$\delta = f(x^M) - f(z) \leq f(x') - f(z).$$

This completes the proof. ■

References

- [1] Enkhbat, R. and Ibaraki, T., Global Optimization Algorithms for General Quadratic Programming, *Journal of the Mongolian Mathematical Society*, 5, p.22-57, 2001.
- [2] Strekalovsky, A.S., On the Global Extremum Problem, *Soviet Math.Doklady*, 292(5), pp.1062-1066, 1987.
- [3] Strekalovsky, A.S., Elements of Nonconvex Optimization, Nauka, Novosibirsk, 2003 (in Russian)
- [4] Strekalovsky, A.S., Global Optimality Conditions for Nonconvex Optimization, *Journal of Global Optimization*, 21(4), pp.415-434, 1998.

COMPUTATIONAL EXPERIMENTS WITH PBC: PARALLEL BRANCH AND CUT LIBRARY

Igor Vasil'ev¹ and Pasquale Avella

Institute of System Dynamics and Control Theory SB RAS

e-mail: vil@icc.ru

RCOST - Research Center on Software Technology, Università del Sannio, Italy

e-mail: avella@unisannio.it

Abstract. In this paper we report on our first computational experiments with a parallel branch and cut (PBC) library. PBC is designed to take advantage of the best sequential B&C solvers. Thus we use these sequential solvers building upon them a parallel layer, which manages their workload. Our preliminary experiments show promising results on small and moderate MIPLIB instances using up to 14 parallel processes.

Key words: combinatorial optimization, mixed integer linear programming, parallel branch and cut algorithm.

Introduction

A wide range of practical problems can be modeled in a form of combinatorial optimization problems. Actually, in many problems, activities and resources are indivisible. Another kind of problems (e.g. scheduling) is defined using logical description which can be transformed to linear constraint description with discrete variables. Combinatorial optimization problems, formulated by continuous and integer variables with linear objective function and constraints, are known as mixed integer linear programming (MILP) problems.

Many of these problems have small and easy formulations, but they are still unsolved and they are always a challenge to deal with. In 1991 an invention of branch and cut (B&C) algorithm was the most significant breakthrough [8]. B&C algorithm is an effective exact method to solve MILP problems. It combines the enumeration approach of branch and bound (B&B) algorithm and the polyhedral approach of cutting planes. These methodological developments together with the tremendous hardware advance made possible a great progress in solving specially structured large-scale combinatorial optimization problems. For example, in 80-th it

¹Support for this author was provided by NATO grant CBP.NR.RIG.911258.

was a big success to solve a traveling salesman problem (TSP) with 100 cities. Today, [2] solve TSP instances with about 25000 cities, just imagine, it contains more than 600 million variables!

This progress became possible also due to the algorithm parallelization. There is a lot of parallel implementations of B&B algorithm. Less number is known for the B&C and B&C&P algorithms. We would like to mention some of them: PARINO [13] and FATCOP [3] are general MILP solvers; PICO [5] is a framework for general B&C algorithms; SYMPHONY [10], COIN/PBC [11], ALPS [12] are frameworks for B&C&P algorithm; CONCORDE [1] is a package to solve TSP.

1. Parallel branch and cut algorithm

In general form, the MILP problem is formulated as follows:

$$\min c^T x \tag{1}$$

$$Ax \geq b \tag{2}$$

$$x_i \text{ is integer } i \in I \tag{3}$$

where $c \in \mathbb{R}^n$, $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$, $I \subseteq \{1, \dots, n\}$ are given.

Problem (1)-(2) is called a *lp relaxation*. Its optimal value provide us with a lower bound of problem (1)-(3). The objective value of any feasible solution of (1)-(3) provides an upper bound of the problem.

The B&B uses a divide and conquer strategy: it partitions the solution space into *subproblems* and then optimizes over each subproblem individually. In B&B algorithm we maintain a list of candidate subproblems which is initialized by placing original problem (1)-(3) in it. We take an entry from this list and solve its lp relaxation. There are four possible results:

1. The relaxed subproblem is infeasible, so the corresponding integer subproblem is infeasible as well, and we *fathom (prune)* it, i.e. we discard it.

2. The lower bound of problem is not less than the best upper bound found so far. It means that the corresponding subproblem does not contain a better solution. Therefore we can prune the subproblem as well.

3. The lower bound is less than the best upper bound and the solution of lp relaxation is integer feasible. It means that the integer subproblem is solved. We can update the best solution and upper bound, and prune this subproblem.

4. The lower bound is less than the best upper bound and the solution of lp relaxation is integer infeasible. In this case we *branch*, i.e. the current subproblem is split into subproblems

on smaller domains and they are added to the candidate list.

This procedure results in a *search tree*. The root of the tree is the original MILP problem. Each node of the tree corresponds to a subproblem. The tree is explored until the candidate set is empty (all the nodes are pruned).

The B&C algorithm is an implementation of B&B algorithm, where special *valid inequalities (cutting planes)* are added dynamically in order to cut the current solution of lp relaxation straitening formulation.

The simple B&B algorithm has a natural parallel structure and it can be easily parallelized. In its more complicated variants, such as B&C, large amount of data must be shared between the parallel processes resulting in increasing parallel overhead and unbalancing of workload.

There is a standard master-slave parallelization model, where the master (tree manager) manages the candidate list and sends the nodes to the slave, which solves the lp relaxation of the corresponding subproblem. This model is implemented in SYMPHONY of [10], COIN/BCP of [11]. Another parallelization model is based on an idea of decentralized and asynchronous subproblem separation. So, we have some parallel workers, which are sequential solvers. They have to divide in some manner the search tree into subtrees and solve them individually. This model is used in PICO [5], ALPS [12].

In B&C a *node selection* strategy plays a very important role. This strategy prescribes the order in which to take subproblems from the candidate list. The most common strategies are the Best first and Depth first strategies. Best first strategy consists of selecting a node with the smallest lower bound. Usually it is one of the nodes at the smallest depth in the search tree. The advantage of this strategy is that it minimizes the size of the search tree. But to follow this strategy it is necessary to keep the large candidate list containing nodes from different branches of search tree. It can be memory consuming and not easy to maintain especially in parallel case.

Depth first strategy consists of selecting a node at the maximum depth (the last processed node). The advantage of this strategy is that the candidate list is small and "local" in sense that all the nodes of the candidate list are on the same branch of the search tree. Also it makes it possible to find quickly a feasible solution which hopefully optimal would be an optimal one. But if it does not happen, the search tree can turned out to be very large, much larger than by using Best first strategy.

In our library we divide the node selection into the layers: parallel and sequential ones. On the sequential layer any solver uses the node selection to process the node by itself. If another solver asks a subproblem, the solver uses the node selection of parallel layer

to send a subproblem to the other solver. It is crucial to have the right choice of node selection strategies on different layers than to get advantage of algorithm parallelization. The main rule is the following: send over only a “hard” subproblems, i.e. the ones for which sending time is negligibly small compared to the time needed to solve this subproblem; solve “easy” subproblems “locally”. It is easy to see that a hard subproblem is a subproblem with the best bound. An easy subproblem is a subproblem with the biggest depth. So, it immediately follows that the node selection of parallel layer must be Best first and the node selection of sequential layer must be Depth first. In fact, only these strategies show good results in our computation experiments on small and moderate MILP instances.

Computational results

The implementation of the library is still in progress. To test our ideas presented in this work, we implemented a prototype – a generic MIP solver. We have made some computational experiments on the following configurations:

- 15 workstations with Pentium Xeon 2.8 Ghz CPU and 1Gb of RAM.
- 100 Mbps local area network.
- Message Passing Interface 1.1 standard implementation MPICH 1.2.4.
- Sequential MIP solvers are Cplex 8.1. The Cplex cutting planes are used at the root node and the automated presolving is switched off.

We have considered some small and moderate instances from MIPLIB [6]. The details of instances are presented in table 1, where column *name* gives the instance names, $\#cols$ – the number of variables, $\#ins$ – the number of integer variables, $\#rows$ – number of constraints, $\#nonzeros$ – the number of nonzero elements in the constraint matrices.

Tables 2-5 illustrate the computational results. Column $\#solvers$ shows the number of parallel solvers. The maximal number is limited with the Cplex licenses we have. Column $\#supprobs$ shows the number of subproblems which were sent over between the solvers, $\#nodes$ – the search tree size, *wall time* – the total wall clock time, and we show speed up and parallel efficiency in columns *speedup* and *effic*. In sequential case ($\#solvers = 1$), the node selection strategy was chosen to contain the fastest result. In parallel case, the node selection strategy was chosen by the following: we have Depth first on the sequential layer and Best first strategy on the parallel layer.

You can see that the first instance *bell3a* is small and easy to deal with. Thus, we were able to achieve linear speedup only with up to 8-9 solvers, after which the time was not

Таблица 1: Instance details

<i>name</i>	<i>#cols</i>	<i>#ints</i>	<i>#rows</i>	<i>#nonzeros</i>
bell3a	133	71	123	347
cap6000	6000	6000	2176	48249
noswot	128	100	182	735
fast0507	63009	63009	507	409249

Таблица 2: bell3a

<i>#solvers</i>	<i>#supbrobs</i>	<i>#nodes</i>	<i>wall time</i>	<i>speedup</i>	<i>effic</i>
1	1.00	32476.00	17.09	1.00	100%
2	9.33	33511.33	7.38	2.32	116%
4	40.67	35430.33	3.64	4.69	117%
8	82.67	34900.67	2.37	7.21	90%
14	124.33	34325.67	2.39	7.16	51%

improved, because the instance is easy and increasing the number of solvers leads to increasing the redundant parallelization work. The next instance is cap6000. It is larger and harder a bit. So, the speed is more close to linear.

Very interesting results were obtained for instance *mas76*. It is small, but the enumeration tree is very big. In this case we have so called acceleration anomaly. The speedup is much better than linear, because the more workers we have, the faster an optimal solution is found and the size of search tree is substantially smaller. The last instance which we want to introduce is *fast0507*. It is quite large and complected. Even the solving of lp relaxation at the root node is time consuming. So, the ramp up time is quite big. Our library prototype is not ready to deal seriously with such problems, but anyway, if we run the algorithm with known optimal solution and strong branching variable selection strategy, the speed up is close to linear.

Таблица 3: cap60000

<i>#solvers</i>	<i>#supbrobs</i>	<i>#nodes</i>	<i>wall time</i>	<i>speedup</i>	<i>effic</i>
1	1.00	33693.00	144.59	1.00	100%
2	7.00	29978.67	68.82	2.10	105%
4	26.67	30123.33	37.48	3.86	96%
8	55.33	24338.67	21.15	6.84	85%
14	80.33	17020.00	13.85	10.44	75%

Таблица 4: noswot

$\#solvers$	$\#supbrobs$	$\#nodes$	<i>wall time</i>	<i>speedup</i>	<i>effic</i>
1	1.00	23025893.00	7373.58	1.00	100%
2	14.67	10861948.67	1866.75	3.95	197%
4	131.67	8757919.33	718.23	10.27	257%
8	579.33	10934671.67	448.72	16.43	205%
14	1442.33	11222512.33	269.19	27.39	196%

Таблица 5: fast0507

$\#solvers$	$\#supbrobs$	$\#nodes$	<i>wall time</i>	<i>speedup</i>	<i>effic</i>
1	1.00	1425.00	3178.61	1.00	100%
2	8.33	1365.00	1821.82	1.74	87%
4	31.67	1292.33	922.35	3.45	86%
8	75.00	1284.00	447.40	7.10	89%
14	106.33	1299.33	291.57	10.90	78%

References

- [1] D. Applegate, R. Bixby, V. Chvátal, and W. Cook. *CONCORDE user documentation*. <http://www.tsp.gatech.edu/concorde/DOC/>, 2003.
- [2] D. Applegate, R. Bixby, V. Chvátal, W. Cook, and K. Helsgaun. *Optimal Tour of Sweden*. <http://www.tsp.gatech.edu/sweden/index.html>, 2004.
- [3] Q. Chen, M. Ferris, and J. Linderoth. Fatcop 2.0: Advanced features in an opportunistic mixed integer programming solver. *Annals of Operations Research*, 103:17–32, 2001.
- [4] *COIN-OR: Common Optimization Interface for Operations Research*. <http://www.coin-or.org>, 2005.
- [5] J. Eckstein, C. A. Phillips, and W. E. Hart. PICO: an object-oriented framework for parallel branch and bound. Technical Report RRR 40-2000, New Brunswick, NJ, 2000.
- [6] A. Martin, T. Achterberg, and T. Koch. *MIPLIB – Mixed Integer Problem Library*. <http://miplib.zib.de/>, 2003.
- [7] G.L. Nemhauser and L.A. Wolsey. *Integer and Combinatorial Optimization*. Wiley, New York, 1988.

- [8] M. Padberg and G. Rinaldi. A branch-and-cut algorithm for the resolution of large-scale traveling salesman problems. *SIAM Review*, 33:60–??, 1991.
- [9] T. Ralphs, L. Ladányi, and M. Saltzman. Parallel branch, cut, and price for large-scale discrete optimization. *Mathematical programming*, to appear, 2004.
- [10] T.K. Ralphs. *SYMPHONY: Single- or Multi-Process Optimization over Networks*. <http://www.branchandcut.org/SYMPHONY>, 2004.
- [11] T.K. Ralphs and L. Ladányi. *COIN/BCP User's manual*. <http://www.coin-or.org>, 2001.
- [12] T.K. Ralphs, L. Ladányi, and M. Saltzman. *ALPS: the abstract library for parallel search*. <http://www.coin-or.org>, 2004.
- [13] M. Savelsbergh, K. Perumalla, J. Linderoth, and U. Ramachandran. *PARINO: A PARallel INteger Optimizer*. <http://www-unix.mcs.anl.gov/~linderot/projects/pio.html#PARINO>, 1997.

О ГЛОБАЛЬНОМ ПОИСКЕ В ЗАДАЧЕ ПОЛИЭДРАЛЬНОЙ ОТДЕЛИМОСТИ

О.В. Дружинина

Институт динамики систем и теории управления, Иркутск
e-mail: strekal@icc.ru

Аннотация. В статье рассматривается задача отделения двух конечных множеств пространства \mathbb{R}^n минимальным числом гиперплоскостей. Производится сведение этой задачи к равносильной невыпуклой негладкой задаче оптимизации. Для решения последней применяется алгоритм глобального поиска, основанный на теории глобального экстремума.

Ключевые слова: д.с., оптимизация, глобальная, безусловная, негладкая, недифференцируемая, не выпуклая, делимость, классификация.

Введение

Широко известно понятие линейной делимости множеств [1], однако на практике часто приходится сталкиваться с более общими понятиями делимости, например полиэдральной делимостью. Практические приложения задачи о полиэдральной делимости, касающиеся диагностики рака груди и предсказания рецидивов, рассмотрены в работе [3].

Далеко не всегда два множества можно отделить линейно, однако в некоторых случаях множества, неотделимые одной гиперплоскостью, можно отделить несколькими гиперплоскостями. Отсюда возникает постановка задачи о полиэдральной делимости заключающаяся в отделении двух конечных множеств точек пространства \mathbb{R}^n минимальным числом гиперплоскостей.

1. Постановка задачи

Определение 1. i) Множества \mathcal{A} и \mathcal{B} полиэдрально делимы семейством гиперплоскостей

$$H_p = H_p(\omega^p, \xi_p) = \{x \in \mathbb{R}^n | \langle \omega^p, x \rangle = \xi_p, \omega^p \in \mathbb{R}^n, \xi_p \in \mathbb{R}\}, p = 1, \dots, P$$

если для каждой точки a_i множества \mathcal{A} и каждой гиперплоскости H_p выполняется неравенство:

$$\langle a_i, \omega^p \rangle \leq \xi_p;$$

и для каждой точки b_j множества \mathcal{B} и хотя бы одной из гиперплоскостей H_p справедливо неравенство:

$$\langle b_j, \omega^p \rangle \geq \xi_p.$$

Из определения 1 видно, что множества \mathcal{A} и \mathcal{B} входят в него несимметрично. С учетом вышесказанного, это определение можно переформулировать следующим образом.

ii) Множества \mathcal{A} и \mathcal{B} полиэдрально делимы семейством гиперплоскостей

$$H_p(v^p, \gamma_p) = \{x \in \mathbb{R}^n | \langle v^p, x \rangle = \gamma_p, v^p \in \mathbb{R}^n, \gamma_p \in \mathbb{R}, p = 1, \dots, P\}$$

если справедливы неравенства

$$\forall i = 1, \dots, M, \forall p = 1, \dots, P, \langle a_i, v^p \rangle \leq \gamma_p - 1; \quad (1)$$

$$\forall j = 1, \dots, N, \exists p \in \{1, \dots, P\}, \langle b_j, v^p \rangle \geq \gamma_p + 1. \quad (2)$$

#

Предложение 1. Множества \mathcal{A} и \mathcal{B} полиэдрально отделимы тогда и только тогда, когда выпуклая оболочка множества \mathcal{A} не пересекается с множеством \mathcal{B} :

$$\text{conv}(\mathcal{A}) \cap (\mathcal{B}) = \emptyset. \quad (3)$$

#

Для точки a^i множества \mathcal{A} выполнение неравенств (1) равносильно одному неравенству:

$$\max_{1 \leq p \leq P} (\langle a^i, v^p \rangle - \gamma_p + 1) \leq 0.$$

Мера нарушения этого неравенства определяется формулой

$$\max\{0, \max_{1 \leq p \leq P} (\langle a^i, v^p \rangle - \gamma_p + 1)\}. \quad (4)$$

Аналогичным образом, для точки b^j множества \mathcal{B} выполнение неравенств (2) равносильно одному неравенству:

$$\min_{1 \leq p \leq P} (-\langle b^j, v^p \rangle + \gamma_p + 1) \leq 0,$$

и мера нарушения этого неравенства есть

$$\max\{0; \min_{1 \leq p \leq P} (-\langle b^j, v^p \rangle + \gamma_p + 1)\}. \quad (5)$$

Определение 2. Общая "ошибка классификации" для всех точек множеств \mathcal{A} и \mathcal{B} принимает следующую форму

$$\Delta = \min_{\{H_p\}} \left[\frac{1}{M} \sum_{i=1}^M \max\{0, \max_{1 \leq p \leq P} (\langle a^i, v^p \rangle - \gamma_p + 1)\} + \frac{1}{N} \sum_{j=1}^N \max\{0; \min_{1 \leq p \leq P} (-\langle b^j, v^p \rangle + \gamma_p + 1)\} \right]. \quad (6)$$

#

Предложение 2. Множества \mathcal{A} и \mathcal{B} полиэдрально отделимы тогда и только тогда, когда

$$\Delta = 0. \quad (7)$$

При этом

1. $v^p = 0$ $p = 1, \dots, P$ не может являться оптимальным.
2. Если найдется непустое подмножество индексов $\emptyset \neq Q = \{p \in P | v^p = 0\}$ то множества \mathcal{A} и \mathcal{B} отделимы $(P \setminus Q)$ гиперплоскостями.

#

Пусть V – матрица размерности $(P \times n)$, столбцами которой являются векторы v^p ; $\Gamma \in \mathbb{R}^P$ – вектор, элементами которого являются γ_p . Функцию ошибки, зависящую от $P \times (n + 1)$ переменных, определим следующим образом:

$$F(V, \Gamma) = F_1(V, \Gamma) + F_2(V, \Gamma), \quad (8)$$

где

$$F_1(V, \Gamma) = \frac{1}{M} \sum_{i=1}^M \max\{0, \max_{1 \leq p \leq P} (\langle a_i, v^p \rangle - \gamma_p + 1)\}, \quad (9)$$

$$F_2(V, \Gamma) = \frac{1}{N} \sum_{j=1}^N \max\{0, \min_{1 \leq p \leq P} (-\langle b_j, v^p \rangle + \gamma_p + 1)\}.$$

Можно показать, что функция ошибки, определяемая формулами (8),(9) является d.c. функцией и ее явное d.c. разложение имеет вид

$$F = g - f,$$

где g и f выпуклые функции, определяемые формулами:

$$g(V, \Gamma) = \frac{1}{M} \sum_{i=1}^M \max\{0, \max_{1 \leq p \leq P} (\langle a_i, v^p \rangle - \gamma_p + 1)\} + \frac{1}{N} \sum_{j=1}^N \max\{0, \max_{1 \leq p \leq P} (\langle b_j, v^p \rangle - \gamma_p - 1)\};$$

$$f(V, \Gamma) = -\frac{1}{N} \sum_{j=1}^N \max_{1 \leq p \leq P} (\langle b_j, v^p \rangle - \gamma_p - 1);$$

Таким образом, задачу полиэдральной отделимости можно свести к задаче безусловной минимизации негладкой невыпуклой функции ошибки $F(V, \Gamma)$. Для её решения применяем теорию и методы глобального поиска, разработанные А.С. Стрекаловским [2], обобщив их для негладкого случая.

2. Локальный поиск

Задача полиэдральной отделимости является невыпуклой. Поэтому даже для нахождения локального решения этой задачи неприменимы стандартные методы выпуклой оптимизации. Для её решения используем специальный метод локального поиска, который будет описан ниже.

Рассмотрим задачу d.c. минимизации

$$F(x) = g(x) - f(x) \downarrow \min, x \in D, \quad (P)$$

где $D \in \mathbb{R}^n$ – выпуклое множество, функции $g(x)$, $f(x)$ – выпуклые. Предположим, что и в любой точке $x \in D$ можем найти некоторый субградиент $x^* \in \partial f(x)$.

Пусть задано некоторое начальное приближение $x^0 \in D$. Далее, если известна допустимая точка $x^s \in D$, то следующую итерацию $x^{s+1} \in D$ будем искать как приближенное решение линеаризованной в точке x^s задачи:

$$g(x) - \langle x_s^*, x \rangle \downarrow \min, x \in D, x_s^* \in \partial f(x^s). \quad (PL_s)$$

Точнее говоря, точка $x^{s+1} \in D$ удовлетворяет условию:

$$g(x^{s+1}) - \langle x_s^*, x^{s+1} \rangle \leq \inf_{x \in D} \{g(x) - \langle x_s^*, x \rangle\} + \delta_s, \quad (10)$$

где последовательность δ_s такова, что

$$\delta_s \geq 0, s = 1, 2, \dots, \sum_{s=0}^{\infty} \delta_s < \infty.$$

Теорема 1. Пусть в задаче (P) целевая функция $F = g - f$ ограничена снизу на допустимом множестве D , и в любой допустимой точке x можно найти некоторый субградиент $x^* \in \partial f(x)$. Тогда последовательность $\{x^s\}$, генерируемая по правилу (10) удовлетворяет условию:

$$\lim_{s \rightarrow \infty} \left\{ \inf_{x \in D} [g(x) - g(x^{s+1}) - \langle x_s^*, x - x^{s+1} \rangle] \right\} = 0, \quad x_s^* \in \partial f(x^s). \quad (11)$$

При этом любая предельная точка z последовательности $\{x^s\}$, удовлетворяет условию

$$\inf_{x \in D} \{g(x) - g(z) - \langle z^*, x - z \rangle\} = 0, \quad z^* \in \partial f(z). \quad (12)$$

Из (12), в случае дифференцируемости функций f и g на открытом множестве $\Omega \subset \mathbb{R}^n$, $D \subset \Omega$, вытекает классическое условие стационарности

$$\langle \nabla g(x_*) - \nabla f(x_*), x - x_* \rangle \geq 0, \quad \forall x \in D.$$

#

При практическом применении вышеописанного метода решающую роль играет выбор метода решения линеаризованной задачи (PL_s) на каждом шаге метода.

В качестве критерия останова метода можно использовать одно из следующих неравенств:

$$\begin{aligned} F(x^s) - F(x^{s+1}) &\leq \tau, \\ g(x^s) - g(x^{s+1}) + \langle x_s^*, x^{s+1} - x^s \rangle &\leq \tau, \quad x_s^* \in \partial f(x^s), \end{aligned}$$

где τ – заданная точность. В этом случае, точка, в которой произошел останов, является τ -критической в задаче (P).

Проведены численные эксперименты по тестированию метода локального поиска на тестовых задачах негладкой д.с. минимизации, а также на специальных задачах о полиэдральной отделимости. Проведена визуализация результатов. По результатам тестов можно сделать вывод, что при хорошем начальном приближении метод локального поиска находит даже точку глобального минимума задачи. Иначе, метод останавливается в некоторой стационарной точке. Таким образом, предлагаемая методика позволяет строить отделяющие гиперплоскости. Однако, пользуясь только методом локального поиска задачу полиэдральной отделимости удастся решить далеко не всегда.

2. Стратегия глобального поиска

В [2] предложена стратегия глобального поиска, модифицированная здесь на негладкий случай. Пусть заданы начальная точка $x_0 \in D$ и числовые последовательности $\{\tau_k\}$ и $\{\delta_k\}$. $\tau_k, \delta_k > 0, k = 0, 1, 2, \dots, \tau_k \downarrow 0, \delta_k \downarrow 0 (k \rightarrow \infty)$.

Шаг 0. Положить $k := 0, \quad x^k := x_0$.

Шаг 1. Начиная с $x^k \in D$ посредством метода локального поиска построить τ_k -критическую точку z^k . Пусть $\zeta_k := F(z^k)$.

Шаг 2. Выбрать некоторое $\beta \in [\beta_-, \beta_+]$. В частности, процесс одномерного поиска по β можно начинать при $\beta_0 = g(z^k)$.

Шаг 3. Построить некоторую аппроксимацию

$$\mathcal{A}_k(\beta) = \{v^1, \dots, v^{N_k} \mid f(v^i) = \beta - \zeta_k, i = 1, \dots, N_k, N_k = N_k(\beta)\}.$$

Шаг 4. Сформировать семейство индексов

$$I_k = I_k(\beta) = \{i \in \{1, \dots, N_k\} \mid g(v^i) \leq \beta\}.$$

Шаг 5. Для каждого $i \in I_k$ найти глобальное δ_k -решение $u^i \in D$ линейризованной задачи:

$$g(u^i) - \langle v_i^*, u^i \rangle - \delta_k \leq \inf_x \{g(x) - \langle v_i^*, x \rangle \mid x \in D\}. \quad (13)$$

где $v_i^* \in \partial f(v^i)$.

Шаг 6. Для каждого $i \in I_k$ найти глобальное δ_k -решение w^i , $f(w^i) = \beta - \zeta_k$, задачи уровня:

$$\langle w_i^*, u^i - w^i \rangle + \delta_k \geq \sup_v \{ \langle v^*, u^i - v \rangle \mid f(v) = \beta - \zeta_k \}, \quad u^* \in \partial f(u). \quad (14)$$

Шаг 7. Положить $\eta_k(\beta) := \eta_k^0(\beta) + \beta$, где

$$\eta_k^0(\beta) := \langle w_j^*, u^j - w^j \rangle - g(u^j) \triangleq \max_{1 \leq i \leq N} \{ \langle w_i^*, u^i - w^i \rangle - g(u^i) \}.$$

Шаг 8. Если $\eta_k(\beta) > 0$, то положить $x^{k+1} := u^j$ и вернуться на Шаг 1.

Шаг 9. Если $\eta_k(\beta) \leq 0$, то положить $\beta := \beta + \Delta\beta \in [\beta_-, \beta_+]$ и идти на Шаг 3. Число $\Delta\beta$ может быть найдено одним из известных методов одномерной максимизации.

Шаг 10. Если $\eta_k(\beta) \leq 0 \forall \beta \in [\beta_-, \beta_+]$ (т.е. одномерный поиск по β завершен), то положить $k := k + 1$, $x^{k+1} := z^k$ и вернуться на Шаг 1.

Сформулирована и доказана теорема сходимости алгоритма глобального поиска для д.с. минимизации для недифференцируемого случая. теоретически обосновано построение разрешающих наборов в задаче д.с. минимизации на основе эквивалентности последней задачи с задачей параметрической выпуклой максимизации.

Для численной реализации алгоритма глобального поиска были конкретизированы все его этапы: построение аппроксимации поверхности уровня, решение линейризованной задачи и задачи уровня, а также одномерный поиск по параметру β . проведено тестирование алгоритма глобального поиска на задачах негладкой д.с. минимизации и задачах о полиэдральной отделимости, экспериментально доказавшие эффективность предложенного подхода.

Список литературы

- [1] Васильев Ф.П. *Численные методы решения экстремальных задач*. – М.: Наука, 1988.
- [2] Стрекаловский А.С. *Элементы невыпуклой оптимизации*. – Новосибирск, Наука, 2003.
- [3] O.L. Mangasarian, W. Nick Street, W.H. Wolberg *Breast cancer diagnosis and prognosis via linear programming* // *Operations Research* 43(4), July-August 1995, p. 570-577.

ON A PROBLEM POLYHEDRAL SEPARABILITY

O.V. Druzhinina

Institute of System Dynamics and Control Theory SB RAS, Irkutsk
e-mail: e-mail: strekal@icc.ru

Abstract. We consider a problem of separating two finite set in \mathbb{R}^n by a minimal number of hyperplanes in this article. we propose a reduction of this problem to an equivalent non-convex and non-smooth optimization problem. In order to solve the last problem one applies the global search theory.

Key words: d.c., optimization, global, unconditional, non-smooth, nodifferential, non-convex, separability, classification.

ON THE MINIMIZATION OF A STRONGLY CONVEX FUNCTION

R.Enkhbat

National University of Mongolia

e-mail: renkhbat@ses.edu.mn

Abstract. This paper is devoted to the problem of minimizing a strongly convex function over an arbitrary set. Based on global optimality conditions for this problem, we develop an algorithm for solving it. The proposed algorithms are shown to be globally convergent.

Key words: convex minimization, optimality condition, algorithm, convergence.

Introduction

In this paper we consider the problem of minimizing a strongly convex function over an arbitrary set (which may be nonconvex) $D \subset R^n$:

$$f(x) \longrightarrow \min, \quad x \in D. \quad (1)$$

Problem (1) is important in nonlinear programming and global optimization from the view of theory and practice. There are a number of well established algorithms; e.g., [1, 3, 4, 5, 7, 8] devoted to theoretical and computational aspects of minimizing a convex function on a convex set, but it seems to us that there is not enough literature [2, 6] on theory and algorithms for solving problem (1) globally over a nonconvex feasible set D . The main purpose of this paper is to develop a new algorithm with global convergence for solving the above problem with arbitrary feasible regions D .

1. Global Minimality Condition

We consider the convex minimization problem:

$$f(x) \longrightarrow \min, \quad x \in D, \quad (2)$$

where $f : R^n \rightarrow R$ is strongly convex and continuously differentiable, and D is an arbitrary compact set in R^n . We can obtain the following global optimality condition as follows:

Theorem 1 Let z be a solution of problem (2). Then

$$\langle f'(x), x - y \rangle \geq 0 \text{ for all } y \in E_{f(z)}(f) \text{ and } x \in D. \quad (3)$$

If, in addition,

$$\min_{x \in D} \|f'(x)\| > 0 \quad (4)$$

holds, then condition (3) is also sufficient.

Proof. Necessity. Assume that z is a solution of problem (2). Consider $x \in D$ and $y \in E_{f(z)}(f)$. Then by the convexity of f , we have

$$0 \leq f(x) - f(z) = f(x) - f(y) \leq \langle f'(x), x - y \rangle.$$

Sufficiency. Let us prove the assertion by contradiction. Assume that (3) holds for a point $z \in D$ and there exists a point $u \in D$ such that

$$f(u) < f(z).$$

We also assume that (4). Now define u_α as follows for $\alpha > 0$:

$$u_\alpha = u + \alpha f'(u).$$

Then, by the convexity of f , we have

$$f(u_\alpha) - f(u) \geq \langle f'(u), u_\alpha - u \rangle = \alpha \|f'(u)\|^2,$$

which implies by (4) that

$$f(u_\alpha) \geq f(u) + \alpha \|f'(u)\|^2 > f(u).$$

Then find $\alpha = \bar{\alpha}$ such that

$$f(u) + \bar{\alpha} \|f'(u)\|^2 = f(z),$$

that is,

$$\bar{\alpha} = \frac{f(z) - f(u)}{\|f'(u)\|^2} > 0.$$

Thus we get

$$f(u_{\bar{\alpha}}) \geq f(u) + \bar{\alpha} \|f'(u)\|^2 = f(z) > f(u).$$

Define a function $h : R^+ \rightarrow R$ as

$$h(\alpha) = f(u + \alpha f'(u)) - f(z),$$

where $R^+ = \{\alpha \in R \mid \alpha \geq 0\}$. It is clear that h is continuous on $[0, +\infty)$. Note that $h(\bar{\alpha}) \geq 0$ and $h(0) < 0$. There are two cases with respect to the values of $h(\bar{\alpha})$ which we should consider.

Case a: $h(\bar{\alpha}) = 0$ (or $f(u + \bar{\alpha} f'(u)) = f(z)$). Then

$$\langle f'(u), u - u_{\bar{\alpha}} \rangle = -\langle f'(u), \bar{\alpha} f'(u) \rangle = -\bar{\alpha} \|f'(u)\|^2 < 0,$$

contradicting condition (3).

Case b: $h(\bar{\alpha}) > 0$ and $h(0) < 0$. Since h is continuous, there exists a point $\alpha_o \in (0, \bar{\alpha})$ such that $h(\alpha_o) = 0$ (or $f(u + \alpha_o f'(u)) = f(z)$). Then we have

$$\langle f'(u), u - u_{\alpha_o} \rangle = -\alpha_o \|f'(u)\|^2 < 0,$$

again contradicting (3).

Thus, in both cases, we find contradictions, proving the theorem. ■

2. Minimization Algorithm and its Global Convergence

For the purpose of describing an algorithm for solving (2), we define auxiliary function $p(y)$:

$$p(y) = \min_{x \in D} \langle f'(x), x - y \rangle, \quad y \in R^n.$$

Lemma 1 Function $p(y)$ is continuous on R^n .

Proof. Note that $\|f'(x)\|$ is continuous on R^n because $f(x)$ is assumed to be continuously differentiable. Let M denote the global maximum of $\|f'(x)\|$ on D :

$$M = \max_{x \in D} \|f'(x)\|.$$

Let $y \in R^n$ be an arbitrary point and $\Delta y \in R^n$ be a small increment at y . Then we have

$$\begin{aligned} |\Delta p(y)| &= |p(y + \Delta y) - p(y)| \\ &= \left| \min_{x \in D} \langle f'(x), x - (y + \Delta y) \rangle - \min_{x \in D} \langle f'(x), x - y \rangle \right| \\ &\leq \max_{x \in D} |\langle f'(x), x - (y + \Delta y) \rangle - \langle f'(x), x - y \rangle| \\ &= \max_{x \in D} |\langle f'(x), -\Delta y \rangle| \leq M \|\Delta y\|. \end{aligned}$$

Consequently, we have

$$\lim_{\Delta y \rightarrow 0} \Delta p(y) = 0.$$

which completes the proof. ■

Now we reformulate Theorem 1 in terms of function $\psi(z)$ defined as follows.

$$\psi(z) = \min_{y \in E_{f(z)}(f)} p(y), \quad z \in D.$$

Theorem 2 Assume that $f : R^n \rightarrow R$ is strongly convex and continuously differentiable, D is a compact set in R^n and $\min_{x \in D} \|f'(x)\| > 0$ holds. If $\psi(z) = 0$ holds for a point $z \in D$, then z is a solution to problem (2).

Proof. Obviously $\psi(z) = 0$ implies

$$0 = \psi(z) \leq p(y) \leq \langle f'(x), x - y \rangle,$$

for all $y \in E_{f(z)}(f)$ and $x \in D$, which shows that z is a solution to problem (2) by Theorem 1. ■

Now we are ready to present an algorithm for solving problem (2). We also suppose that one can solve the problem of computing $p(y) = \min_{x \in D} \langle f'(x), x - y \rangle$ for any given $y \in R^n$.

Algorithm MIN

Input: A strongly convex and continuously differentiable function f , and a compact set D .

Output: A solution x to the minimization problem (2).

Step 1. Choose a feasible point $x^0 \in D$. Set $k := 0$.

Step 2. Solve the following problem

$$p(y) \longrightarrow \min, \quad y \in E_{f(x^k)}(f).$$

Let y^k be a solution of this problem (i.e., $p(y^k) = \min_{x \in D} \langle f'(x), x - y^k \rangle = \min_{y \in E_{f(x^k)}(f)} p(y)$), and let x^{k+1} realizes $p(y^k)$ (i.e., $\psi(x^k) = p(y^k) = \langle f'(x^{k+1}), x^{k+1} - y^k \rangle$).

Step 3. If $\psi(x^k) = 0$ then output $x = x^k$ and terminate. Otherwise, let $k := k + 1$ and return Step 2. ■

Theorem 3 Assume that $f : R^n \rightarrow R$ is strongly convex and continuously differentiable, D is a compact set in R^n , and $\min_{x \in D} \|f'(x)\| > 0$ holds. Then the sequence $\{x^k, k = 0, 1, \dots\}$ generated by Algorithm MIN is a minimizing sequence for problem (2), that is,

$$\lim_{k \rightarrow \infty} f(x^k) = \min_{x \in D} f(x),$$

and an accumulation point of the sequence is a global minimizer of (2).

Proof. From the construction of $\{x^k\}$, we have $x^k \in D$ and $f(x^k) \geq f_*$ for all k , where $f_* = f(x_*) = \min_{x \in D} f(x)$. Also, note that, for all $y \in E_{f(x^k)}(f)$ and $x \in D$, we have

$$\psi(x^k) = \min_{y \in E_{f(x^k)}(f)} \min_{x \in D} \langle f'(x), x - y \rangle \leq \langle f'(x), x - y \rangle \leq 0.$$

If there exists a k such that $\psi(x^k) = 0$ then, by theorem 2, x^k is a solution to problem (3.1) and the proof is complete. Therefore, we assume $\psi(x^k) < 0$ for all k , and prove the theorem by contradiction. If $\{x^k\}$ is not a minimizing sequence for problem (2), the following inequality holds.

$$\lim_{k \rightarrow \infty} \inf f(x^k) > f_*. \quad (5)$$

By Algorithm MIN, we have

$$p(y^k) = \psi(x^k) = \min_{y \in E_{f(x^k)}(f)} \min_{x \in D} \langle f'(x), x - y \rangle = \langle f'(x^{k+1}), x^{k+1} - y^k \rangle$$

and $f(y^k) = f(x^k)$. The convexity of f implies that

$$f(x^k) - f(x^{k+1}) = f(y^k) - f(x^{k+1}) \geq \langle f'(x^{k+1}), y^k - x^{k+1} \rangle = -\psi(x^k) > 0. \quad (6)$$

Hence, we obtain $f(x^{k+1}) < f(x^k)$ for all k , and the sequence $\{f(x^k)\}$ is strictly decreasing. Since the sequence is bounded from below by f_* , it has a limit and satisfies

$$\lim_{k \rightarrow \infty} (f(x^{k+1}) - f(x^k)) = 0. \quad (7)$$

Then, from (6) and (7), we obtain

$$\lim_{k \rightarrow \infty} \psi(x^k) = 0.$$

Now define v_α as follows:

$$v_\alpha = x_* + \alpha f'(x_*), \quad \alpha > 0.$$

Then, by the convexity of f , we have

$$f(v_\alpha) - f(x_*) \geq \langle f'(x_*), v_\alpha - x_* \rangle = \alpha \|f'(x_*)\|^2 > 0,$$

which implies

$$f(v_\alpha) \geq f(x_*) + \alpha \|f'(x_*)\|^2 > f(x_*),$$

for all $\alpha > 0$. Choose $\alpha = \alpha_k$ such that

$$f(x_*) + \alpha_k \|f'(x_*)\|^2 > f(x_k),$$

that is,

$$\alpha_k > \frac{f(x^k) - f(x_*)}{\|f'(x_*)\|^2} > 0.$$

Define a function $h_k : R^+ \rightarrow R$ as

$$h_k(\alpha) = f(x_* + \alpha f'(x_*)) - f(x^k),$$

where $R^+ = \{\alpha \in R \mid \alpha \geq 0\}$. It is clear that h_k is continuous on $[0, +\infty)$. Note that $h_k(\alpha_k) > 0$ and $h_k(0) < 0$ hold. Since h_k is continuous, there exists a point $\bar{\alpha}_k \in (0, \alpha_k)$ such that $h_k(\bar{\alpha}_k) = 0$, that is $f(v_{\bar{\alpha}_k}) = f(x^k)$ and $v_{\bar{\alpha}_k} = x_* + \bar{\alpha}_k f'(x_*)$. Also, note that

$$\psi(x^k) = \min_{y \in E_{f(x^k)}(f)} \min_{x \in D} \langle f'(x), x - y \rangle \leq \langle f'(x_*), x_* - v_{\bar{\alpha}_k} \rangle.$$

Taking into account $v_{\bar{\alpha}_k} = x_* + \bar{\alpha}_k f'(x_*)$, we have

$$-\psi(x^k) \geq \langle f'(x_*), v_{\bar{\alpha}_k} - x_* \rangle = \|f'(x_*)\| \|v_{\bar{\alpha}_k} - x_*\| \geq \min_{x \in D} \|f'(x)\| \|v_{\bar{\alpha}_k} - x_*\| > 0.$$

Since $\lim_{k \rightarrow \infty} \psi(x^k) = 0$, this implies

$$\lim_{k \rightarrow \infty} v_{\bar{\alpha}_k} = x_*.$$

Then the continuity of f on R^n yields

$$\lim_{k \rightarrow \infty} f(x^k) = \lim_{k \rightarrow \infty} f(v_{\bar{\alpha}_k}) = f(x_*), \quad (8)$$

which is a contradiction to (5).

Consequently, $\{x^k\}$ is a minimizing sequence for problem (2). Since D is compact, we can always select a convergent subsequence $\{x^{k_l}\}$ from $\{x^k\}$ such that

$$\lim_{l \rightarrow \infty} x^{k_l} = \bar{x} \in D.$$

Then together with (8), we obtain

$$\lim_{l \rightarrow \infty} f(x^{k_l}) = f(\bar{x}) = f_*,$$

which completes the proof. ■

References

- [1] Bertsekas, D.P., Constrained Optimization and Lagrangean Multiplier Methods, *Academic Press*, New York, 1982.
- [2] Helmut Dietrich., Global Optimization Conditions for Certain Nonconvex Minimization Problems, *Journal of Global Optimization*, 5, pp.359-370, 1994.
- [3] Jacoby, S.L., Kowalik, J.S. and Pizzo, J.T., Iterative Methods for Nonlinear Optimization Problems, *Prentice-Hall, Inc.*, 1972.
- [4] Karmanov, V.G., Mathematical Programming, *Mir Publisher*, Moscow, 1989.

- [5] Podinovskii, V.V. and Timohov, A.V., Introduction to Mathematical Programming, *Moscow University*, 1988.
- [6] Rockafellar, R.T., Convex Analysis, *Princeton University Press*, Princeton, 1970.
- [7] Vasiliev, F.P., Numerical Methods for Solving Extremum Problems, *Nauka*, Moscow, 1987.
- [8] Vasiliev, O.V., Optimization Methods, *World Federation Publishers*, Atlanta, 1996.

О ЛОКАЛЬНОМ И ГЛОБАЛЬНОМ ПОИСКЕ В ЗАДАЧАХ БИЛИНЕЙНОГО ПРОГРАММИРОВАНИЯ

А.В. Орлов

*Институт динамики систем и теории управления СО РАН, Иркутск
e-mail: anor@icc.ru*

Аннотация. В статье предлагается подход к решению билинейных задач математического программирования с несвязанными переменными, который базируется на стратегии глобального поиска в задачах d.c. максимизации, разработанной А.С. Стрекаловским. Представлены два метода локального поиска в этих задачах и приведена схема глобального поиска.

Ключевые слова: билинейное программирование, d.c. представление целевой функции, локальный поиск, глобальный поиск.

Введение

Ряд задач теории принятия решений, исследования операций и экономики могут быть представлены в виде задач билинейного программирования. Например, так могут быть сформулированы задача поиска ситуации равновесия по Нэшу в биматричной игре [3, 4] и задача билинейной отделимости двух множеств в пространстве \mathbb{R}^n [5].

Обычно в билинейных задачах переменные обозначаются через x и y . Целевая функция линейна по каждой из этих переменных при фиксированной другой, но при этом, в ней обязательно присутствуют члены, содержащие произведения $x_i y_j$, возможно, с некоторым коэффициентом, которые порождают невыпуклость в данной задаче. Как известно, невыпуклые задачи обладают большим количеством локальных решений и стационарных точек, далеких от глобального оптимума даже по значению целевой функции, а классические методы выпуклой оптимизации в подобных задачах неэффективны [1].

Можно выделить два типа задач билинейного программирования: со связанными переменными и с несвязанными переменными. В первом случае ограничения на переменные задаются совместно, во втором — каждая из двух переменных имеет свои ограничения. Задача со связанными переменными представляется более трудной чем задача с несвязанными. Первые исследования, результаты которых представлены в данной работе, посвящены более простой задаче.

За более чем четыре десятилетия исследования задачи с несвязанными переменными различными авторами было предложено несколько методов ее решения (см. обзор в [6]), однако они демонстрируют сильный рост времени решения при возрастании размерности, либо содержат плохо алгоритмизуемые операции (см., например, [3]). Поэтому для поиска глобального решения в этой задаче предлагается использовать разрабатываемый в нашей группе подход, основанный на условиях глобальной оптимальности [2]. Суть этого подхода заключается в декомпозиции сложной невыпуклой задачи на последовательность более простых.

1. Постановка задачи и d.c. представление целевой функции

В общем виде целевая функция задачи билинейного программирования записывается

следующим образом:

$$F(x, y) = \langle c, x \rangle + \langle x, Cy \rangle + \langle d, y \rangle, \quad (1)$$

где $c, x \in \mathbb{R}^m$; $d, y \in \mathbb{R}^n$; C — матрица размерности $(m \times n)$.

Чаще всего в литературе рассматривается задача максимизации функции $F(\cdot)$ на многогранном множестве. Задача с несвязанными переменными на таком множестве записывается следующим образом:

$$\left. \begin{array}{l} F(x, y) \uparrow \max_{(x, y)}, \\ x \in X \triangleq \{x \in \mathbb{R}^m \mid Ax \leq a, x \geq 0\}, \\ y \in Y \triangleq \{y \in \mathbb{R}^n \mid By \leq b, y \geq 0\}. \end{array} \right\} \quad (BLP)$$

Здесь A — матрица размерности $(m_1 \times m)$, B — матрица размерности $(n_1 \times n)$; $a \in \mathbb{R}^{m_1}$, $b \in \mathbb{R}^{n_1}$. Всюду ниже будем предполагать, что X и Y являются непустыми и ограниченными множествами.

Легко видеть, что целевая функция задачи (BLP) может быть представлена в виде разности двух выпуклых функций

$$F(x, y) = f(x, y) - g(x, y), \quad (2)$$

где $f(x, y) = \frac{1}{4}\|x + Cy\|^2 + \langle c, x \rangle$, $g(x, y) = \frac{1}{4}\|x - Cy\|^2 - \langle d, y \rangle$.

Поэтому для решения задачи (BLP) применима стратегия глобального поиска для задач д.с. программирования из [2].

2. Локальный поиск

Одним из основных модулей упомянутой стратегии глобального поиска является поиск критических (стационарных) точек или локальных решений в рассматриваемой невыпуклой задаче. При этом понятие критической точки зависит от структуры конкретной задачи и соответствующего метода локального поиска. Ниже предлагается два специальных метода локального поиска в задаче (BLP) .

Пусть задана точка (x_0, y_0) , начиная с которой требуется производить локальный поиск в задаче (BLP) .

X-процедура.

Шаг 0. Положить $s := 0$, $x^s := x_0$.

Шаг 1. Некоторым методом линейного программирования найти $\rho_s/2$ -решение y^{s+1} следующей задачи:

$$\langle d + x^s C, y \rangle \uparrow \max_y, \quad y \in Y, \quad (LP_y)$$

так что справедливым будет неравенство

$$\langle d + x^s C, y^{s+1} \rangle + \rho_s/2 \geq \sup_y \{ \langle d + x^s C, y \rangle \mid y \in Y \}. \quad (3)$$

Шаг 2. Найти $\rho_s/2$ -решение x^{s+1} задачи линейного программирования по другой переменной

$$\langle c + Cy^{s+1}, x \rangle \uparrow \max_x, \quad x \in X, \quad (LP_x)$$

так что справедливым будет неравенство

$$\langle c + Cy^{s+1}, x^{s+1} \rangle + \rho_s/2 \geq \sup_x \{ \langle c + Cy^{s+1}, x \rangle \mid x \in X \}. \quad (4)$$

Шаг 3. Если

$$F(x^{s+1}, y^{s+1}) - F(x^s, y^{s+1}) \leq \tau, \quad (5)$$

где τ — заданная точность, то STOP. Иначе положить $s := s + 1$ и перейти на шаг 1. #

Заметим, что метод фактически состоит в поочередном решении задач линейного программирования то по одной, то по другой переменной, при этом учитывается возможность приближенного решения этих вспомогательных задач. Данный метод является модификацией алгоритма локального поиска для задач билинейного программирования из [3].

Можно доказать следующую теорему сходимости данного метода.

Теорема 1. Пусть функция $F(\cdot)$ ограничена сверху на множестве $X \times Y$. Тогда последовательность пар (x^s, y^s) , генерируемая X-процедурой при условии: $\rho_s > 0$, $s = 1, 2, \dots$ и $\sum_{s=1}^{\infty} \rho_s < +\infty$, сходится к точке (\hat{x}, \hat{y}) , удовлетворяющей условиям:

$$F(\hat{x}, \hat{y}) \geq F(\hat{x}, y) \quad \forall y \in Y, \quad (6)$$

$$F(\hat{x}, \hat{y}) \geq F(x, \hat{y}) \quad \forall x \in X. \quad (7)$$

#

Точку (\hat{x}, \hat{y}) , удовлетворяющую условиям (6) и (7), будем называть критической точкой задачи (BLP). Эта точка является глобальным решением задачи (BLP) по каждой из переменных x и y в отдельности (но не обязательно вместе!). При останове алгоритма на шаге 3 (т.е. при выполнении неравенства (5)), мы получаем приближенную критическую точку задачи (BLP).

Отметим особенности этого метода локального поиска. Во-первых, для начала работы алгоритма не требуется целиком точка (x_0, y_0) , достаточно только ее части x_0 , а во-вторых, эта точка может быть вообще недопустимой. Несмотря на это, удается доказать сходимость предложенного метода к точке, удовлетворяющей неравенствам (6) и (7).

Для осуществления локального поиска в задаче (BLP) можно предложить и другой метод, "симметричный" вышеописанному. В нем для начала работы алгоритма из стартовой точки используется не x_0 , а y_0 , и вспомогательные задачи линейного программирования решаются в другом порядке.

Y-процедура.

Шаг 0. Положить $s := 0$, $y^s := y_0$.

Шаг 1. Некоторым методом линейного программирования найти $\rho_s/2$ -решение x^{s+1} следующей задачи:

$$\langle c + Cy^s, x \rangle \uparrow \max_x, \quad x \in X, \quad (LP_x)$$

так что справедливым будет неравенство

$$\langle c + Cy^s, x^{s+1} \rangle + \rho_s/2 \geq \sup_x \{ \langle c + Cy^s, x \rangle \mid x \in X \}. \quad (8)$$

Шаг 2. Найти $\rho_s/2$ -решение x^{s+1} задачи линейного программирования по другой переменной

$$\langle d + x^{s+1}C, y \rangle \uparrow \max_y, \quad y \in Y, \quad (LP_y)$$

так что справедливым будет неравенство

$$\langle d + x^{s+1}C, y^{s+1} \rangle + \rho_s/2 \geq \sup_y \{ \langle d + x^{s+1}C, y \rangle \mid y \in Y \}. \quad (9)$$

Шаг 3. Если

$$F(x^{s+1}, y^{s+1}) - F(x^{s+1}, y^s) \leq \tau, \quad (10)$$

где τ — заданная точность, то STOP. Иначе положить $s := s + 1$ и перейти на шаг 1. #

Теорема 2. В условиях теоремы 1 последовательность пар (x^s, y^s) , генерируемая Y-процедурой, сходится к точке (\hat{x}, \hat{y}) , удовлетворяющей (6) и (7). #

Отметим, что оба метода сходятся к точке с одинаковыми свойствами и обладают одними и теми же особенностями.

3. Алгоритм глобального поиска

Поскольку локальный поиск не обеспечивает достижения глобального решения, то далее представим алгоритм глобального поиска в задаче (BLP), базирующийся, как уже упоминалось, на стратегии глобального поиска для задач д.с. программирования.

Пусть даны допустимая точка $(x^0, y^0) \in D \triangleq X \times Y$, числовые последовательности $\{\tau_k\}$, $\{\delta_k\}$. $\tau_k, \delta_k > 0, k = 0, 1, 2, \dots, \tau_k \downarrow 0, \delta_k \downarrow 0, (k \rightarrow \infty)$, множество направлений $Dir = \{(u^1, v^1), \dots, (u^N, v^N) \in \mathbb{R}^{m+n} \mid (u^s, v^s) \neq 0, s = 1, \dots, N\}$, числа $\gamma_- \triangleq \inf(g, D)$ и $\gamma_+ \triangleq \sup(g, D)$ и параметры алгоритма ν и q .

Алгоритм глобального поиска

Шаг 0. Положить $k := 1, (\bar{x}^k, \bar{y}^k) := (x^0, y^0), s := 1, p := 1, \gamma := \gamma_-, \Delta\gamma = (\gamma_+ - \gamma_-)/q$.

Шаг 1. Начиная из точки $(\bar{x}^k, \bar{y}^k) \in D$, с помощью специального метода локального поиска построить τ_k -критическую точку $(x^k, y^k) \in D$ в задаче (P) такую, что $F(x^k, y^k) \geq F(\bar{x}^k, \bar{y}^k)$. Положить $\zeta_k := F(x^k, y^k)$.

Шаг 2. По точке $(u^s, v^s) \in Dir$ построить точку (\bar{u}^s, \bar{v}^s) аппроксимации поверхности уровня функции $f(\cdot)$, такую что $f(\bar{u}^s, \bar{v}^s) = \gamma + \zeta_k$.

Шаг 3. Если $g(\bar{u}^s, \bar{v}^s) > \gamma + \nu\gamma$, то положить $s := s + 1$ и перейти на шаг 2, иначе перейти на шаг 4.

Шаг 4. Начиная с точки (\bar{u}^s, \bar{v}^s) , специальным методом локального поиска построить δ_k -критическую точку $(\hat{x}^s, \hat{y}^s) \in D$ в задаче (P).

Шаг 5. Найти δ_k -решение $(x_0^s, y_0^s), f(x_0^s, y_0^s) = \gamma + \zeta_k$, задачи уровня

$$\begin{aligned} & \langle \nabla_x f(x_0^s, y_0^s), \hat{x}^s - x_0^s \rangle + \langle \nabla_y f(x_0^s, y_0^s), \hat{y}^s - y_0^s \rangle + \delta_k \geq \\ & \geq \sup_{x,y} \{ \langle \nabla_x f(x, y), \hat{x}^s - x \rangle + \langle \nabla_y f(x, y), \hat{y}^s - y \rangle : f(x, y) = \gamma + \zeta_k \}. \end{aligned} \quad (11)$$

Шаг 6. Вычислить величину

$$\eta_k(\gamma) = \gamma - g(\hat{x}^s, \hat{y}^s) + \langle \nabla_x f(x_0^s, y_0^s), \hat{x}^s - x_0^s \rangle + \langle \nabla_y f(x_0^s, y_0^s), \hat{y}^s - y_0^s \rangle. \quad (12)$$

Шаг 7. Если $\eta_k(\gamma) \leq 0, s < N$, то положить $s := s + 1$ и вернуться на шаг 2.

Шаг 8. Если $\eta_k(\gamma) \leq 0, s = N$, то положить $\gamma := \gamma + \Delta\gamma, s := p$ и вернуться на шаг 2.

Шаг 9. Если $\eta_k(\gamma) > 0$, то положить $(\bar{x}^{k+1}, \bar{y}^{k+1}) := (\hat{x}^s, \hat{y}^s)$, $k := k + 1$, $s := s + 1$, $p := s$ и перейти на шаг 1.

Шаг 10. Если $s = N$, $\eta_k(\gamma) \leq 0$, $\forall \gamma \in [\gamma_-, \gamma_+]$ (т.е. одномерный поиск по γ на отрезке $[\gamma_-, \gamma_+]$ закончен), то Stop. #

Поясним подробнее каждый шаг алгоритма и отметим его особенности по сравнению с общей стратегией глобального поиска.

Шаг 0 — инициализация, шаг 1 — локальный поиск в задаче (BLP) с помощью одного из представленных выше методов. Шаг 2 — построение аппроксимации поверхности уровня функции $f(\cdot)$ с помощью заданного множества направлений Dir . Шаг 3 — проверка точки на пригодность к дальнейшим исследованиям с помощью неравенства, вытекающего из условий оптимальности для задач д.с. программирования из [2]; ν — параметр алгоритма, с помощью варьирования которого можно изменять точность выполнения этого неравенства (с целью уменьшить влияние машинных ошибок округления). Шаг 4 — дополнительный локальный поиск, который введен в алгоритм вместо решения линеаризованной задачи (см. схему стратегии в [2]). Можно показать, что такое изменение схемы глобального поиска допустимо. Шаг 5 — решение вспомогательной задачи уровня. Шаг 6 — вычисление оценки качества проведенной итерации. Шаги 7-10 — критерий останова и возвраты (внутренние циклы — по точкам аппроксимации и "кусочкам" отрезка $[\gamma_-, \gamma_+]$, внешний цикл — по найденным различным критическим точкам); q — параметр алгоритма, с помощью варьирования которого происходит разбиение отрезка одномерного поиска на различное количество частей.

Тестирование данного алгоритма производилось при решении специальной задачи билинейного программирования, возникающей при отыскании ситуации равновесия по Нэшу в биматричной игре.

Список литературы

- [1] Ф.П. Васильев. *Методы оптимизации*. М.: Факториал Пресс, 2002.
- [2] А.С. Стрекаловский *Элементы невыпуклой оптимизации*. Новосибирск: Наука, 2003, 356 с.
- [3] Б.М. Мухамедиев. *О решении задачи билинейного программирования и отыскании всех ситуаций равновесия в биматричных играх*. - Журн. вычисл. матем. и матем. физики, 1978, т.18, N2, с.351-359.
- [4] А.В. Орлов, А.С. Стрекаловский. *О поиске ситуаций равновесия в биматричных играх*. - Автоматика и телемеханика, 2004, N2, с. 55-68.
- [5] O.L. Mangasaryan. *Bilinear Separation of Two Sets in n-Space*. - Computational Optimization and Applications, 1993, N2, p. 207-227.
- [6] C.A. Floudas, V. Visweswaran V. *Quadratic optimization*. - В кн.: Handbook of global optimization. Ed. by R. Horst., P. Pardalos. Dordrecht: Kluwer Academic Publishers, 1995, p.224-228.

ON A LOCAL AND GLOBAL SEARCH IN BILINEAR PROGRAMMING

A.V. Orlov

*Institute of system dynamic and control theory SB RAS, Irkutsk
e-mail: anor@icc.ru*

Abstract. In this paper we suggest the approach for solving bilinear programming problems with disjoint constraints. This approach is based on the strategy of global search in d.c. maximization problems developed by A.S. Strekalovsky. We consider two local search methods in this problems and the scheme of global search algorithm.

Key words: bilinear programming, d.c. decomposition of the goal function, local search, global search

О ВАРИАЦИОННЫХ МЕТОДАХ РЕШЕНИЯ СИСТЕМ НЕЛИНЕЙНЫХ УРАВНЕНИЙ

Е.Г. Петрова, А.С. Стрекаловский

Институт динамики систем и теории управления СО РАН

Аннотация. В данной работе рассматривается решение систем нелинейных уравнений с d.c. функциями посредством глобального поиска, основанного на теории d.c. минимизации.

Ключевые слова: невыпуклая, негладкая, оптимизация, d.c. функция, глобальный, локальный.

1. Постановка задачи

Рассмотрим систему нелинейных уравнений

$$f_i(x) = 0, \quad i = 1, \dots, m, \quad (1)$$

где $x \in \mathbb{R}^n$, а $f_i(x)$ - d.c. функция, т.е. функция, представляемая в виде разности двух выпуклых функций:

$$f_i(x) = g_i(x) - h_i(x), \quad (2)$$

где $g_i(x), h_i(x)$ - выпуклые на \mathbb{R}^n функции, $i = 1, \dots, m$.

Класс d.c. функций является достаточно широким векторным пространством, порожденным конусом выпуклых функций и замкнутым относительно часто встречающихся в оптимизации операций, таких, как сумма, разность, максимум, минимум и т.д.

Как известно [5], при численном решении задачи (1), например, методом Ньютона, область сходимости метода (множество, из которого выбираются стартовые приближения, при которых метод сходится) очень мала. С другой стороны, известно [1], что система уравнений может быть сведена к задаче минимизации некоторой функции. Действительно, если

$$F(y_1, \dots, y_m) > 0 \quad \text{при} \quad (y_1, \dots, y_m) \neq (0, \dots, 0), \quad (3)$$

$$F(0, \dots, 0) = 0, \quad (4)$$

то решение системы (1) равносильно минимизации функции $F(f_1(x), \dots, f_m(x))$. Таким образом, вместо задачи (1) возникает следующая вариационная задача:

$$F(x) \downarrow \min, \quad x \in \mathbb{R}^n. \quad (5)$$

Очевидно, что в качестве целевой функции, удовлетворяющей условиям (3)-(4) можно взять

$$F(x) = \sum_{i=1}^m |f_i(x)| \quad (6)$$

Используя известные свойства d.c. функций [7], нетрудно видеть, что

$$|f_i(x)| = 2 \max\{g_i(x); h_i(x)\} - [g_i(x) + h_i(x)]$$

Функции $\bar{g}_i(x) = 2 \max\{g_i(x); h_i(x)\}$ и $\bar{h}_i(x) = g_i(x) + h_i(x)$ являются выпуклыми.

Тогда справедливы следующие равенства:

$$F(x) \triangleq \sum_{i=1}^m |f_i(x)| = \sum_{i=1}^m [\bar{g}_i(x) - \bar{h}_i(x)] = \sum_{i=1}^m \bar{g}_i(x) - \sum_{i=1}^m \bar{h}_i(x) = g(x) - h(x),$$

где $g(x) = \sum_{i=1}^m \bar{g}_i(x)$, $h(x) = \sum_{i=1}^m \bar{h}_i(x)$ - очевидно, выпуклые функции.

Итак, функция $F(x)$, определяемая формулой (6), также является д.с. функцией. Заметим, что $F(x)$ - негладкая функция, а значит, для решения задачи (5) можно использовать лишь алгоритмы недифференцируемой оптимизации. Среди негладких задач минимизации наиболее изученными являются выпуклые задачи. Из методов негладкой оптимизации можно выделить следующие классы:

1. Обобщенный градиентный спуск. Это класс немонотонных субградиентных процессов минимизации выпуклых функций, основанных на движении в направлении, которое дает уменьшение расстояния до точки минимума, если шаговый множитель достаточно мал.

2. Обобщенные градиентные методы с растяжением пространства, в которых для ускорения сходимости используются преобразования пространства.

3. Использование секущих гиперплоскостей для аппроксимации графика функции или для последовательного уменьшения объема области локализации экстремума. К числу методов этого типа относится и метод эллипсоидов и его модификации.

Однако в невыпуклых задачах эти методы крайне неэффективны и не могут гарантировать, что полученное с их помощью решение является глобальным.

Далее будет описан подход для негладкой д.с. минимизации [7], развернутый для решения задачи (5) – (6).

2. Локальный поиск

Рассмотрим задачу д.с. минимизации

$$F(x) = g(x) - f(x) \downarrow \min, \quad x \in \mathbb{R}^n, \quad (7)$$

где функции $g(x)$, $f(x)$ - выпуклые. Предположим, что в любой точке $x \in \mathbb{R}^n$ можно найти некоторый субградиент $x^* \in \partial f(x)$. Опишем специальный метод локального поиска, используемый для решения этой задачи.

Пусть задано некоторое начальное приближение $x^0 \in \mathbb{R}^n$. Далее, если известна допустимая точка $x^s \in \mathbb{R}^n$, то следующее приближение x^{s+1} будем искать как приближенное решение линеаризованной в точке x^s задачи:

$$g(x) - \langle x_s^*, x \rangle \downarrow \min, \quad x \in \mathbb{R}^n, \quad x_s^* \in \partial f(x^s). \quad (8)$$

Более точно это означает, что точка x^{s+1} удовлетворяет условию:

$$g(x^{s+1}) - \langle x_s^*, x^{s+1} \rangle \leq \inf_x \{g(x) - \langle x_s^*, x \rangle\} + \delta_s, \quad (9)$$

где последовательность δ_s такова, что

$$\delta_s \geq 0, \quad s = 1, 2, \dots, \quad \sum_{s=0}^{\infty} \delta_s < \infty. \quad (10)$$

Приведем теорему сходимости данного алгоритма.

Теорема 0..1. Пусть в задаче (7) целевая функция $F = g - h$ ограничена снизу на \mathbb{R}^n , и в любой допустимой точке x можно найти некоторый субградиент $x^* \in \partial f(x)$. Тогда последовательность $\{x^s\}$, генерируемая по правилу (9) удовлетворяет условию:

$$\lim_{s \rightarrow \infty} \left\{ \inf_{x \in \mathbb{R}^n} [g(x) - g(x^{s+1}) - \langle x_s^*, x - x^{s+1} \rangle] \right\} = 0, \quad x_s^* \in \partial f(x^s). \quad (11)$$

При этом, если последовательность субградиентов сходится, т.е.

$$x_s^* \rightarrow z^* \in \partial f(z), \quad (12)$$

то любая предельная точка z последовательности $\{x^s\}$ удовлетворяет условию

$$\inf_{x \in \mathbb{R}^n} \{g(x) - g(z) - \langle z^*, x - z \rangle\} = 0, \quad z^* \in \partial f(z). \quad (13)$$

Из (13), в случае дифференцируемости функций f и g , вытекает классическое условие стационарности

$$\langle \nabla g(x_*) - \nabla f(x_*), x - x_* \rangle \geq 0. \quad (14)$$

Проведен численный эксперимент, доказывающий целесообразность применения предложенного алгоритма для решения задачи (5)-(6).

2. Глобальный поиск

Перейдем к нахождению глобального решения задачи (7). Приведем алгоритм поиска глобального минимума д.с функции, для которого, как и в предыдущем случае, доказана теорема сходимости.

Пусть заданы начальная точка $x_0 \in \mathbb{R}^n$ и числовые последовательности τ_k и δ_k :

$$\tau_k, \delta_k > 0, \quad k = 0, 1, 2, \dots, \quad \tau_k \downarrow 0, \quad \delta_k \downarrow 0 \quad (k \rightarrow \infty).$$

Шаг 0. Положить $k := 0$, $x^k := x_0$.

Шаг 1. Начиная с $x^k \in \mathbb{R}^n$ посредством некоторого метода локального поиска построить τ_k -стационарную точку z^k , $\zeta_k := F(z^k) \leq F(x^k)$.

Шаг 2. Выбрать некоторое $\beta \in [\beta_-, \beta_+]$.

Шаг 3. Построить некоторую аппроксимацию

$$\mathcal{A}_k(\beta) = \{v^1, \dots, v^{N_k} | f(v^i) = \beta - \zeta_k, \quad i = 1, \dots, N_k, \quad N_k = N_k(\beta)\}. \quad (15)$$

Шаг 4. Сформировать семейство индексов

$$I_k = I_k(\beta) = \{i \in \{1, \dots, N_k | g(v^i) \leq \beta\}\}. \quad (16)$$

Шаг 5. Для каждого $i \in I_k$ найти u^i :

$$g(u^i) - \langle v_i^*, u^i \rangle - \delta_k \leq \inf_x \{g(x) - \langle v_i^*, x \rangle | x \in \mathbb{R}^n\}, \quad (17)$$

где $v_i^* \in \partial f(v^i)$.

Шаг 6. Для каждого $i \in I_k$ найти w^i :

$$\langle w_i^*, u^i - w^i \rangle + \delta_k \leq \sup_v \{\langle v^*, u^i - v \rangle | f(v) = \beta - \zeta_k\}, \quad (18)$$

где $w_i^* \in \partial f(w^i)$, $u^* \in \partial f(u)$.

Шаг 7. Положить $\eta_k(\beta) := \eta_k^0(\beta) + \beta$, где

$$\eta_k^0(\beta) := \langle w_j^*, u^j - w^j \rangle - g(u^j) \stackrel{\Delta}{=} \max_{1 \leq i \leq N} \{ \langle w_i^*, u^i - w^i \rangle - g(u^i) \}, \quad (19)$$

а $w_i^* \in \partial f(w^i)$.

Шаг 8. Если $\eta_k(\beta) > 0$, то положить $x^{k+1} := u^j$ и вернуться на шаг 1.

Шаг 9. Если $\eta_k(\beta) \leq 0$, то положить $\beta := \beta + \Delta\beta \in [\beta_-, \beta_+]$ и идти на шаг 3.

Шаг 10. Если $\eta_k(\beta) \leq 0 \quad \forall \beta \in [\beta_-, \beta_+]$ (т.е. одномерный поиск по β завершен), то положить $k := k + 1$, $x^{k+1} := z^k$ и вернуться на шаг 1.

Наибольшую трудность в предложенном методе вызывают шаги 3, 5 и 6, на которых возникает необходимость построить аппроксимацию, решить линеаризованную задачу:

$$g(x) - \langle y_i^*, x \rangle \downarrow \min_x, \quad y_i^* \in \partial f(y^i), \quad (20)$$

и задачу уровня:

$$\langle v^*, u^i - v \rangle \uparrow \max_v, \quad f(v) = \beta - \zeta_k, \quad v^* \in \partial f(v). \quad (21)$$

Введем определение, связанное с шагом 3.

Определение 0..1. Аппроксимацию $\mathcal{R}(\zeta, \beta)$ назовем $(\Delta, \delta, \varepsilon, \theta)$ -разрешающей, где $\Delta, \delta, \varepsilon > 0$, $0 < \theta < 1$, если из того, что z не является Δ -решением задачи (7), т.е.

$$F(z) > \inf(F, \mathbb{R}^n) + \Delta, \quad (22)$$

следует выполнение двух неравенств

$$\eta(\zeta, \beta) \geq 0, \quad (23)$$

$$\eta(\zeta, \beta) \geq \theta\varphi(z) - \varepsilon. \quad (24)$$

Если неравенства (23) и (24) выполняются как строгие, то аппроксимация $\mathcal{R}(\zeta, \beta)$ называется строго разрешающим набором.

Будем предполагать, что на каждой итерации стратегии глобального поиска на шаге 3 строится (строго) $(\Delta_k, \delta_k, \varepsilon_k, \theta_k)$ -разрешающий набор $\mathcal{R}(\zeta_k, \beta) := \mathcal{A}(\beta)$. Для того, чтобы вышеописанная стратегия была в определенной мере обоснованной, сделаем следующие предложения:

(HL) $\forall \delta > 0 \quad \forall \beta \in [\beta_-, \beta_+] \quad \forall z \in \mathbb{R}^n \quad \forall v : f(v) = \beta - F(z), g(v) \leq \beta$ можно найти точку u , удовлетворяющую (17) при $u^i = u$, $v^i = v$, $\delta_k = \delta$.

(HU) $\forall \delta > 0 \quad \forall \beta \in [\beta_-, \beta_+] \quad \forall z, u \in \mathbb{R}^n$ можно отыскать точку $w : f(w) = \beta - F(z)$, $g(v) \leq \beta$, удовлетворяющую (18) при $u^i = u$, $w^i = w$, $\delta_k = \delta$.

(HR) $\forall \beta \in [\beta_-, \beta_+] \quad \forall (\Delta, \tau, \delta, \varepsilon, \theta) : \Delta, \tau, \delta, \varepsilon > 0, 0 < \theta \leq 1, \varepsilon \geq \theta\Delta$, для любой τ -стационарной точки $z \in \mathbb{R}^n$, не являющейся Δ -решением задачи (7) можно построить (строго) $(\Delta, \delta, \varepsilon, \theta)$ -разрешающий набор $\mathcal{R}(\zeta, \beta)$, $\zeta \stackrel{\Delta}{=} F(z)$.

Далее, рассмотрим следующее условие на начальное приближение:

$$\exists v \in \mathbb{R}^n \quad \exists \chi > 0 : F(v) \geq F(x^0) + \chi. \quad (25)$$

Кроме того, предположим, что числовые последовательности $\{\Delta_k\}$, $\{\tau_k\}$, $\{\delta_k\}$, $\{\varepsilon_k\}$ и $\{\theta_k\}$ таковы, что

$$\left. \begin{aligned} \Delta_k, \delta_k, \varepsilon_k > 0, \quad 0 < \theta \leq \theta_k \leq 1, \\ k = 0, 1, 2, \dots, \quad \tau_k \downarrow 0, \quad \delta_k \downarrow 0, \quad \varepsilon_k \downarrow 0 (k \rightarrow \infty). \end{aligned} \right\} \quad (26)$$

Теорема 0..2. Пусть в задаче (7) целевая функция $F(\cdot)$ ограничена снизу на \mathbb{R}^n и в любой точке $x \in \mathbb{R}^n$ можем найти некоторый субдифференциал функции $f(\cdot) : x^* \in \partial f(x)$. Пусть, кроме того, выполнены предложения (HL), (HU), (HR), (25) и (26). Наконец, пусть выполнено условие согласования последовательностей:

$$\varepsilon_k \geq \theta_k \Delta_k, \quad k = 0, 1, 2, \dots \quad (27)$$

Тогда i) Последовательность $\{z^k\}$, генерируемая стратегией глобального поиска, является минимизирующей в задаче (7).

ii) При этом всякая предельная точка последовательности z^k доставляет точную нижнюю грань функции $F(\cdot)$.

Проведен численный эксперимент по проверке эффективности стратегии глобального поиска применительно к решению систем нелинейных уравнений.

Список литературы

- [1] Н.С. Бахвалов, Н.П. Жидков, Г.М. Кобельков *Численные методы*. М.: Наука, 1987, 600 с.
- [2] Ф.П. Васильев *Численные методы решения экстремальных задач*. М.: Наука, 1988, 552 с.
- [3] В.Ф. Демьянов, Л.В. Васильев *Недифференцируемая оптимизация*. М.: Наука, 1981, 384 с.
- [4] В.С. Михалевич, В.А. Трубин, Н.З. Шор *Оптимизационные задачи производственно-транспортного планирования: Модели, методы, алгоритмы*. М.: Наука, Гл. ред. физ.-мат. лит., 1986, 264 с.
- [5] Дж. Ортега, В. Рейнболдт *Итерационные методы решения нелинейных систем уравнений со многими неизвестными*. М.: Мир, 1975, 560 с.
- [6] А.С. Стрекаловский *Минимизация разности двух выпуклых функций*. - Серия: Оптимизация и управление, Вып.5, Иркутск: Иркутский университет, 2002, 53 с.
- [7] А.С. Стрекаловский *Элементы невыпуклой оптимизации*. Новосибирск: Наука, 2003, 356 с.
- [8] Н.З. Шор *Методы минимизации недифференцируемых функций и их приложения*. Киев: Наукова думка, 1976, 200 с.

ABOUT VARIATIONAL METHODS OF THE DECISION OF SYSTEMS OF THE NONLINEAR EQUATIONS

E.G. Petrova, A.S. Strekalovsky

Institute of system dynamics and control theory SB RAS

Abstract. We consider a solving of a system of the nonlinear equations represented by d.c. functions by means of global search methods based on the theory of d.c. minimization.

Key words: nonconvex, nondifferentiable, optimization, d.c. function, global, local.

О НЕКОТОРЫХ ПРИКЛАДНЫХ ЗАДАЧАХ ОПТИМИЗАЦИИ

А.С. Стрекаловский

Институт динамики систем и теории управления СО РАН, Иркутск
e-mail: strekal@icc.ru

Аннотация. В работе рассматриваются возникающие в приложениях задачи оптимизации, в формулировках которых участвуют d.c. функции, т.е. функции, представимые разностью двух выпуклых функций. Представлены основные этапы поиска глобального решения в этих невыпуклых задачах.

Ключевые слова: d.c. функция, локальный поиск, глобальный поиск, линеаризованная задача, аппроксимация.

Введение

Как известно [1, 2, 4, 5, 7], класс $DC(\Omega)$ d.c. функций на выпуклом множестве $\Omega \in \mathbb{R}^n$ является линейным пространством, порожденным конусом выпуклых функций, хорошо изученным в литературе [1, 2, 3]. Далее, класс $C^2(\Omega)$ дважды непрерывно дифференцируемых функций является подпространством $DC(\Omega)$. Например, любой полином является d.c. функцией. Кроме того, в случае компактного Ω пространство $DC(\Omega)$ плотно в пространстве непрерывных функций в топологии равномерной сходимости. Следовательно, любая непрерывная задача оптимизации может быть аппроксимирована с любой точностью некоторой задачей d.c. программирования. Поэтому понятен интерес специалистов к задачам оптимизации с d.c. функциями [4]. Некоторые из них даже полагают, что любая прикладная задача является задачей d.c. программирования, с возможно скрытым "d.c. характером".

Думается, основной вопрос настоящего момента не в этом, а в возможности и методах численного решения подобных задач. Дело в том, что тысячи инженеров и информатиков по всему миру, воспитанные в идеологии методов ветвей и границ, отсечений, перенумерации вершин многогранников и т.д. и т.п., пытаются решить тем или иным методом интересные прикладные задачи достаточно большой размерности. Однако, вышеперечисленные подходы, как известно [5], страдают от "проклятия размерности", когда объем вычислений растет экспоненциально относительно размерности задачи.

С другой стороны, классические методы выпуклой оптимизации по существу забыты этой категорией специалистов, поскольку они оказываются крайне неэффективными в смысле поиска глобальных решений в невыпуклых (например, d.c.) задачах оптимизации, отыскивая в зависимости от начального приближения лишь локальные решения или стационарные точки.

Нами был разработан подход [7], возвращающий возможность использования классических методов оптимизации для решения невыпуклых задач. С другой стороны, эта методика может быть оценена как редуцирование невыпуклых задач d.c. оптимизации к семейству (частично) линеаризованных (по базовой невыпуклости) задач, для решения которых и можно применить методы выпуклой оптимизации. Ниже будут приведены основные этапы разработанного в [7] подхода, а также три непрерывные задачи оптимизации, имеющие прикладное значение. Эти "проекты" были предложены мною моим

сотрудникам для исследования и решались под моим руководством.

1. Основные этапы невыпуклой оптимизации

Первым этапом невыпуклой оптимизации является локальный поиск, при этом методы локального поиска отражают специфику задачи. Например, для задачи д.с. минимизации

$$F(x) \triangleq g(x) - f(x) \downarrow \min, \quad x \in D, \quad (1)$$

где $g(\cdot)$ и $f(\cdot)$ — выпуклые функции, а $D \subset \mathbb{R}^n$, весьма эффективной себя показала [7] следующая процедура генерирования последовательности $\{x^s\}$. Если известно приближение $x^s \in D$, то следующая итерация x^{s+1} выбирается как приближенное решение линейаризованной задачи

$$g(x) - \langle \nabla f(x^s), x \rangle \downarrow \min, \quad x \in D. \quad (2)$$

В случае выпуклости D задача (2) выпукла и может быть решена одним из методов выпуклой оптимизации. Кроме того, предельная точка x_* последовательности $\{x^s\}$ будет решением линейаризованной (в этой точке x_*) задачи, а поэтому и критической в исходной задаче (1). Отметим, что довольно часто можно предложить метод локального поиска, отличный от (2) и специфичный только для конкретной задачи [7].

Вторым и новым этапом глобального поиска является (чаще всего экспериментальное) построение аппроксимации некоторой поверхности, например, поверхности уровня выпуклой функции или границы надграфика некоторой выпуклой функции [7]. Этот момент продиктован условиями глобальной оптимальности [7] и позволяет рассматривать даже недопустимые точки, с помощью которых генерируются новые (приближенно) критические точки. Так, например, исходя из точек аппроксимации можно вновь применить локальный поиск или новые методы решения линейаризованных (в точках аппроксимации) задач.

Смысл *третьего этапа* вычислений заключается в построении некоторой оценки η_k , которая отражает эффективность проведенных вычислений, в частности, качество построенной аппроксимации. Как обычно [7], если $\eta_k > 0$, то удалось построить точку лучшую, чем исследуемая, и можно вновь переходить к локальному поиску, причем, с повышенной точностью.

Как показывает теория и практика вычислений [7], в случае "хороших" аппроксимаций (разрешающих наборов) удастся решить достаточно трудные прикладные задачи даже динамические и комбинаторные. Новая группа задач, исследуемая в настоящий момент, приводится ниже.

2. Задачи классификации

Задача классификации представляет собой отнесение образа к одному из нескольких попарно непересекающихся множеств. Примером таких задач служат распознавание текста или речи, идентификация личности, задача определения кредитоспособности клиента банка, задачи диагностики и прогнозирования в медицине, а также определения жизнеспособных и склонных к банкротству фирм.

Часто каждый объект x описывается совокупностью $(x_1, \dots, x_i, \dots, x_n)$ основных признаков (свойств), где i -ая координата вектора x определяет значение i -ого признака.

К сожалению, в большинстве случаев классы нельзя разделить между собой гиперплоскостями, т.е. линейная отделимость не имеет места. Поэтому необходимо предложить новые понятия отделимости, соответствующие конкретному случаю.

Рассмотрим задачу об отделении двух конечных множеств \mathcal{A} и \mathcal{B} из \mathbb{R}^n минимальным количеством гиперплоскостей. Пусть множества \mathcal{A} и \mathcal{B} состоят из M и N точек соответственно. Представим их в виде матриц A и B размерности $M \times n$ и $N \times n$, столбцами которых являются координаты точек из множеств \mathcal{A} и \mathcal{B} соответственно.

Скажем, что множества \mathcal{A} и \mathcal{B} полиэдрально отделимы P гиперплоскостями, если найдутся $(v^p, \gamma_p) \in \mathbb{R}^{n+1}$, $p = 1, \dots, P$ при которых

$$\left. \begin{aligned} \langle v^p, a^i \rangle &\leq \gamma_p - 1, \quad \forall i = 1, \dots, M \quad \forall p = 1, \dots, P, \\ \forall j = 1, \dots, N \quad \exists p \in \{1, \dots, P\} : \langle b^j, v^p \rangle &> \gamma_p + 1. \end{aligned} \right\} \quad (3)$$

Можно показать, что множества \mathcal{A} и \mathcal{B} полиэдрально отделимы в том и только в том случае, когда $\text{conv}(\mathcal{A}) \cap (\mathcal{B}) = \emptyset$.

Далее, из определения следует, что ошибка классификации для точки a^i выражается числом $\max\{0; \max_{1 \leq p \leq P} (\langle v^p, a^i \rangle - \gamma_p + 1)\}$, а для точки b^j — $\max\{0; \min_{1 \leq p \leq P} (-\langle v^p, b^j \rangle + \gamma_p + 1)\}$.

Тогда общая взвешенная ошибка классификации имеет вид:

$$\Delta(V, \Gamma) = \frac{1}{M} \sum_{i=1}^M \max\{0; \max_{1 \leq p \leq P} (\langle v^p, a^i \rangle - \gamma_p + 1)\} + \frac{1}{N} \sum_{j=1}^N \max\{0; \min_{1 \leq p \leq P} (-\langle v^p, b^j \rangle + \gamma_p + 1)\}, \quad (4)$$

где V — матрица $(P \times n)$ со столбцами v^p , $p = 1, \dots, P$, а $\Gamma = (\gamma_1, \dots, \gamma_P)^T \in \mathbb{R}^P$.

Предложение 1. Множества \mathcal{A} и \mathcal{B} полиэдрально отделимы тогда и только тогда, когда $\Delta = 0$.

Можно показать, что функция ошибки $\Delta(V, \Gamma)$ является d.c. функцией.

Решая задачу минимизации

$$\Delta(V, \Gamma) \downarrow \min, \quad x \in \mathbb{R}^n, \quad (5)$$

в случае, если $\Delta(V_*, \Gamma_*) = 0$, мы решаем и задачу классификации. В случае, если $\inf \Delta(V, \Gamma) > 0$, множества \mathcal{A} и \mathcal{B} полиэдрально неотделимы.

Итак, задача (5) является задачей негладкой d.c. минимизации. Результаты по ее исследованию представлены в докладе Дружининой О.В.

3. Задачи с d.c. ограничениями

Довольно большое число прикладных задач обладает d.c. структурой в ограничениях, отражающей пороговые значения некоторых параметров [3, 4, 5, 7]. Рассмотрим простейшую задачу такого типа

$$h(x) \downarrow \min, \quad x \in S, \quad F(x) \triangleq g(x) - f(x) \geq 0, \quad (6)$$

частным случаем которой является известная обратно-выпуклая задача

$$h(x) \downarrow \min, \quad x \in S, \quad g(x) \geq 0.$$

Здесь все данные $h(\cdot)$, $g(\cdot)$, $f(\cdot)$, S — выпуклы.

В задаче (6) даже разработка специальных методов локального поиска представляется трудной проблемой. Тем не менее, нами разработаны два варианта методов локального

спуска, один из которых основан на условиях глобальной оптимальности [7, 8], а второй — на развитии идеи Розена [6]. Совместно с Яковлевой Т.В. разработан первый вариант программы глобального поиска, доказавший свою сравнительную эффективность во время численного эксперимента.

4. Решение систем нелинейных уравнений

Весьма перспективным выглядит применение новых технологий для решения систем нелинейных уравнений

$$f_i(x) = 0, \quad i = 1, \dots, m, \quad (7)$$

где каждая функция $f_i(\cdot)$ является d.c. функцией $f_i(x) = g_i(x) - h_i(x)$, $i = 1, \dots, m$, а функции $g_i(\cdot)$ и $h_i(\cdot)$ выпуклы, $i = 1, \dots, m$.

Как известно [1, 2], система (7) равносильна решению экстремальной задачи:

$$\Phi(x) \triangleq \sum_{i=1}^m |f_i(x)| \downarrow \min. \quad (8)$$

Операции взятия модуля от $f_i(\cdot)$ и суммирования, как известно [3, 4, 5, 7], не выводят из класса d.c. функций. Поэтому задача (8) является задачей негладкой d.c. минимизации, т.е. негладкой невыпуклой задачей. Следовательно, к ней также применим общий подход, представленный в п. 1. Результаты исследований и численного эксперимента представлены моей дипломницей Петровой Е.Г.

Список литературы

- [1] Ф.П. Васильев *Методы оптимизации*. М.: Факториал Пресс, 2002.
- [2] В.Ф. Демьянов, Л.В.Васильев *Недифференцируемая оптимизация*. М.: Наука, 1981.
- [3] Hiriart-Urruty J.-B. *Generalized Differentiability, Duality and Optimization for Problem dealing with Difference of Convex Functions*. - Lecture Notes in Economics and Mathematical Systems, Springer - Verlag, Berlin. 1985, V. 256, p. 37–69.
- [4] R. Horst, H.Tuy *Global Optimization (Deterministic Approach)*. Berlin: Springer-Verlag , 1993.
- [5] R. Horst, P. M. Pardalos, V. Thoai *Introduction to Global Optimization*. Dordrecht: Kluwer, 1995.
- [6] J.B. Rosen *Iterative Solution of Nonlinear Optimal Control Problems*. - Journal SIAM Control, 1966, V. 4, № 1, p. 223-244.
- [7] А.С. Стрекаловский *Элементы невыпуклой оптимизации*. Новосибирск: Наука, 2003.
- [8] А.С. Стрекаловский, Т.В. Яковлева *О локальном и глобальном поиске в невыпуклых задачах оптимизации*. - Автоматика и телемеханика, 2004, № 3, с. 23-34.
- [9] А.С. Стрекаловский *Минимизирующие последовательности в задачах с d.c. ограничениями*. - ЖВММФ, 2005, т. 45, № 3, с. 435-447.

ABOUT SOME OPTIMIZATION PROBLEMS FROM APPLICATIONS

A.S. Strekalovsky

*Institute of system dynamic and control theory SB RAS, Irkutsk
e-mail: strekal@icc.ru*

Abstract. We consider a few optimization problems arising from different application fields. These problems deal with (d.c.) functions which can be represented by difference of two convex functions. We present here principal steps of a global solution search in such nonconvex problems.

Key words: d.c. function, local search, global search, linearized problem, approximation.

О МЕТОДАХ ЛОКАЛЬНОГО ПОИСКА ДЛЯ ЗАДАЧИ С D.C. ОГРАНИЧЕНИЕМ

Т.В. Яковлева

*Институт динамики систем и теории управления СО РАН, Иркутск
e-mail: yak@icc.ru*

Аннотация. В работе рассматриваются невыпуклые задачи оптимизации с ограничениями, в формировании которых участвуют d.c. функции, т.е. функции, представимые разностью двух выпуклых функций. Описываются два метода локального поиска, сочетающие спуск на поверхность уровня d.c. функции и решение частично линеаризованных задач. Изучается сходимость этих методов и приводятся критерии останова.

Ключевые слова: d.c. функция, локальный поиск, линеаризованная задача, поверхность уровня, теорема сходимости.

Введение

Достаточно большое число прикладных задач обладает d.c. структурой в ограничениях. Рассмотрим простейшую задачу такого типа

$$h(x) \downarrow \min, \quad x \in S, \quad F(x) \triangleq g(x) - f(x) \geq 0, \quad (P)$$

частным случаем которой является известная обратно-выпуклая задача

$$h(x) \downarrow \min, \quad x \in S, \quad g(x) \geq 0.$$

Здесь функции $h(\cdot)$, $g(\cdot)$, $f(\cdot)$ и множество S выпуклы.

Будем также рассматривать двойственную к (P) задачу

$$F(x) \uparrow \max, \quad x \in S, \quad h(x) \leq \beta, \quad (Q_\beta)$$

где β — некоторый параметр.

Предположим, что для задачи (P) выполнено условие регулярности:

$$\exists v \in S, \quad h(v) < h_* \triangleq \mathcal{V}(P), \quad F(v) < 0. \quad (H_0)$$

В задачах с d.c. ограничениями даже разработка специальных методов локального поиска представляется достаточно сложной задачей. Тем не менее, разработаны два варианта методов локального спуска, один из которых основан на условиях глобальной оптимальности [1, 2], а второй — на развитии идеи Розена [3]. Проведен вычислительный эксперимент по тестированию предложенных алгоритмов локального поиска.

1. Специальный метод локального поиска

Этот метод состоит из процедуры спуска на поверхность уровня $F(x) = 0$ и последовательного решения линеаризованных задач

$$f(x) - \langle \nabla g(u), x \rangle \downarrow \min, \quad x \in S, \quad h(x) \leq \beta; \quad (LQ(u, \beta))$$

где u и β — параметры. Перейдем к более детальному описанию процесса вычислений.

Процедура 1. Пусть задана допустимая точка $y \in S$, $F(y) \geq 0$. Если $F(y) = 0$, то полагаем $x(y) = y$. Если же $F(y) > 0$, то в силу выпуклости функций $g(\cdot)$ и $f(\cdot)$ найдется $\lambda \in]0, 1[$, такое что $F(x_\lambda) = 0$, где $x_\lambda = \lambda v + (1 - \lambda)y$, поскольку $F(y) > 0 > F(v)$ в силу (H_0) . В этом случае полагаем $x(y) := x_\lambda$.

Процедура 2 стартует из допустимой точки $\tilde{x} \in S$, $F(\tilde{x}) = 0$, и строит последовательность $\{u^r\}$, такую что $(r = 0, 1, 2, \dots)$

$$u^r \in S, \quad F(u^r) \geq 0, \quad h(u^r) \leq \beta, \quad (1)$$

где $\beta := h(\tilde{x})$, $u^0 := \tilde{x}$, по следующему правилу: если известна точка u^r , $r \geq 0$, удовлетворяющая (1), то следующая точка u^{r+1} строится как приближенное решение линейризованной выпуклой задачи $(LQ(u^r, \beta))$. Другими словами, справедливо

$$f(u^{r+1}) - \langle \nabla g(u^r), u^{r+1} \rangle - \delta_r \leq \inf_x \{f(x) - \langle \nabla g(u^r), x \rangle \mid x \in S, \quad h(x) \leq \beta\}, \quad (2)$$

где числовая последовательность $\{\delta_r\}$ такова, что

$$\delta_r > 0, \quad r = 0, 1, 2, \dots, \quad \sum_{r=1}^{\infty} \delta_r < +\infty. \quad (3)$$

Предложение 1. Предположим, что значение двойственной к (P) задачи (Q_ζ) конечно для некоторого $\zeta \geq \beta$, а функции $g(\cdot)$ и $f(\cdot)$ являются выпуклыми и непрерывно дифференцируемыми на открытой области Ω , $S \cup \{x \in \mathbb{R}^n \mid F(x) = 0\} \subset \Omega$.

Тогда последовательность $\{u^r\}$, генерируемая процедурой 2, удовлетворяет условию

$$\lim_{r \rightarrow \infty} \inf_x \{f(x) - f(u^r) - \langle \nabla g(u^r), x - u^r \rangle \mid x \in S, \quad h(x) \leq \beta\} = 0.$$

При этом для всякой предельной точки u_* последовательности $\{u^r\}$ справедливы неравенства:

$$\langle \nabla f(u_*) - \nabla g(u_*), x - u_* \rangle \geq 0 \quad \forall x \in S : h(x) \leq \beta,$$

$$F(u_*) \stackrel{\Delta}{=} g(u_*) - f(u_*) \geq 0.$$

В случае замкнутости S эта предельная точка u_* является нормально стационарной в двойственной задаче (Q_β) . #

Теперь перейдем к вопросу о том, как с помощью последовательности $\{u^r\}$, где $u^0 := \tilde{x}$, $\beta = h(\tilde{x})$, определить точку $y(\tilde{x})$. Для этого зададимся некоторым числом $\varepsilon > 0$, и пусть $r \geq 0$. Тогда, если $\delta_r \leq \varepsilon/2$ и справедливо неравенство

$$f(u^{r+1}) - f(u^r) - \langle \nabla g(u^r), u^{r+1} - u^r \rangle \geq -\varepsilon/2, \quad (4)$$

то полагаем $y = y(\tilde{x}, \varepsilon) := u^r$. При этом вместо (4) можно использовать неравенство

$$F(u^{r+1}) - F(u^r) \leq \varepsilon/2.$$

В обоих случаях точка $y = y(\tilde{x}, \varepsilon)$ является ε -приближенным решением линейризованной задачи $(LQ(y, \beta))$, где $\beta = h(\tilde{x})$.

Итак, пусть задана точка $x_0 \in S$, $F(x_0) \geq 0$, а также числовые последовательности $\{\delta_r\}$ и $\{\varepsilon_s\}$, удовлетворяющие (3) и условиям $\varepsilon_s > 0$, $s = 0, 1, 2, \dots$, $\varepsilon_s \downarrow 0$ ($s \rightarrow +\infty$).

Специальный метод локального поиска (СМЛП).

Шаг 0. (Инициализация) Положить $s := 0$, $x^s := x_0$, $\beta_s := h(x^s)$.

Шаг 1. (Процедура 2) Начиная из точки x^s , построить точку $y^s = y(x^s, \varepsilon_s)$ со свойствами $y^s \in S$, $F(y^s) \geq 0$, $h(y^s) \leq \beta_s$, являющуюся ε_s -решением линейризованной задачи $(LQ(y^s, \beta_s))$, т.е.

$$f(y^s) - \langle \nabla g(y^s), x - y^s \rangle \geq -\varepsilon_s \quad \forall x \in S : h(x) \leq \beta_s.$$

Шаг 2. (Критерий останова) Если $F(y^s) \leq 0$, то **СТОП**.

Шаг 3. (Процедура 1) С помощью y^s построить точку $u := x(y^s)$ со свойствами
 $u \in S, F(u) = 0, h(u) < h(y^s) \leq \beta_s$.

Шаг 4. Положить $s := s + 1, x^s := u, \beta_s := h(u)$ и вернуться на шаг 1. #

Теорема 1. Пусть функция $h(\cdot)$ и множество S выпуклы, а множество $\mathcal{F}_0 = \{x \in S \mid h(x) \leq h(x_0)\}$ ограничено, и справедливо также условие (H_0) .

Тогда специальный метод локального поиска:

а) при конечном числе итераций получает точку $y^N \in S, F(y^N) = 0$, которая является ε_N -решением линейризованной задачи $(LQ(y_N, \beta_N))$, где N — номер итерации останова;

б) при бесконечном числе итераций для последовательностей $\{x^s\}$ и $\{y^s\}$ со свойствами

$$\begin{aligned} x^s \in S, F(x^s) = 0, y^s \in S, F(y^s) > 0, \\ \beta^{s+1} := h(x^{s+1}) < h(y^s) \leq \beta_s := h(x^s), \\ \beta_* := \lim_{s \rightarrow \infty} \beta_s = \lim_{s \rightarrow \infty} h(y^s), \end{aligned}$$

справедливы условия $0 = F(x^s) = \lim_{s \rightarrow \infty} F(y^s), x_* = \lim_{s \rightarrow \infty} x^s = \lim_{s \rightarrow \infty} y^s$, для некоторого $x_* \in \mathbb{R}^n, F(x_*) = 0$.

Кроме того, точка x^* является решением линейризованной задачи $(LQ(x_*, \beta_*))$ и нормально стационарной в двойственной задаче $(Q(\beta_*))$. #

2. Модифицированный метод Розена

Другой метод локального поиска для задачи с д.с. ограничением основан на идее предложенного в 1966 г. Розеном [3] метода решения обратнo-выпуклых задач.

Рассмотрим линейризованные (в точке u) задачи вида:

$$h(x) \downarrow \min, x \in S, \langle \nabla g(u), x - u \rangle + g(u) - f(x) \geq 0. \quad (5)$$

Здесь в отличие от СМЛП целевая функция задачи (P) остается прежней, а линейризуется, причем лишь частично (по функции $g(\cdot)$), ограничение $F(x) \geq 0$.

Пусть в задаче (P) выполнено следующее условие:

$$\forall y \in S, F(y) = 0 \exists d = d(y) \in S : f(d) - f(y) < \langle \nabla g(y), d - y \rangle, \quad (6)$$

и имеется некоторая точка $u \in S : F(u) = 0$, в которой линейризуется функция $g(\cdot)$.

Метод локального поиска состоит из Процедуры 1 СМЛП и следующей процедуры.

Процедура 3 стартует из допустимой точки $\tilde{x} \in S, F(\tilde{x}) = 0$, и строит последовательность $\{u^r\}$ ($r = 0, 1, 2, \dots$) по следующему правилу. Если известно $u^r, r \geq 0$, удовлетворяющее (1), то u^{r+1} строится как приближенное решение выпуклой задачи:

$$h(x) \downarrow \min, x \in S, g(u^r) - f(x) + \langle \nabla g(u^r), x - u^r \rangle \geq 0. \quad (PL_r)$$

Другими словами, справедливо неравенство:

$$h(u^{r+1}) \leq \inf_x \{h(x) \mid x \in S, g(u^r) - f(x) + \langle \nabla g(u^r), x - u^r \rangle \geq 0\} + \delta_r, \quad (7)$$

где $\{\delta_r\}$ — числовая последовательность, удовлетворяющая условиям (3).

Пусть $u_0 \in S$ — некоторая точка. Введем обозначение для допустимого множества задачи (PL_r) : $D_r := D(u^r) \triangleq \{x \in S \mid g(u^r) - f(x) + \langle \nabla g(u^r), x - u^r \rangle \geq 0\}$.

Предложение 2. i) Предположим, что значение задачи (P) конечно, а функции $h(\cdot)$ и $g(\cdot)$ выпуклы и непрерывно дифференцируемы на открытой области Ω , $S \subset \Omega$.

Тогда последовательность $\{u^r\}$, генерируемая процедурой 3, удовлетворяет условию

$$\lim_{r \rightarrow \infty} [\inf_x \{h(x) - h(u^{r+1}) \mid x \in D_r\}] = 0.$$

ii) При этом, если множество $\mathcal{F}_0 = \{x \in S \mid h(x) \leq h(u_0) + \sum_{r=1}^{\infty} \delta_r\}$ ограничено, всякая предельная точка u^* последовательности $\{u^r\}$ является решением задачи:

$$h(x) \downarrow \min, \quad x \in D_* \triangleq D(u^*).$$

iii) В случае замкнутости S точка u^* является нормально стационарной в (P) . #

Теперь с помощью последовательности $\{u^r\}$, где $u_0 = \tilde{x}$, определим точку $y(\tilde{x})$. Для этого зададим число $\varepsilon > 0$. Тогда, если $\delta_r \leq \varepsilon$ и справедливо неравенство $h(u^r) - h(u^{r+1}) \leq \varepsilon$, то полагаем $y = y(\tilde{x}, \varepsilon) = u^{r+1}$, поскольку в этом случае u^{r+1} является ε -решением линейризованной задачи (PL_r) .

Модифицированный метод Розена (ММР).

Шаг 0. (Инициализация) Положить $s := 0$, $x^s := x_0$.

Шаг 1. (Процедура 3) Начиная с точки x^s , построить точку $y^s = y(x^s, \varepsilon_s)$, являющуюся ε_s -решением линейризованной задачи (PL_s) , т.е.

$$h(x) + \varepsilon_s \geq h(y^s) \quad \forall x \in S : \langle \nabla g(y^s), x - y^s \rangle + g(y^s) - f(x) \geq 0.$$

Шаг 2. (Критерий останова) Если $g(y^s) \leq 0$, то Stop.

Шаг 3. (Процедура 1) С помощью y^s построить точку $u := x(y^s)$

$$u \in S, \quad g(u) = 0, \quad h(u) < h(y^s) \leq h(x^s).$$

Шаг 4. Положить $x^{s+1} := u$, $s := s + 1$ и вернуться на шаг 1. #

Теорема 2. Пусть функция $h(\cdot)$ и множество S выпуклы, а множество $\mathcal{F}_0 = \{x \in S \mid h(x) \leq h(x_0)\}$ ограничено, и справедливы условия (H_0) и (6).

Тогда модифицированный метод Розена :

i) при конечном числе итераций получает точку $y^N \in S$, $F(y^N) = 0$, которая является ε_N -решением линейризованной задачи (PL_N) , где N — номер итерации останова;

ii) при бесконечном числе итераций для последовательностей $\{y^s\}$ и $\{x^s\}$, со свойствами

$$x^s \in S, \quad F(x^s) = 0, \quad y^s \in S, \quad F(y^s) > 0,$$

$$h(x^{s+1}) < h(y^s) \leq h(x^s),$$

$$\lim_{s \rightarrow \infty} h(x^s) = \lim_{s \rightarrow \infty} h(y^s),$$

справедливы условия: $0 = F(x^s) = \lim_{s \rightarrow \infty} F(y^s)$, $y_* = \lim_{s \rightarrow \infty} x^s = \lim_{s \rightarrow \infty} y^s$, для некоторого $y_* \in \mathbb{R}^n$, $F(y_*) = 0$.

Кроме того, y_* является решением задачи (PL_*) и нормально стационарной в (P) . #

Список литературы

- [1] А.С. Стрекаловский *Элементы невыпуклой оптимизации*. Новосибирск: Наука, 2003.

- [2] А.С. Стрекаловский, Т.В. Яковлева *О локальном и глобальном поиске в невыпуклых задачах оптимизации.* - Автоматика и телемеханика, 2004, № 3, с. 23-34.
- [3] J.B. Rosen *Iterative Solution of Nonlinear Optimal Control Problems.* - Journal SIAM Control, 1966, V. 4, № 1, p. 223-244.

ABOUT LOCAL SEARCH METHODS FOR OPTIMIZATION PROBLEM WITH D.C. CONSTRAINT

T.V. Yakovleva

*Institute of system dynamic and control theory SB RAS, Irkutsk
e-mail: yak@icc.ru*

Abstract. We consider an optimization problem with d.c. constraint arising from different application fields. These problems deal with (d.c.) functions which can be represented by difference of two convex functions. We present here two local search methods. One of them arising from the global optimality conditions, another based on the idea of Rosen's method for reverse-convex problems.

Key words: d.c. function, local search, linearized problem, level surface, convergence theorem.

МНОГОКРИТЕРИАЛЬНАЯ ОПТИМИЗАЦИЯ

КРИТЕРИЙ УСТОЙЧИВОСТИ ВЕКТОРНОЙ ЗАДАЧИ МИНИМИЗАЦИИ ПОРОГОВЫХ ФУНКЦИЙ¹

В. А. Емеличев, К. Г. Кузьмин

*Белорусский государственный университет, Минск
e-mail: emelichev@bsu.by, kuzminkg@mail.ru*

Аннотация. Рассматривается векторная (многокритериальная) булева задача минимизации пороговых функций. Указано необходимое и достаточное условие того типа устойчивости задачи, который является дискретным аналогом свойства полунепрерывности сверху по Хаусдорфу многозначного отображения, задающего паретовскую функцию выбора.

Ключевые слова: пороговые функции, векторная задача, множество Парето, устойчивость.

Введение

Проблема устойчивости задач дискретной оптимизации в последнее время привлекает внимание многих специалистов. В первую очередь это связано с широким распространением дискретных оптимизационных моделей в экономике, управлении, проектировании. Необходимость исследования устойчивости задач оптимизации возникает в связи с тем, что исходная информация большинства прикладных задач носит приближенный характер, требует учета различных факторов неопределенности и риска. Накопленный опыт показывает, что нельзя говорить о корректном решении произвольной оптимизационной задачи без исследования ее на устойчивость. При этом под устойчивостью задачи, как правило, подразумевают одно из классических свойств непрерывности или полунепрерывности (например, по Хаусдорфу или Бержу) оптимального отображения, которое каждому набору параметров задачи ставит в соответствие множество искомых решений (см., например, [1], [2]). В соответствии с этим понятием под устойчивостью задачи в дискретном случае понимают наличие такой окрестности исходных данных в пространстве параметров задачи, что по отношению к первоначальной всякая «возмущенная» задача с параметрами из этой окрестности обладает некоторым наперед заданным свойством инвариантности. К примеру, полунепрерывность сверху по Хаусдорфу оптимального отображения в задачах дискретной оптимизации превращается в свойство непооявления новых оптимальных решений при любых изменениях ее параметров в пределах «малой» окрестности исходных данных.

Исследованию устойчивости разнообразных векторных задач дискретной оптимизации посвящен ряд публикаций. Так, в работе [3] (см., также [2]) показано, что совпадение множества Парето с множеством Слейтера является необходимым и достаточным условием устойчивости векторной задачи целочисленного линейного программирования, состоящей в поиске множества Парето. В [4] этот результат перенесен на векторные задачи булева программирования с линейно-квадратичными частными критериями, а в [5] найден широкий класс вектор-функций, гарантирующих, что совпадение множеств Парето и Слейтера является критерием устойчивости векторной траекторной задачи. В этом же русле лежат

¹Работа выполнена в рамках Государственной программы фундаментальных исследований Республики Беларусь «Математические структуры 29» (проект 913/28).

и некоторые недавно появившиеся публикации. Так, например, в [6] получен критерий устойчивости векторной l_∞ -экстремальной траекторной задачи, а в [7] проведен сравнительный анализ пяти типов устойчивости по ограничениям векторной задачи оптимизации с конечным множеством целочисленных допустимых решений.

В настоящей работе приводится необходимое и достаточное условие устойчивости векторной задачи булева программирования, частными критериями которой являются пороговые функции. Интерес к пороговым функциям обусловлен величиной спектра их применения в дискретной математике и математической кибернетике (см., например, [8], [9]). Пороговые булевы функции используются в линейном булевом программировании, математической логике, при решении задач искусственного интеллекта, распознавания образов, некоторых задач теории графов (в частности, задач, связанных с проблематикой степенных последовательностей [10]). Кроме того, они находят применение в системах с параллельной обработкой информации, системах принятия решений, при моделировании нейронных сетей, диагностике дискретных устройств и многих других разделах прикладной математики.

1. Постановка задачи

На множестве булевых векторов (решений задачи) $X \subseteq \mathbf{E}^n = \{0, 1\}^n$, $|X| \geq 2$, $n \geq 2$, зададим векторный критерий

$$f(x, A, b) = (f_1(x, A_1, b_1), f_2(x, A_2, b_2), \dots, f_m(x, A_m, b_m)) \rightarrow \min_{x \in X}$$

компонентами которого (частными критериями) являются пороговые функции

$$f_i(x, A_i, b_i) = \begin{cases} 0, & \text{если } A_i x \leq b_i, \\ 1, & \text{если } A_i x > b_i, \end{cases}$$

где $i \in N_m = \{1, 2, \dots, m\}$, $m \geq 1$, A_i — i -я строка матрицы $A = [a_{ij}]_{m \times n} \in \mathbf{R}^{m \times n}$, $b = (b_1, b_2, \dots, b_m) \in \mathbf{R}^m$.

Под векторной (m -критериальной) задачей $Z^m(A, b)$ будем понимать задачу нахождения множества Парето (множества эффективных решений)

$$P^m(A, b) = \{x \in X : \pi^m(x, A, b) = \emptyset\},$$

где $\pi^m(x, A, b) = \{x' \in X : f(x, A, b) \geq f(x', A, b) \text{ \& } f(x, A, b) \neq f(x', A, b)\}$.

Следуя [6], [11], [12] и [13], задачу $Z^m(A, b)$ назовем устойчивой, если справедлива формула

$$\exists \varepsilon > 0 \quad \forall (A', b') \in \Omega(\varepsilon) \quad (P^m(A, b) \supseteq P^m(A + A', b + b')),$$

где $\Omega(\varepsilon)$ — множество таких пар (A', b') , что $A' \in \mathbf{R}^{m \times n}$, $\|A'\| < \varepsilon$, $b' \in \mathbf{R}^m$, $\|b'\| < \varepsilon$. Здесь $\|\cdot\|$ — норма в соответствующем пространстве.

Тем самым, задача $Z^m(A, b)$ устойчива, если существует окрестность пары (A, b) в пространстве $\mathbf{R}^{m \times (n+1)}$, в которой отсутствуют новые эффективные решения. Иначе говоря, устойчивость задачи $Z^m(A, b)$ — это дискретный аналог полунепрерывности сверху по Хаусдорфу в точке $(A, b) \in \mathbf{R}^{m \times (n+1)}$ оптимального отображения

$$P^m : \mathbf{R}^{m \times (n+1)} \rightarrow 2^X,$$

т. е. точечно-множественного отображения, которое каждому набору параметров задачи ставит в соответствие множество Парето.

Обозначим через $H^m(A, b)$ множество таких решений $x \in X$, что для всякого решения $x' \neq x$ выполняется одно из условий:

- (i) существует хотя бы один такой индекс $s \in N_m$, что справедливы неравенства

$$A_s x \leq b_s, \quad A_s x' \geq b_s;$$

- (ii) для любого индекса $i \in N_m$ верно только одно из неравенств

$$A_i x \leq b_i \quad \text{или} \quad A_i x' \geq b_i.$$

Очевидно, что $P^m(A, b) \subseteq H^m(A, b)$ при любых $A \in \mathbf{R}^n$, $b \in \mathbf{R}$.

2. Условия устойчивости

Теорема. *Задача $Z^m(A, b)$, $m \geq 1$, устойчива тогда и только тогда, когда выполняется равенство*

$$P^m(A, b) = H^m(A, b).$$

При переходе к однокритериальному случаю ($m = 1$) множество эффективных решений превращается в множество оптимальных решений $P^1(A, b)$ скалярной задачи $Z^1(A, b)$, где $A \in \mathbf{R}^n$, $b \in \mathbf{R}$, при этом легко видеть, что $P^1(A, b) = H^1(A, b)$. Поэтому однокритериальным аналогом теоремы является следующее утверждение.

Следствие 1. *Скалярная задача $Z^1(A, b)$ устойчива при любых $A \in \mathbf{R}^n$, $b \in \mathbf{R}$.*

Введем традиционное множество Слейтера $S^m(A, b)$, т. е. множество слабо эффективных решений задачи $Z^m(A, b)$:

$$S^m(A, b) = \{x \in X : \sigma(x) = \emptyset\},$$

где

$$\sigma(x) = \{x' \in X : \forall i \in N_m (f_i(x, A_i, b_i) > f_i(x', A_i, b_i))\}.$$

Очевидно, что $P^m(A, b) \subseteq H^m(A, b) \subseteq S^m(A, b)$ при любых $A \in \mathbf{R}^{m \times n}$, $b \in \mathbf{R}^m$. Поэтому на основании теоремы справедливо

Следствие 2. *Для того чтобы задача $Z^m(A, b)$, $m \geq 1$, была устойчива достаточно, чтобы выполнялось равенство $P^m(A, b) = S^m(A, b)$.*

Следующий пример свидетельствует о том, что совпадение множеств Парето и Слейтера не является, вообще говоря, необходимым условием устойчивости нашей задачи.

Пример. Пусть $n = 3$, $m = 2$, $X = \{x^1, x^2, x^3\}$, $x^1 = (1, 0, 0)^T$, $x^2 = (0, 1, 0)^T$, $x^3 = (0, 0, 1)^T$,

$$A = \begin{bmatrix} -2 & 2 & 2 \\ 1 & 3 & 4 \end{bmatrix}, \quad b = \begin{bmatrix} 0 \\ 0 \end{bmatrix}.$$

Тогда $f(x^1, A, b) = (0, 1)$, $f(x^2, A, b) = f(x^3, A, b) = (1, 1)$, $R^2(A, b) = P^2(A, b) = \{x^1\}$. Поэтому на основании теоремы задача $Z^2(A, b)$ устойчива. В тоже время $S^2(A, b) = X$, т. е. $P^2(A, b) \neq S^2(A, b)$.

Отметим, что критерий квазиустойчивости задачи $Z^m(A, b)$ (или иначе — дискретного аналога полунепрерывности снизу по Хаусдорфу многозначного отображения, задающего паретовскую функцию выбора) анонсирован в [14].

Список литературы

- [1] Белоусов Е. Г., Андронов В. Г. *Разрешимость и устойчивость задач полиномиального программирования*. – М.: Изд-во МГУ, 1993, 172 с.
- [2] Сергиенко И. В., Козерацкая Л. Н., Лебедева Т. Т. *Исследование устойчивости и параметрический анализ дискретных оптимизационных задач*. Киев: Наукова думка, 1995, 170 с.
- [3] Козерацкая Л. Н., Лебедева Т. Т., Сергиенко И. В. *Задачи целочисленного программирования с векторным критерием: параметрический анализ и исследование устойчивости* – Докл. АН СССР, 1989, т. 307, № 3, с. 527–529.
- [4] Емеличев В. А., Никулин Ю. В. *О двух типах устойчивости векторной линейно квадратичной задачи булева программирования* – Дискрет. анализ и исслед. операций, сер. 2, 1999, т. 6, № 2, с. 23–31.
- [5] Подкопаев Д. П. *Условия устойчивости множества Парето векторной траекторной задачи* – Вестн. Белорус. ун-та, сер. 1, 1998, № 3, с. 65–68.
- [6] Емеличев В. А., Кузьмин К. Г., Леонович А. М. *Устойчивость в векторных комбинаторных задачах оптимизации* – Автоматика и телемеханика, 2004, № 2, с. 79–92.
- [7] Лебедева Т. Т., Сергиенко Т. И. *Сравнительный анализ различных типов устойчивости по ограничениям векторной задачи целочисленной оптимизации* – Кибернетика и системный анализ, 2004, № 1, с. 63–70.
- [8] Зуев Ю. А. *Пороговые функции и пороговые представления булевых функций* – Матем. вопросы кибернетики, вып. 5, 1994, с. 5–61.
- [9] Коршунов А. Д. *Монотонные булевы функции* – Успехи математических наук, 2003, т. 58, вып. 5, с. 89–162.
- [10] Емеличев В. А., Мельников О. И., Сарванов В. И., Тышкевич Р. И. *Лекции по теории графов*. М.: Наука, 1990, 384 с.
- [11] Леонтьев В. К. *Устойчивость в линейных дискретных задачах* – М. Наука, 1979, вып. 35, с. 169–184.
- [12] Бухтояров С. Е., Емеличев В. А., Степанишина Ю. В. *Вопросы устойчивости векторных дискретных задач с параметрическим принципом оптимальности* – Кибернетика и системный анализ, 2003, № 4, с. 155–166.
- [13] Emelichev V. A., Girlich E., Nikulin Yu. V., Podkopaev D. P. *Stability and regularization of vector problems of integer linear programming* – Optimization, 2002, v. 51, N 4, p. 645–676.
- [14] Емеличев В. А., Кузьмин К. Г. *Об одном виде устойчивости векторной задачи минимизации пороговых функций* – Материалы международной конференции «Алгебра, логика и кибернетика», Иркутск, Издательство ГОУ ВПО ИГУ, 2004, с. 147–149.

ON STABILITY OF THE VECTOR BOOLEAN PROBLEM OF THRESHOLD FUNCTION MINIMIZATION

V. A. Emelichev, K. G. Kuzmin

Belarusian State University, Minsk
e-mail: emelichev@bsu.by, kuzminkg@mail.ru

Abstract. Criteria of stability of vector Boolean problem of threshold functions minimization is obtained.

Key words: threshold functions, vector problem, Pareto set, stability.

МЕТОД ЭКОНОМНОГО ОПИСАНИЯ ПОЛИЭДРАЛЬНОЙ АППРОКСИМАЦИИ ВЫПУКЛОГО МНОЖЕСТВА В ЗАДАЧЕ МНОГОКРИТЕРИАЛЬНОЙ ОПТИМИЗАЦИИ ¹

Р.В. Ефремов, Г.К. Каменев, А.В. Лотов

*Вычислительный центр имени А.А. Дородницына Российской академии наук, Москва
e-mails: efremor@ccas.ru, kamenev@ccas.ru, lotov1@ccas.ru*

Аннотация. Метод достижимых целей / Диалоговых карт решений (МДЦ/ДКР) является методом многокритериальной оптимизации, в котором пользователь изучает паретову границу множества достижимых критериальных векторов и получает визуальную поддержку в процессе идентификации целевой точки. Метод основан на аппроксимации оболочки Эджворта-Парето (ОЭП), то есть максимального множества, имеющего ту же паретову границу, что и множество достижимых критериальных векторов. Исследование с помощью МДЦ/ДКР можно разбить на два этапа - этап аппроксимации ОЭП в автоматическом режиме и этап интерактивной визуализации сечений ОЭП. В рамках выпуклых задач, ОЭП аппроксимируется выпуклым многогранным телом. На практике используется метод Уточнения оценки (УО), который аппроксимирует ОЭП многогранным телом с небольшим числом вершин. При этом, однако, число гиперграней в аппроксимации может оказаться велико. В то же время, в технологии ДКР, основанной на интерактивной анимации большого числа сечений ОЭП, скорость визуализации сечений падает с ростом числа гиперграней в аппроксимации. В работе предлагается метод Экономного описания полиэдральной аппроксимации (ЭОПА), который предназначен для построения аппроксимации выпуклого компактного тела (ВКТ) многогранником с достаточно небольшим числом гиперграней. Метод ЭОПА основан на использовании метода УО и двойственных методов аппроксимации ВКТ.

Ключевые слова: многокритериальная оптимизация, полиэдральная аппроксимация, визуализация, диалоговые карты решений, паретова граница.

1. Постановка проблемы

В процессе поиска эффективных решений в сложных задачах принятия решений все более важную роль играют методы многокритериальной оптимизации, позволяющие учесть противоречивые требования, предъявляемые к рассматриваемым решениям. Одним из таких методов является Метод достижимых целей / Диалоговых карт решений (МДЦ/ДКР) [1, 2], предназначенный для визуального изучения паретовой границы множества достижимых критериальных векторов выбора решений и прямого указания достижимой целевой точки на дисплее при числе критериев от трех до семи-восьми. Визуализация паретовой границы в МДЦ/ДКР осуществляется на основе ее представления пользователю в виде карт решений – специально отобранных совокупностей двумерных сечений так назы-

¹Работа выполнена при финансовой поддержке Российского фонда фундаментальных исследований (код проекта 04-01-00662), по программе государственной поддержки ведущих научных школ (код проекта НШ -1843.2003.1), при поддержке программы №3 фундаментальных исследований ОМН РАН “Вычислительные и информационные проблемы решения больших задач”, при поддержке программы №16 фундаментальных исследований РАН, проект №2 “Математическое моделирование и интеллектуальные системы”

ваемой оболочки Эджворта-Парето (ОЭП), то есть максимального множества, имеющего ту же паретову границу, что и множество достижимых критериальных векторов. Благодаря применению МДЦ/ДКР пользователь получает информацию о возможных значениях критериев и, что особенно важно, об эффективных (критериальных) замещениях. Эта информация способствует пониманию взаимосвязей между критериями и осознанному выбору достижимой цели.

Важной особенностью МДЦ/ДКР является разбиения исследования на два этапа: этап аппроксимации ОЭП в автоматическом режиме и этап визуализации паретовой границы сечений ОЭП в диалоге с пользователем, включающей анимацию карт решений. В данной работе мы концентрируемся на задаче аппроксимации ОЭП в выпуклом случае (методы для случая невыпуклого ОЭП описаны в [2, 3]). Для аппроксимации ОЭП в выпуклом, в том числе и в линейном случае, в рамках МДЦ используются итеративные методы полиэдральной аппроксимации. В рассматриваемом случае ОЭП является геометрической суммой выпуклого компактного множества и заданного конуса, поэтому анализ эффективности методов аппроксимации ОЭП сводятся к вопросу об эффективности методов аппроксимации выпуклых компактных тел (ВКТ). В наших исследованиях мы используем адаптивные итеративные методы аппроксимации ВКТ многогранниками, которые строятся с помощью расчета значений опорной функции аппроксимируемого множества. Наиболее часто используется метод Уточнения оценки (УО), до недавнего времени являвшийся единственным адаптивным методом полиэдральной аппроксимации ВКТ, используемым на практике.

Теоретические исследования [4] показали, что при применении метода УО для аппроксимации ВКТ с гладкой границей полученные многогранники имеют асимптотически оптимальное число вершин. Опыт использования метода УО в прикладных задачах [1, 2] показывает, что при аппроксимации многогранных тел хорошая точность аппроксимации достигается уже при достаточно малом числе вершин аппроксимирующего многогранника. Вместе с тем, число гиперграней аппроксимирующего многогранника может оказаться велико. Это существенный недостаток метода, поскольку практическое его применение связано с диалоговой визуализацией и анимацией сечений построенной аппроксимации. Дело в том, что технология ДКР, которая с помощью интерактивной анимации цветных карт решений позволяет изобразить Паретову границу в наглядной форме, требует быстрой и стабильной работы программного обеспечения, а скорость визуализации падает с ростом числа гиперграней в визуализируемом многограннике.

В [5] предложен метод Экономного описания многогранника (ЭОМ), который позволяет избавляться от тех гиперграней в описании многогранника, исключение которых не приводит к существенному ухудшению точности аппроксимации. Он основан на использовании понятия двойственности ВКТ и концепций теории двойственных методов аппроксимации ВКТ [6].

Метод Экономного описания полиэдральной аппроксимации (ЭОПА), предлагаемый в данной работе, предназначен для построения аппроксимации ВКТ многогранником с относительно небольшим числом гиперграней, в пространстве размерности порядка семи-десяти. Метод ЭОПА представляет собой синтез методов УО и ЭОМ, а именно, в нем для построения предварительной аппроксимации некоторого ВКТ многогранником P_ϵ с относительно большим числом гиперграней используется метод УО, а затем многогранник P_ϵ аппроксимируется методом ЭОМ.

2. Описание метода

В дальнейшем изложении нам понадобятся некоторые понятия теории двойственности ВКТ. Обозначим через C_0 класс ВКТ, для элементов которого начало координат является внутренней точкой. Для ВКТ $C \in C_0$ определим *полярное множество* C^* относительно начала координат в \mathbb{R}^n , как множество

$$\{x \in \mathbb{R}^n: \langle x, y \rangle \leq 1, y \in C\},$$

где $\langle x, y \rangle = \sum_{i=1}^n x_i y_i$ – скалярное произведение в \mathbb{R}^n . Тогда $C^* \in C_0$ и $C^{**} = (C^*)^* = C$.

Если многогранник P является вписанным в ВКТ C из C_0 (вершины P лежат на границе C), то сопряженный многогранник P^* является описанным вокруг сопряженного тела C^* (гиперграни P^* касаются C^*).

Будем считать, что аппроксимируемое ВКТ $C \in C_0$. В предлагаемом методе тело C вначале аппроксимируется с помощью метода УО. Приведем его краткое описание в контексте метода ЭОПА. В методе УО сначала строится симплекс P_0 , вершины которого лежат на границе C , а затем на основе расчета опорной функции C для адаптивно выбираемых направлений находится последовательность вписанных многогранников P^k с растущим числом вершин. На k -й итерации должен быть построен вписанный в тело C многогранник P^k . Многогранники заданы в виде так называемого двойного описания. Во-первых, каждый многогранник задан множеством решений системы линейных неравенств. Таким образом, задано множество единичных внешних нормалей к гиперграням многогранника P^k , которое обозначим через $U(P^k)$. Во-вторых, каждый многогранник задан списком вершин, для которых указана принадлежность к гиперграням многогранника. Множество вершин многогранника P^k обозначим через $M(P^k)$. Рассмотрим $k+1$ -ую итерацию метода УО. На первом шаге для каждого направления $u \in U(P^k)$ рассчитываем опорную функцию тела C и выбираем то направление, для которого величина $g(u, C) - g(u, P^k)$ максимальна. Отметим, что в процессе расчета опорной функции находятся и точки p границы тела C , такие что $\langle u, p \rangle = g(u, C)$.

В методе УО процесс аппроксимации обычно не доводится до конца, он прекращается при достижении некоторой заранее заданной точности. Для исследования точности аппроксимации тела C многогранником P понадобится функция [4]:

$$\delta_P(C_1, C_2) := \max\{|g(u, C_1) - g(u, C_2)| : u \in U(P)\}, \quad (1)$$

где C_1, C_2 – произвольные ВКТ. Функция (1) является псевдометрикой (т.е. симметричная, неотрицательная, $\delta_P(C, C) = 0$ для любого ВКТ C и выполняется неравенство треугольника).

Под точностью при аппроксимации ВКТ C многогранником P в методе будем понимать величину $\delta_P(C, P)$. Отметим, что эта точность может быть пересчитана в расстояние между P и C в метрике Хаусдорфа [4].

Пусть P_ε – многогранник внутренней аппроксимации тела C , обеспечивающий требуемую точность $\varepsilon > 0$, то есть $\delta_{P_\varepsilon}(C, P_\varepsilon) \leq \varepsilon$. Точность аппроксимации выбирается такой, что $P_\varepsilon \in C_0$. Тогда к многограннику P_ε может быть применен метод ЭОМ.

В методе ЭОМ сопряженный к P_ε многогранник P_ε^* аппроксимируется с точностью ε методом УО. Вершины P_ε^* задаются гипергранями P_ε . Заметим, что в данном случае, когда многогранник P_ε^* задан множеством своих вершин $M(P_\varepsilon^*)$, его опорная функция для направления u , принадлежащего единичной сфере в \mathbb{R}^n , определяется как

$$g(u, P_\varepsilon^*) = \max\{\langle u, p \rangle : p \in M(P_\varepsilon^*)\}.$$

Пусть Q_ε^* – многогранник внутренней аппроксимации тела P_ε^* , обеспечивающий требуемую точность, то есть $\delta_{Q_\varepsilon^*}(P_\varepsilon^*, Q_\varepsilon^*) \leq \varepsilon$. Опять же предполагается, что точность выбрана так, что выполняется $Q_\varepsilon^* \in C_0$. Многогранник Q_ε^* содержит сравнительно небольшое число вершин. Тогда сопряженный многогранник $Q_\varepsilon := (Q_\varepsilon^*)^*$ содержит небольшое число гиперграней, которые задаются вершинами Q_ε^* . Многогранник Q_ε и является искомой аппроксимацией P_ε в методе ЭОМ. Легко видеть, что многогранник Q_ε получен из многогранника P_ε выбрасыванием из описания P_ε некоторых граней. Многогранник Q_ε является также и искомой аппроксимацией тела C в методе ЭОПА.

3. Точность аппроксимации

В [5] доказано следующее утверждение о точности аппроксимации методом ЭОМ.

Утверждение. Пусть дан исходный многогранник P . Пусть $\varepsilon \geq 0$ – точность аппроксимации многогранника P^* внутренним многогранником P_ε^* . Тогда точность δ аппроксимации P внешним многогранником Q_ε такова, что

$$\delta \leq (d_{max}/d_{min}) \cdot \varepsilon,$$

где d_{max} – максимальное среди значений правых частей в описании исходного многогранника P в виде системы линейных неравенств, а d_{min} – минимальное среди значений правых частей в описании многогранника P_ε^* .

Замечание. В процессе аппроксимации многогранника P^* последовательностью $\{P_k^*\}_{k=0,1,\dots}$ значение выражения $(d_{max}/d_{min}) \cdot \varepsilon$ известно после каждой итерации и монотонно не возрастает. Таким образом, точность аппроксимации многогранника P многогранником $(P_k^*)^*$ контролируется из процесса аппроксимации сопряженного многогранника P^* и не становится хуже.

Теорема. Пусть методом УО для тела C получена аппроксимация многогранником P_ε , причем $\delta_{P_\varepsilon}(C, P_\varepsilon) \leq \varepsilon$. Пусть многогранник P_ε аппроксимирован методом ЭОМ многогранником Q_ε и $\delta_{P_\varepsilon}(P_\varepsilon, Q_\varepsilon) \leq \varepsilon$. Тогда

$$\delta_{Q_\varepsilon}(C, Q_\varepsilon) \leq \delta_{P_\varepsilon}(C, Q_\varepsilon) \leq \varepsilon. \quad (2)$$

Кроме того, для расстояния $h(C, Q_\varepsilon)$ между C и Q_ε в метрике Хаусдорфа, справедлива оценка

$$h(C, Q_\varepsilon) \leq \alpha(P_\varepsilon) \cdot \varepsilon, \quad (3)$$

где $\alpha(P_\varepsilon)$ – асферичность P_ε .

Доказательство. Из того, что $P_\varepsilon \subseteq C$ и $\delta_{P_\varepsilon}(C, P_\varepsilon) \leq \varepsilon$ следует, что для любого направления $u \in U(P_\varepsilon)$ выполняется

$$g(u, P_\varepsilon) \leq g(u, C) \leq g(u, P_\varepsilon) + \varepsilon. \quad (4)$$

Точно так же из $P_\varepsilon \subseteq Q_\varepsilon$ и $\delta_{P_\varepsilon}(P_\varepsilon, Q_\varepsilon) \leq \varepsilon$ следует, что для любого $u \in U(P_\varepsilon)$

$$g(u, P_\varepsilon) \leq g(u, Q_\varepsilon) \leq g(u, P_\varepsilon) + \varepsilon. \quad (5)$$

Из (4), (5) и определения (1) следует правое неравенство в (2):

$$\delta_{P_\varepsilon}(C, Q_\varepsilon) \leq \varepsilon.$$

Из того, что $U(Q_\varepsilon) \subseteq U(P_\varepsilon)$ следует левое неравенство в (2).

В [4] доказано, что для произвольного ВКТ C и вписанного в него многогранника P связь между точностью $\delta_P(C, P)$ аппроксимации тела C многогранником P и расстоянием $h(P, C)$ между P и C в метрике Хаусдорфа выражается формулой:

$$h(P, C) \leq \alpha(P) \cdot \delta_P(C, P),$$

где $\alpha(P)$ – асферичность P . Отсюда и из (2) сразу следует (3).

Итак многогранник Q_ε аппроксимирует исходное выпуклое тело с той же точностью ε , что и P_ε , и имеет не большее, чем P_ε , число граней.

4. Экспериментальное применение метода

Для исследования возможностей метода ЭОПА он был использован при аппроксимации так называемых циклических многогранников. Под циклическим многогранником в \mathbb{R}^n понимается выпуклая оболочка точек, заданных параметрически формулой $x(t) := (t, t^2, \dots, t^n)$, где параметр $t > 0$ принимает несовпадающие вещественные значения t_1, t_2, \dots, t_M . Известно, что все точки, полученные таким образом, являются вершинами циклического многогранника, причем число его граней задается простой формулой ([7], теоремы 13.4 и 13.5). Был рассмотрен четырехмерный циклический многогранник C_1 с 1000 вершинами, имеющий 498500 граней. Так как этот многогранник задан сравнительно небольшим числом своих вершин, то может быть легко вычислена его опорная функция и для аппроксимации использован метод УО. Однако в нашем исследовании в качестве объекта аппроксимации необходим многогранник, заданный очень большим числом гиперграней. Поэтому этот циклический многогранник был предварительно аппроксимирован многогранником C_2 , состоящим из 186302 граней, с помощью метода Quickhull [8].

Заметим, что для того, чтобы применить метод ЭОПА к построенному многограннику, необходимо было сдвинуть его так, чтобы он содержал в своей внутренности начало координат. Сдвиг был выбран так, чтобы начало координат совпало с центром тяжести множества вершин. Кроме того, полученный многогранник обладал большой асферичностью, которая была устранена соответствующим линейным преобразованием в \mathbb{R}^n , сохраняющим число гиперграней многогранника (сжатие вдоль оси, соединяющей начало координат и максимально удаленную от него вершину сопряженного многогранника).

К полученному в результате этих операций многограннику C_3 был применен метод ЭОПА, который позволил построить многогранник Q_ε , который аппроксимирует многогранник C_3 с относительной точностью 1%. Здесь под относительной точностью мы понимаем выражение $(\delta_{P_\varepsilon}(C, Q_\varepsilon)/D) \cdot 100\%$, где D – диаметр многогранника C_3 : $D := \max\{\rho(x, y) : x, y \in M(C_3)\}$. Многогранник C_3 состоит из 383 граней. Построенный методом УО многогранник P_ε , аппроксимирующий C_3 с той же относительной точностью $(\delta_{P_\varepsilon}(C, P_\varepsilon)/D) \cdot 100\% = 1\%$, содержит 2269 граней и 109 вершин.

Список литературы

- [1] Лотов А.В., Бушенков В.А., Каменев Г.К., Черных О.Л. *Компьютер и поиск компромисса. Метод достижимых целей*. М.: Наука, 1997.
- [2] Lotov A.V., Bushenkov V.A., and Kamenev G.K. *Interactive Decision Maps. Approximation and Visualization of Pareto Frontier*. Kluwer Academic Publishers, 2004.
- [3] Лотов А.В., Каменев Г.К., Березкин В.Е. *Аппроксимация и визуализация паретовой границы для невыпуклых многокритериальных задач // Доклады АН*. 2002. Т.386, № 6, С. 738-741.
- [4] Каменев Г.К. *Исследование одного алгоритма аппроксимации выпуклых тел // Ж. вычисл. матем. и матем. физ.* 1994. Т.34, № 4, С. 608-616.
- [5] Ефремов Р.В., Каменев Г.К., Лотов А.В. *Построение экономного описания многогранника на основе теории двойственности выпуклых множеств // Доклады АН*. 2004. Т.399, №5, С. 594-596.
- [6] Каменев Г.К. *Сопряженные адаптивные алгоритмы полиэдральной аппроксимации выпуклых тел // Ж. вычисл. матем. и матем. физ.* 2002. 42(9), С. 1351-1367.
- [7] Бренстед А. *Введение в теорию выпуклых многогранников*. М. Мир, 1988.
- [8] Barber, C.B., Dobkin, D.P., and Huhdanpaa, H.T. *The Quickhull algorithm for convex hulls // ACM Trans. on Mathematical Software*. 1996. 22(4), P. 469-483.

METHOD FOR ECONOMICAL DESCRIPTION OF POLYHEDRAL APPROXIMATIONS OF CONVEX BODIES IN MULTI-OBJECTIVE OPTIMIZATION PROBLEMS

R.V.Efremov, G.K.Kamenev, A.V.Lotov

Dorodnicyn Computing Centre of the Russian Academy of Sciences, Moscow
e-mails: efremor@ccas.ru, kamenev@ccas.ru, lotov1@ccas.ru

Abstract. The Feasible Goals Method / Interactive Decision Maps (FGS/IDM) is the technique for multiple criteria optimization that was designed for studying the Pareto frontier of the feasible criterion set and supporting the user in the process of identification of the goal. The FGM/IDM technique consists of two stages. The first stage is provided by the approximation of the Edgeworth-Pareto Hull (EPH) of the feasible criterion set, that is, the maximal set that has the same Pareto frontier as the feasible criterion set. The second stage consists in an interactive visualization of slices of the EPH provided in the form of decision maps. For the problems with a convex EPH, it is approximated by a polyhedral convex body. In applications, the Estimate Refinement (ER) method is used that generates the polyhedral approximation of an EPH with a sufficiently small number of vertices; however, the number of its facets could be large. At the same time, the IDM technique, which is based on interactive animation of numerous slices of the EPH, requires an approximation with a relatively small number of facets. In this paper, the method for Economical Description of Polyhedral Approximations (EDPA) is described. It is designed for constructing approximations of a convex compact body (CCB) by a polyhedron with a sufficiently small number of facets. The EDPA method integrates the ER method and concepts of the duality theory of methods for approximation of CCB.

Key words: multiple criteria optimization, polyhedral approximation, visualization, interactive decision maps, Pareto frontier

ОБ УСТОЙЧИВОСТИ ВЕКТОРНОЙ ТРАЕКТОРНОЙ ЗАДАЧИ С ОБОБЩЕННЫМИ НЕЛИНЕЙНЫМИ ЧАСТНЫМИ КРИТЕРИЯМИ

А.М. Леонович

Белорусский государственный университет, Минск
e-mail: leonovich@tut.by

Аннотация. В работе рассматривается векторная траекторная задача поиска множества Парето с обобщенными нелинейными частными критериями, частными случаями которых являются такие широко известные критерии, как линейный и "узкого места". Получены два необходимые и одновременно достаточные условия для того типа устойчивости задачи, который является дискретным аналогом свойства полунепрерывности сверху по Хаусдорфу оптимального отображения, ставящего в соответствие каждому набору параметров задачи множество искомых решений.

Ключевые слова: устойчивость, множество Парето, траекторная задача, частный критерий.

1. Основные определения

Рассмотрим следующую модель векторной (n -критериальной) траекторной задачи. Пусть задано множество $E = \{e_1, e_2, \dots, e_m\}$, $m \geq 2$, и система непустых подмножеств (траекторий) $T \subseteq 2^E \setminus \{\emptyset\}$, $|T| \geq 2$. Пусть частными критериями вектор-функции

$$f(t, A) = (f_1(t, A_1, k_1), f_2(t, A_2, k_2), \dots, f_n(t, A_n, k_n)), \quad n \geq 1,$$

заданной на множестве траекторий T , являются критерии следующих двух видов

$$\Sigma\text{-MINMAX} : f_i(t, A_i, k_i) = \max\{g_i(s, A_i) : s \in S(t, k_i)\} \longrightarrow \min_{t \in T}, \quad i \in I_1, \quad (1)$$

$$\Sigma\text{-MINMIN} : f_i(t, A_i, k_i) = \min\{g_i(s, A_i) : s \in S(t, k_i)\} \longrightarrow \min_{t \in T}, \quad i \in I_2, \quad (2)$$

где

$$g_i(s, A_i) = \sum_{j \in N(s)} a_{ij},$$

$$S(t, k) = \{s \subseteq t : |s| = \min\{|t|, k\}\},$$

$$N(s) = \{j \in N_m : e_j \in s\},$$

$$N_m = \{1, 2, \dots, m\},$$

$$I_1 \cup I_2 = N_n, \quad I_1 \cap I_2 = \emptyset,$$

A_i — i -я строка матрицы $A = [a_{ij}] \in \mathbf{R}^{n \times m}$, k_1, k_2, \dots, k_n — заданные числа из множества N_m . Будем считать, что $g_i(\emptyset, A_i) = 0$, $S(t, 0) = \emptyset$. Для любого непустого множества $p \subseteq E$ и числа k будем использовать прежнее обозначение $f_i(p, A_i, k)$, полагая лишь $f_i(p, A_i, k) = 0$, если $k = 0$.

Легко видеть, что критерии (1) и (2) при $k_i \geq \max\{|t| : t \in T\}$ превращаются в линейный критерий

$$\text{MINSUM :} \quad f_i(t, A_i, k_i) = \sum_{j \in N(t)} a_{ij} \longrightarrow \min_{t \in T},$$

критерий (1) при $k_i = 1$ превращается в критерий «узкого места»

$$\text{MINMAX :} \quad f_i(t, A_i, k_i) = \max_{j \in N(t)} a_{ij} \longrightarrow \min_{t \in T},$$

а критерий (2) при $k_i = 1$ — в критерий

$$\text{MINMIN :} \quad f_i(t, A_i, k_i) = \min_{j \in N(t)} a_{ij} \longrightarrow \min_{t \in T}.$$

Множество тех индексов из N_n , которыми занумерованы частные критерии вида MINSUM, MINMAX и MINMIN, будем обозначать соответственно через I_{SUM} , I_{MAX} и I_{MIN} .

Под векторной (n -критериальной) траекторной задачей $Z^n(A)$, $n \geq 1$, будем понимать задачу поиска множества Парето $P^n(A)$, состоящего из всех эффективных траекторий:

$$P^n(A) = \{ t \in T : P^n(t, A) = \emptyset \},$$

где

$$P^n(t, A) = \{ t' \in T : f(t, A) \geq f(t', A), f(t, A) \neq f(t', A) \}.$$

Очевидно, что $P^1(A)$, где A — m -мерный вектор, является множеством всех оптимальных решений скалярной траекторной задачи $Z^1(A)$. В схему таких задач вкладываются многие широко известные задачи оптимизации на графах, задачи булева программирования, разнообразные задачи теории расписаний.

Следуя [2, 2, 3, 4], задачу $Z^n(A)$ назовем устойчивой, если выполняется формула

$$\exists \varepsilon > 0 \forall B \in \mathcal{B}(\varepsilon) (P^n(A) \supseteq P^n(A + B)),$$

где

$$\mathcal{B}(\varepsilon) = \{ B \in \mathbf{R}^{n \times m} : \|B\| < \varepsilon \},$$

$$\|B\| = \max\{ |b_{ij}| : (i, j) \in N_n \times N_m \}, \quad B = [b_{ij}] \in \mathbf{R}^{n \times m}.$$

Легко понять, что свойство устойчивости задачи $Z^n(A)$ является дискретным аналогом свойства полунепрерывности сверху в смысле Хаусдорфа в точке $A \in \mathbf{R}^{n \times m}$ оптимального отображения

$$P^n : \mathbf{R}^{n \times m} \rightarrow 2^E,$$

т. е. точно-множественного (многозначного) отображения, которое каждому набору параметров (каждой матрице A) ставит в соответствие множество Парето $P^n(A)$ (см., например, [1]).

Очевидно, что при выполнении равенства $P^n(A) = T$, задача $Z^n(A)$ устойчива. Поэтому в дальнейшем этот случай будем исключать из рассмотрения, а задачу $Z^n(A)$, для которой множество $\bar{P}^n(A) = T \setminus P^n(A)$ не пусто, будем называть нетривиальной.

Для всякой $t \in T$ и индекса $i \in N_n$ будем обозначать

$$W_i(t, A) = \begin{cases} \{t' \in T \setminus \{t\} : U_i(t', A_i, k_i) \subseteq U_i(t, A_i, k_i)\}, & \text{если } i \in I_1, \\ \{t' \in T \setminus \{t\} : U_i(t, A_i, k_i) \subseteq U_i(t', A_i, k_i)\}, & \text{если } i \in I_2, \end{cases}$$

где

$$U_i(t, A_i, k) = \begin{cases} \text{Arg max}\{g_i(s, A_i) : s \in S(t, k)\}, & \text{если } i \in I_1, \\ \text{Arg min}\{g_i(s, A_i) : s \in S(t, k)\}, & \text{если } i \in I_2, \end{cases}$$

Для любых $i \in N_n, t \in T, t' \in T \setminus \{t\}$ положим

$$\zeta_i(t, t', A_i) = \begin{cases} f_i(t, A_i, k_i) - f_i(t', A_i, k_i), & \text{если } |t \cap t'| < k_i, \\ f_i(t, A_i, k_i) - f_i(t, A_i, k_i - 1) - f_i(t' \setminus t, A_i, 1), & \text{если } |t \cap t'| \geq k_i, i \in I_1, \\ f_i(t \setminus t', A_i, 1) + f_i(t', A_i, k_i - 1) - f_i(t', A_i, k_i), & \text{если } |t \cap t'| \geq k_i, i \in I_2. \end{cases}$$

При этом будем считать, что

$$f_i(\emptyset, A_i, k_i) = \begin{cases} -\infty, & \text{если } i \in I_1, \\ +\infty, & \text{если } i \in I_2. \end{cases}$$

2. Теорема и следствия

Теорема. Для нетривиальной векторной траекторной задачи $Z^n(A)$, $n \geq 1$, с любой комбинацией частных критериев (1) и (2) следующие утверждения эквивалентны:

1⁰ задача $Z^n(A)$ устойчива;

2⁰ $\forall t \in \bar{P}^n(A) \exists t' \in T \setminus \{t\} \forall i \in N_n$

$$f_i(t, A_i, k_i) = f_i(t', A_i, k_i) \implies t' \in W_i(t, A_i);$$

3⁰ $\forall t \in \bar{P}^n(A) \exists t' \in T \setminus \{t\} \forall i \in N_n$

$$f_i(t, A_i, k_i) = f_i(t', A_i, k_i) \implies \zeta_i(t, t', A_i) > 0.$$

Аналогичный результат был получен ранее в [3] для для векторной траекторной l_∞ -экстремальной задачи.

Множество слабо эффективных траекторий (множество Слейтера) определим следующим образом

$$Q^n(A) = \{t \in T : Q^n(t, A) = \emptyset\},$$

где

$$Q^n(t, A) = \{t' \in T : f(t, A) > f(t', A)\}.$$

Пусть I_{SUM} — множество тех индексов из N_n , которыми занумерованы соответственно частные критерии MINSUM.

Следствие 1 [4]. Для того чтобы нетривиальная задача $Z^n(A)$, $n \geq 1$, с частными критериями вида (1) и (2) была устойчива, достаточно, а в случае, когда $I_{SUM} = N_n$, и необходимо, чтобы выполнялось равенство

$$P^n(A) = Q^n(A).$$

Следствие 2 [4]. Однокритериальная задача $Z^1(A)$ устойчива при любом векторе $A \in \mathbf{R}^m$.

Список литературы

- [1] И.В. Сергиенко, Л.Н. Козерацкая, Т.Т. Лебедева *Исследование устойчивости и параметрический анализ дискретных оптимизационных задач*. Киев: Наук. думка, 1995, 169 с.
- [2] V.A. Emelichev, E. Girlich, Yu.V. Nikulin, D.P. Podkopaev *Stability and regularization of vector problem of integer linear programming* - Optimization, 2002, v. 51, N 4, p. 645-676.
- [3] В. А. Емеличев, К. Г. Кузьмин, А. М. Леонович *Устойчивость в векторных комбинаторных задачах оптимизации* - Автоматика и телемеханика, 2004, N 2, с. 79-92.
- [4] Э. Гирлих, М.Н. Ковалев, М.К. Кравцов *Стабильность, устойчивость и квазиустойчивость многокритериальной задачи на системе подмножеств* - Кибернетика и системный анализ, 1999, N 5, с. 111-124.

ON STABILITY OF VECTOR TRAJECTORIAL PROBLEM WITH GENERALIZED NONLINEAR PARTIAL CRITERIA

A.M. Leonovich

Belorussian State University, Minsk

e-mail: leonovich@tut.by

Abstract. We consider a vector trajectorial problem of finding of Pareto set. The components of vector criterion are of generalized nonlinear kind. Well-known bottleneck criterion and linear criterion are partial cases of considered partial criteria. Under stability of vector trajectorial problem we understand a discrete analogue of upper semicontinuity property of many-valued mapping. This mapping assigns the Pareto set to any set of problem parameters. Two necessary and at the same time sufficient conditions for stability of vector combinatorial problem are provided.

Key words: stability, Pareto set, trajectorial problem, partial criterion.

АЛГОРИТМ АППРОКСИМАЦИИ ВЫПУКЛОЙ ОБОЛОЧКИ ЭДЖВОРТА-ПАРЕТО В НЕКОТОРЫХ ЗАДАЧАХ ЦЕЛОЧИСЛЕННОЙ МНОГОКРИТЕРИАЛЬНОЙ ОПТИМИЗАЦИИ ¹

А.И. Поспелов

ВМК МГУ, Москва, e-mail: alexispospelov@yandex.ru

Аннотация. В статье рассматриваются задачи целочисленной многокритериальной оптимизации. Для целочисленных многокритериальных задач с целевыми функциями из специального класса предложен и обоснован алгоритм полиэдральной аппроксимации выпуклой оболочки Эджворта-Парето. Алгоритм основан на методе ветвей и границ и позволяет строить внешнюю и внутреннюю полиэдральные аппроксимации выпуклой оболочки Эджворта-Парето таких задач с любой заданной степенью точности.

Ключевые слова: многокритериальная дискретная оптимизация, выпуклая оболочка Эджворта-Парето, паретова граница

Введение

При поиске эффективных решений существенным является многостороннее изучение проблемы. В связи этим важную роль играют методы многокритериальной оптимизации, позволяющие учесть противоречивые требования предъявляемые системе. Общая формулировка задачи многокритериальной оптимизации имеет следующий вид:

$$f(x) \rightarrow \max_{x \in X},$$

где

$$f(X) = Y \subset \mathbb{R}^d.$$

Большинство методов многокритериальной оптимизации основано на взаимодействии с лицом, принимающим решение (ЛПР), и поиске подходящими альтернативами для ЛПР на недоминируемой (паретовой) границе $P(Y)$, то есть множестве всех недоминируемых альтернативы

$$P(Y) = \{y \in Y \mid \{y' \in Y \mid y' \leq y, y' \neq y\} = \emptyset\},$$

и представление найденных альтернатив ЛПР. Одним из таких методов является Метод Достижимых Целей (МДЦ)[1],[2]. Решение задачи многокритериальной оптимизации в рамках МДЦ можно разделить на два основных этапа. Первый этап заключается в аппроксимации оболочки Эджворта-Парето (ОЭП)

$$Y_P = Y + \mathbb{R}_+^d.$$

ОЭП является максимальным по включению множеством, недоминируемая граница, которого совпадает с исходным множеством. Второй этап состоит в визуализации паретовой

¹Работа выполнена при финансовой поддержке Российского фонда фундаментальных исследований (код проекта 04-01-00662), по программе государственной поддержки ведущих научных школ (код проекта НШ -1843.2003.1), при поддержке программы №3 фундаментальных исследований ОМН РАН "Вычислительные и информационные проблемы решения больших задач", при поддержке программы №16 фундаментальных исследований РАН, проект №2 "Математическое моделирование и интеллектуальные системы"

границы на основе полученной аппроксимации в виде диалоговых карт решений. На основе визуального представления недоминируемой границы ЛПР анализирует возможные решения проблемы и выбирает наиболее подходящее решение.

На практике часто приходится сталкиваться с ситуацией, когда множество стратегий X дискретно. Для данного типа задач в рамках МДЦ был разработан специальный подход — метод разумных целей (МРЦ)[4]. В рамках МДЦ осуществляется аппроксимация и визуализация выпуклой оболочки Эджворта-Парето (ВОЭП)

$$Y_P^C = \text{conv}(Y) + R_+^d.$$

Визуальное изучение ВОЭП позволяет ЛПР исследовать взаимосвязи между критериями. Проанализировав эту связь, ЛПР может выбрать наиболее предпочтительное сочетание значений критериев (цель) на недоминируемой границе ВОЭП. Цель обычно не совпадает ни с одним из критериальных векторов, но в то же время, она близка к совокупности критериальных векторов в целом. Это отличает метод разумных целей от целевых методов, в которых пользователь вынужден выбирать цель без информации о предельных возможностях и связях критериев. Так как обычно целевая точка не совпадает ни с одним из критериальных векторов, то выбираются один или несколько векторов, близких к ней в том или ином смысле.

В смысле вычислительных затрат этап аппроксимации самый трудоемкий в МРЦ, поэтому он требует использования эффективных методов.

Аппроксимация ВОЭП, вообще говоря, может быть осуществлена на основе рассмотрения совокупности всех допустимых точек множества и вычисления опорной функции множества [4]. Такой подход, однако, может быть реализован лишь в случае относительно малого числа вариантов. В данной работе для специального класса целевых функции предложен более эффективный алгоритм аппроксимации ВОЭП для целочисленных задач, использующий идеи, разработанные для решения задач целочисленной оптимизации с одним критерием. Задачи вида

$$f(x) \rightarrow \min, \quad x \in X,$$

где множество X конечно, а f действует из X в R^1 хорошо изучены в дискретном программировании. Общим методом решения подобных задач является метод ветвей и границ, позволяющей уменьшить объем перебора при поиске решения задачи. Под методом ветвей и границ понимается алгоритм решения задачи, имеющий древовидную структуру поиска оптимального решения и использующий результаты решения оценочных задач.

1. Постановка задачи

Рассмотрим следующую многокритериальную задачу целочисленной оптимизации:

$$\begin{aligned} c_i(x) &\rightarrow \min, & i &= \overline{1, N_c}, \\ f_j(x) &\rightarrow \min, & j &= \overline{1, N_f}, \end{aligned}$$

где

$$\begin{aligned} X &= \{x \in X_0 \mid g_k(x) \leq 0, k = \overline{1, N_g}\}, \\ X_0 &= \{0, 1, \dots, K\}^n, \end{aligned}$$

а функции c_i, f_j, g_k , такие что

$$\begin{aligned} c_i : X_0 &\rightarrow \mathbb{R}, & c_i(x') &\geq c_i(x''), & x' &\geq x'', & i &= \overline{1, N_c}; \\ f_j : X_0 &\rightarrow \mathbb{R}, & f_j(x') &\leq f_j(x''), & x' &\geq x'', & j &= \overline{1, N_f}; \\ g_k : X_0 &\rightarrow \mathbb{R}, & g_k(x') &\leq g_k(x''), & x' &\geq x'', & k &= \overline{1, N_g}. \end{aligned}$$

Для данной задачи необходимо аппроксимировать выпуклую оболочку Эджворта-Парето

$$Y_P^C = \text{conv}([c(X), f(X)]) + \mathbb{R}_+^{N_c + N_f}. \quad (1)$$

Здесь и далее под c, f и g понимается векторнозначные функции со значениями в пространствах $\mathbb{R}^{N_c}, \mathbb{R}^{N_f}$ и \mathbb{R}^{N_g} , соответственно. Функции c_i, f_j и g_k совпадают с соответствующими компонентами c, f и g .

Отметим, что внешнюю оценку множества Y_P^C можно построить, исключив условие целочисленности из задачи и построив ВОЭП.

Для некоторых классов многокритериальных задач дискретного программирования может быть указан способ нахождения парето-оптимальных достижимых целей на основе решение серии однокритериальных задач (например, в [5] рассматривается алгоритм построения $P(Y)$ в бикритериальной задаче о ранце). Там же предложен алгоритм ветвей и границ для точного решения задачи о ранце по первому критерию. В данной статье предложен способ аппроксимации ВОЭП задачи целочисленной многокритериальной оптимизации, основанный на методе ветвей и границ.

2. Обозначения

Введем некоторые обозначения.

$$X_m^\alpha = \{x \in X_0 \mid x_i = \alpha_i, i = \overline{1, m}\}, \quad \text{где} \quad \alpha = [\alpha_1, \dots, \alpha_m] \in \{0, 1, \dots, K\}^m.$$

Каждое из множеств X_m^α представляет собой $(n-m)$ -мерную грань целочисленного куба X_0 . Отметим, что

$$X_m^\alpha = \bigcup_{i=0}^K X_{m+1}^{[\alpha, i]}.$$

Далее, пусть

$$x^i(X_m^\alpha) = [a_1, a_2, \dots, a_m, i, \dots, i],$$

Альтернативы $x^0(X_m^\alpha)$ и $x^K(X_m^\alpha)$ являются экстремальными среди других альтернатив X_m^α с точки зрения выбранных критериальных функций. С помощью этих альтернатив можно оценить значения критериев и ограничений на множестве X_m^α . Обозначим эти оценки следующим образом

$$\begin{aligned} \underline{c}(X_m^\alpha) &= c(x^0(X_m^\alpha)), \quad \bar{c}(X_m^\alpha) = c(x^K(X_m^\alpha)), \\ \underline{f}(X_m^\alpha) &= f(x^K(X_m^\alpha)), \quad \bar{f}(X_m^\alpha) = f(x^0(X_m^\alpha)), \\ \underline{g}(X_m^\alpha) &= g(x^K(X_m^\alpha)), \quad \bar{g}(X_m^\alpha) = g(x^0(X_m^\alpha)). \end{aligned}$$

Вектор $\underline{d}(X_m^\alpha) = [\underline{c}(X_m^\alpha), \underline{f}(X_m^\alpha)]$, будем называть нижней оценкой для грани X_m^α в том смысле, что вектор $\underline{d}(X_m^\alpha) \leq [c(x), f(x)]$ для всех $x \in X_m^\alpha$. Каждый из векторов $x^i(X_m^\alpha)$ может как принадлежать, так и не принадлежать X в зависимости от значений ограничений. Обозначим через

$$\Lambda(X_m^\alpha) = \{x^i(X_m^\alpha) \in X, i = 0, \dots, K\}.$$

3. Описание алгоритма

Опишем нулевую и произвольную l -ую итерации алгоритма.

В начале на нулевой итерации для множества X_0 рассчитаем значения оценок

$$\underline{c}(X_0), \bar{c}(X_0), \underline{f}(X_0), \bar{f}(X_0), \underline{g}(X_0), \bar{g}(X_0).$$

Если $\underline{g}(X_0) \not\leq 0$, то $g(x) \not\leq 0$ для всех $x \in X_0$, следовательно, $X = \emptyset$, и выпуклая оболочка Эджворта-Парето

$$Y_P^C = \text{conv}([c(X), f(X)]) + \mathbb{R}_+^{N_c + N_f} = \emptyset,$$

В этом случае работу алгоритма можно считать законченной. Если $\underline{g}(X_0) \leq 0$, то $Y_P^C \neq \emptyset$, так как содержит хотя бы вектор $[c(x^K(X_0)), f(x^K(X_0))]$, тогда на нулевой итерации определим начальное аппроксимирующее многогранное множество P_0 следующим образом

$$P_0 = \text{conv}(\Lambda(X_0)) + \mathbb{R}_+^{N_f + N_c}.$$

Перед переходом к первой итерации проверим, не содержит ли множество P_0 оценку $\underline{d}(X_0)$. Если вдруг оказалось, что $\underline{d}(X_0) \in P_0$, то в этом случае P_0 будет совпадать с Y_P^C и процесс построения будет завершен. В противоположном случае положим $\Xi_0 = \{X_0\}$.

Рассмотрим l -ю итерации. К началу l -й итерации должны быть определены многогранное множество P_{l-1} и набор граней Ξ_{l-1} . Для каждого множества $A \in \Xi_{l-1}$ должны быть вычислены оценки

$$\underline{c}(A), \bar{c}(A), \underline{f}(A), \bar{f}(A), \underline{g}(A), \bar{g}(A),$$

и известно, что

$$\underline{d}(A) \notin P_{l-1}, \underline{g}(A) \leq 0, \Lambda(A) \subset P_{l-1}.$$

Выберем множество $X_m^\alpha \in \Xi_{l-1}$, для которого оценка $\underline{d}(X_m^\alpha)$ наиболее удалена от построенной аппроксимации P_{l-1} , т. е.

$$X_m^\alpha \in \text{Arg} \max_{A \in \Xi_{l-1}} \rho(\underline{d}(A), P_{l-1}), \quad (2)$$

и разобьем на $(K+1)$ новое множество $X_{m+1}^{[\alpha, i]}$, $i = 0 \dots K$, для которых рассчитаем значения оценок. На основе полученных оценок и имеющегося многогранного множества P_{l-1} построим P_l :

$$P_l = \overline{\text{conv} \left(\bigcup_{i=0}^K \Lambda \left(X_{m+1}^{[\alpha, i]} \right) \cup P_{l-1} \right)}.$$

Положим

$$\Xi_l = \left\{ A \in \left\{ X_{m+1}^{[\alpha, s]} \mid \underline{g} \left(X_{m+1}^{[\alpha, s]} \right) \leq 0, i = 0, \dots, K \right\} \cup \Xi_{l-1} \setminus \{X_m^\alpha\} \mid \underline{d}(A) \notin P_l \right\}.$$

Если Ξ_l не пусто, то можно переходить к следующей итерации.

Если в какой-то момент при ветвлении у нас получились одноточечные множества, то они не войдут в Ξ_l , так как для одноточечного множества A выполнено $\underline{d}(A) \in \Lambda(A)$, поэтому ветвление всегда будет корректной операцией.

Если Ξ_l оказалось пусто, то построение закончилось. В этом случае

$$Y_P^C = P_l.$$

Если Ξ_l не пусто алгоритм может продолжать работу. На каждой итерации оценка достигнутой точности можно получена следующим образом

$$\varepsilon_l = \max_{A \in \Xi_l} \rho(\underline{d}(A), P_l) \geq \rho(Y_P^C, P_l).$$

Следует также отметить, что если для какой-то из граней X_m^α оказалось выполнено условие $\bar{g}(X_m^\alpha) \leq 0$, то для всех граней, полученных из X_m^α ветвлением, можно не проверять аналогичные условия, так как они будут выполнены автоматически.

4. Свойства алгоритма

Рассмотрим последовательность многогранников

$$Q_0 = \underline{d}(X_0) + \mathbf{R}_+^{N_c + N_f}, \quad Q_{l+1} = \overline{\text{conv} \left(\bigcup_{A \in \Xi_{l+1}} \{\underline{d}(A)\} \cup P_{l+1} \right)}. \quad (3)$$

Теорема Алгоритм заканчивает работу за конечное число итераций N , не превышающее 2^n , и порождает монотонные по включению последовательности многогранников $\{P_l\}$ и $\{Q_l\}$, причем

$$P_0 \subset P_1 \subset \dots \subset P_N = Y_P^C = Q_N \subset Q_{N-1} \subset \dots \subset Q_0.$$

Из теоремы следует, что многогранники Q_l и P_l являются соответственно внешней и внутренней аппроксимациями Y_P^C . При этом на каждой итерации происходит улучшение аппроксимации.

Выбор грани для разбиения, указанный в алгоритме, можно интерпретировать, как стремление минимизировать на каждой итерации ε_l .

Список литературы

- [1] A. V. Lotov, V. A. Bushenkov and G. K. Kamenev, *Interactive Decision Maps. Approximation and Visualization of Pareto frontier* (Kluwer Academic Publishers, Boston, 2004).
- [2] Лотов А.В., Бушенков В.А., Каменев Г.К. Компьютер и поиск компромисса. Метод достижимых целей. М.: Наука, 1997.
- [3] A.Lotov, V. Bushenkov, and G. Kamenev. (2001) Feasible Goals Method. Search for Smart Decisions. Computing Center RAS, Moscow, Russia, 240 p. (in English).
- [4] Л.В.Бурмистрова, Р.В.Ефремов, А.В.Лотов. (2002) Методика визуальной поддержки принятия решений и ее применение в системах управления водными ресурсами. Известия АН. Сер. Теория и Системы Управления. 2002, № 5, 89-100.
- [5] Меламед И.И., Сигал И.Х., Владимиров Н.Ю. Исследование линейной свертки критериев в бикритериальной задаче о ранце. Ж. вычисл. матем. и матем. физ., 1999. т.39. №5.

ALGORITHM FOR APPROXIMATION OF EDWORTH-PARETO CONVEXHULL IN SPECIAL INTEGER MULTICRITERIA OPTIMIZATION PROBLEMS

A.I. Pospelov

CMC MSU, Moscow

e-mail: iamgenius@iwo.rstrsk-da.ru

Abstract. Integer multicriteria optimization problems are considered in article. An algorithm of polyhedral approximation of Edworth-Pareto convex hull for integer multicriteria optimization problems with criteria functions from special class is presented. The algorithm is based on the method of branches and baunds and build external and internal polyhedral approximation of Edgworth-Pareto convex hull with arbitrary accuracy.

Key words: multicriteria discrete optimization, Edgeworth-Pareto convex hull, Pareto frontier

ОБ УСТОЙЧИВОСТИ ОБОБЩЕННО-ОПТИМАЛЬНОЙ СИТУАЦИИ В КОНЕЧНОЙ КОАЛИЦИОННОЙ ИГРЕ С ПАРАМЕТРИЧЕСКОЙ КОНЦЕПЦИЕЙ РАВНОВЕСИЯ (ОТ ЛЕКСИКОГРАФИЧЕСКОЙ ОПТИМАЛЬНОСТИ ДО РАВНОВЕСНОСТИ ПО НЭШУ)¹

С. Е. Бухтояров, В. А. Емеличев

Белорусский государственный университет, Минск, Беларусь
e-mail: emelichev@bsu.by

Аннотация. Вводится параметрическая концепция равновесия (принцип оптимальности) в конечной коалиционной игре нескольких лиц в нормальной форме. Это концепция определяется способом разбиения множества игроков на коалиции. При этом двум крайним случаям такого разбиения соответствуют лексикографически оптимальная ситуация и ситуация равновесия по Нэшу. Проведен анализ устойчивости обобщенно-оптимальной ситуации к независимым возмущениям коэффициентов линейных функций выигрыша игроков. Найден предельный уровень таких возмущений, сохраняющих оптимальность ситуации.

Ключевые слова: конечная коалиционная игра, концепция равновесия, обобщенно-оптимальная ситуация, лексикографически оптимальная ситуация, ситуация равновесия по Нэшу, радиус устойчивости.

Введение

Целью теоретико-игровой модели обычно является отыскание классов решений, рациональных (согласованных) с точки зрения возможных действий и интересов охватываемых моделью лиц (участников, игроков) и организаций (коалиций). Для каждой игры в нормальной форме используются коалиционные и бескоалиционные принципы оптимальности – концепции равновесия, – приводящие, как правило, к различным ситуациям (исходам). В теории неантогонистических игр нет единого подхода к выработке таких концепций. Наиболее известной является концепция равновесия по Нэшу [1], а также ее разнообразные обобщения. В данной работе рассматривается параметрическая концепция равновесия конечной игры в нормальной форме, порождающая обобщенно-оптимальные ситуации. Параметром этой концепции является способ разбиения игроков на коалиции так, что двум крайним случаям такого разбиения (единая коалиция всех игроков и совокупность одноэлементных коалиций) соответствуют лексикографически оптимальные ситуации и ситуации равновесия по Нэшу. Приводится формула радиуса устойчивости ситуации, оптимальной на любом разбиении, к независимым возмущениям коэффициентов функций выигрыша игроков.

1. Основные определения и обозначения

Рассмотрим основной объект изучения теории игр – конечную игру нескольких лиц в нормальной форме [2],[3], в которой каждый игрок $i \in N_n = \{1, 2, \dots, n\}$, $n \geq 2$, располагает конечным числом возможностей выбора – стратегий $X_i \subset \mathbf{R}$, $2 \leq |X_i| < \infty$.

¹Работа выполнена при поддержке Государственной программы фундаментальных исследований Республики Беларусь "Математические структуры 29"(грант 913/28).

Реализация игры и ее исход (ситуация) однозначно определяется выбором стратегий каждым из игроков. Пусть на множестве ситуаций игры $X = \prod_{j=1}^n X_j \subset \mathbf{R}^n$ заданы линейные функции выигрыша (платежные функции) игроков $f_i(x) = C_i x$, $i \in N_n$, где C_i – i -я строка матрицы $C = [c_{ij}] \in \mathbf{R}^{n \times n}$, $x = (x_1, x_2, \dots, x_n)^T$, $x_j \in X_j$, $j \in N_n$. В результате игры, которую будем называть игрой с матрицей C , каждый игрок i получает выигрыш $f_i(x)$, который стремится максимизировать, используя определенные отношения предпочтения.

Всякое непустое подмножество $J \subseteq N_n$ назовем коалицией игроков. Для любой коалиции J и всякой ситуации $x^0 = (x_1^0, x_2^0, \dots, x_n^0) \in X$ введем множество ситуаций, достижимых коалицией J из ситуации x^0 :

$$W(x^0, J) = \prod_{j=1}^n W_j(x^0, J),$$

где

$$W_j(x^0, J) = \begin{cases} X_j, & \text{если } j \in J, \\ \{x_j^0\}, & \text{если } j \in N_n \setminus J. \end{cases}$$

Далее введем бинарное отношение лексикографического порядка " \prec " в пространстве \mathbf{R}^d любой размерности $d \in \mathbf{N}$, считая, что для двух различных векторов $y = (y_1, y_2, \dots, y_d)$ и $y' = (y'_1, y'_2, \dots, y'_d)$ этого пространства справедлива формула

$$y \prec y' \Leftrightarrow y_k < y'_k,$$

где $k = \min\{i \in N_d : y_i \neq y'_i\}$.

Пусть $s \in N_n$, $N_n = \bigcup_{r \in N_s} J_r$ – разбиение множества N_n на s коалиций, т. е. $J_r \neq \emptyset$, $r \in N_s$; $p \neq q \Rightarrow J_p \cap J_q = \emptyset$. Для этого разбиения введем множество $G^n(C, J_1, J_2, \dots, J_s)$ обобщенно-оптимальных, или иначе (J_1, J_2, \dots, J_s) -оптимальных, ситуаций игры с матрицей C согласно формуле

$$G^n(C, J_1, J_2, \dots, J_s) = \{x \in X : \forall r \in N_s (\zeta(x, C, J_r) = \emptyset)\},$$

где $\zeta(x, C, J_r) = \{x' \in W(x, J_r) : C_{J_r} x \prec C_{J_r} x'\}$, C_{J_r} – подматрица матрицы C , состоящая из строк с номерами коалиции J_r .

Очевидно, что любая N_n -оптимальная ситуация $x \in G^n(C, N_n)$ (все игроки образуют одну коалицию) является лексикографически оптимальной в пространстве всех ситуаций X . Тем самым, в этом случае все игроки упорядочены (перенумерованы) по важности так, что каждый предыдущий важнее, чем все последующие. Это соответствует общей постановке задачи оптимизации по последовательно применяемым критериям (выигрышам) [4],[5],[6].

В другом крайнем случае, когда игра бескоалиционная ($s = n$), любая индивидуально-оптимальная ситуация $x \in G^n(C, \{1\}, \{2\}, \dots, \{n\})$ является ситуацией равновесия по Нэшу (или равновесной) [1] (см. также [2] и [3]) и рациональность такой ситуации заключается в том, что любые отклонения одного игрока от этой ситуации (в то время как другие ее придерживаются) не приносит ему выгоды.

Таким образом, в данном контексте под параметризацией принципа оптимальности понимается введение такой коалиционной характеристики ситуации, которая позволяет связать классические понятия лексикографической оптимальности и равновесности по

Нэш. При этом внутри каждой коалиции задается свой лексикографический принцип оптимальности.

Не исключая общности, будем в дальнейшем полагать, что элементы разбиения $N_n = \bigcup_{r \in N_s} J_r$ имеют вид $J_1 = \{1, 2, \dots, t_1\}$, $J_2 = \{t_1 + 1, t_1 + 2, \dots, t_2\}, \dots, J_s = \{t_{s-1} + 1, t_{s-1} + 2, \dots, n\}$.

Теорема 1. Для любого разбиения (J_1, J_2, \dots, J_s) справедлива формула

$$G^n(C, J_1, J_2, \dots, J_s) = \prod_{r=1}^s L^{|J_r|}(C^r, X_{J_r}),$$

где каждый множитель $L^{|J_r|}(C^r, X_{J_r})$ является множеством лексикографически оптимальных решений $|J_r|$ -критериальной векторной задачи

$$C^r z \rightarrow \text{lex max}, \quad z \in X_{J_r}$$

т. е.

$$L^{|J_r|}(C^r, X_{J_r}) = \{z \in X_{J_r} : \lambda(z, C^r) = \emptyset\},$$

где $\lambda(z, C^r) = \{z' \in X_{J_r} : C^r z \prec C^r z'\}$, C^r – квадратная матрица размера $|J_r| \times |J_r|$, состоящая из элементов матрицы C , находящихся на пересечении строк и столбцов с номерами J_r ; X_{J_r} – проекция множества X на координатные оси пространства \mathbb{R}^n с номерами из J_r , т. е.

$$X_{J_r} = \prod_{j \in J_r} X_j \subset \mathbf{R}^{|J_r|}.$$

Очевидно, что при любом разбиении множество $G^n(C, J_1, J_2, \dots, J_s) \neq \emptyset$.

Хорошо известно (см., например, [5] и [7]), что всякое множество $L^{|J_r|}(C^r, X_{J_r})$, $r \in N_s$, есть результат решения последовательности $|J_r|$ задач:

$$L_i^r = \text{Arg max}\{C_i^r z : z \in L_{i-1}^r\}, \quad i \in N_{|J_r|},$$

где $L_0^r = X_{J_r}$; C_i^r – i -я строка матрицы C^r . Таким образом, $L^{|J_r|}(C^r, X_{J_r}) = L_{|J_r|}^r \neq \emptyset$ для каждого индекса $r \in N_s$.

Для всякого натурального числа q в пространстве \mathbf{R}^q зададим нормы l_1 и l_∞ :

$$\|z\|_1 = \sum_{i \in N_q} |z_i|, \quad \|z\|_\infty = \max\{|z_i| : i \in N_q\}, \quad z = (z_1, z_2, \dots, z_q) \in \mathbf{R}^q.$$

Под нормой матрицы $C = [c_{ij}] \in \mathbf{R}^{n \times n}$ будем понимать норму вектора $(c_{11}, c_{12}, \dots, c_{n, n-1}, c_{nn}) \in \mathbf{R}^{nn}$. Для произвольного числа $\varepsilon > 0$ определим множество возмущающих матриц $\Omega(\varepsilon) = \{B \in \mathbf{R}^{n \times n} : \|B\|_\infty < \varepsilon\}$.

2. Формула радиуса устойчивости

По аналогии с [8],[9] и [10] под радиусом устойчивости (J_1, J_2, \dots, J_s) -оптимальной ситуации $x \in G^n(C, J_1, J_2, \dots, J_s)$ будем понимать число

$$\rho^n(x, C, J_1, J_2, \dots, J_s) = \begin{cases} \sup \Xi, & \text{если } \Xi \neq \emptyset, \\ 0 & \text{в противном случае,} \end{cases}$$

где $\Xi = \{\varepsilon > 0 : \forall B \in \Omega(\varepsilon) (x \in G^n(C + B, J_1, J_2, \dots, J_s))\}$.

Через $x_{J_r}^0$ будем обозначать проекцию ситуации $x^0 = (x_1^0, x_2^0, \dots, x_n^0) \in X$ на координатные оси пространства \mathbb{R}^n с номерами из коалиции J_r . Тем самым, $x_{J_r}^0 \in X_{J_r}$.

Теорема 2. Пусть $C \in \mathbf{R}^{n \times n}$, $s \in N_n$, $n \geq 2$. Для радиуса устойчивости $\rho^n(x^0, C, J_1, J_2, \dots, J_s)$ обобщенно-оптимальной ситуации $x^0 \in G^n(C, J_1, J_2, \dots, J_s)$ игры с матрицей C справедлива формула

$$\rho^n(x^0, C, J_1, J_2, \dots, J_s) = \min_{r \in N_s} \min_{z \in X_{J_r} \setminus \{x_{J_r}^0\}} \frac{C_1^r(x_{J_r}^0 - z)}{\|x_{J_r}^0 - z\|_1},$$

где C_1^r – первая строка матрицы C^r .

3. Следствия

Ситуацию $x^0 \in G^n(C, J_1, J_2, \dots, J_s)$ назовем устойчивой, если $\rho^n(x^0, C, J_1, J_2, \dots, J_s) > 0$, и строгой, если для любых $r \in N_s$ и $z \in X_{J_r} \setminus \{x_{J_r}^0\}$ выполняется неравенство

$$C_1^r x_{J_r} > C_1^r z.$$

Из теоремы 2 вытекает ряд следствий.

Следствие 1. Для ситуации $x^0 \in G^n(C, J_1, J_2, \dots, J_s)$ следующие утверждения эквивалентны:

- (i) ситуация x^0 устойчива,
- (ii) ситуация x^0 строгая,
- (iii) для любого индекса $r \in N_s$ справедливо равенство

$$|\text{Arg max}\{C_1^r z : z \in X_{J_r}\}| = 1.$$

Следствие 2. Если (J_1, J_2, \dots, J_s) -оптимальная ситуация x^0 игры с матрицей C устойчива, то

$$G^n(C, J_1, J_2, \dots, J_s) = \{x^0\}.$$

Нетрудно понять, что обратное утверждение, вообще говоря, не верно при условии существования хотя бы одной коалиции, состоящей по крайней мере из двух игроков.

Следствие 3. Если $|G^n(C, J_1, J_2, \dots, J_s)| > 1$, то никакая ситуация $x \in G^n(C, J_1, J_2, \dots, J_s)$ не может быть устойчивой.

Следствие 4. Для радиуса устойчивости лексикографически-оптимальной ситуации $x^0 \in G^n(C, N_n)$ игры с матрицей C справедлива формула

$$\rho^n(x^0, C, N_n) = \min_{x \in X \setminus \{x^0\}} \frac{C_1(x^0 - x)}{\|x^0 - x\|_1}.$$

Следствие 5. [8],[9] Для радиуса устойчивости ситуации равновесия по Нэшу $x^0 \in G^n(C, \{1\}, \{2\}, \dots, \{n\})$ игры с матрицей C справедлива формула

$$\rho^n(x^0, C, \{1\}, \{2\}, \dots, \{n\}) = \min\{|c_{ii}| : i \in N_n\}.$$

Теорема 1 позволяет сформулировать следующий сопутствующий результат.

Следствие 6. [10] Ситуация $x^0 = (x_1^0, x_2^0, \dots, x_n^0) \in X$ игры с матрицей $C \in \mathbf{R}^{nn}$ является ситуацией равновесия по Нэшу тогда и только тогда, когда стратегия каждого игрока $i \in N_n$ имеет вид

$$x_i^0 = \begin{cases} \max\{x_i : x_i \in X_i\}, & \text{если } c_{ii} > 0, \\ \min\{x_i : x_i \in X_i\}, & \text{если } c_{ii} < 0, \\ x_i \in X_i, & \text{если } c_{ii} = 0. \end{cases}$$

Список литературы

- [1] Нэш Дж. *Бескоалиционные игры*. – В кн. Матричные игры, Москва: Физматгиз, 1961, с. 205–221.
- [2] Мулен Э. *Теория игр*. Москва: Мир, 1985, 200 с.
- [3] Петросян Л.А., Зенкевич Н.А., Семина Е.А. *Теория игр*. Москва: Высш. школа, 1998, 304 с.
- [4] Подиновский В. В., Гаврилов В. М. *Оптимизация по последовательно применяемым критериям*. Москва: Советское радио, 1975, 192 с.
- [5] Сергиенко И. В. *Математические модели и методы решения задач дискретной оптимизации*. Киев: Наукова думка, 1988, 170 с.
- [6] Червак Ю. Ю. *Непокрацуванный выбор*. Ужгород: Ужгородский национальный университет, 2002, 312 с.
- [7] Еремин И. И. *О задачах последовательного программирования*. Сибирский математический журнал, 1973, т. 14, N 1, с. 53–63.
- [8] Emelichev V. A., Bukhtoyarov S. E. *Stability of generally efficient situation in finite cooperative games with parametric optimality principle ("from Pareto to Nash")*. Computer Science J. of Moldova, 2003, v. 11, N 3, p. 316–323.
- [9] Бухтояров С. Е., Емеличев В. А., Степанишина Ю. В. *Вопросы устойчивости векторных дискретных задач с параметрическим принципом оптимальности*. Кибернетика и системный анализ, 2003, N 4, с. 155–166.
- [10] Бухтояров С. Е., Емеличев В. А. *Конечные коалиционные игры: параметризация принципа оптимальности ("от Парето до Нэша") и устойчивость обобщенно-эффективных ситуаций*. Доклады НАН Беларуси, 2002, т. 46, N 6, с. 36–38.

ON STABILITY OF AN GENERALLY-OPTIMAL SITUATION IN A FINITE COOPERATIVE GAME WITH PARAMETRIC CONCEPT OF EQUILIBRIUM (FROM LEXICOGRAPHIC OPTIMALITY TO NASH EQUILIBRIUM)

S. E. Bukhtoyarov, V. A. Emelichev

Belorussian State University, Minsk, Belarus
e-mail: emelichev@bsu.by

Abstract. A parametric concept of equilibrium (principle of optimality) in a finite cooperative game in normal form of several players is introduced. This concept is defined by means of the partition of players into coalitions. Lexicographically optimal situation and Nash equilibrium situation are correspond to two special cases of this partition. The quantitative analysis of stability of an optimal situation to the independent perturbations of players' payoff functions is performed. The maximum level of such perturbations which save the optimality of a situation is found.

Key words: finite cooperative game, concept of equilibrium, generally-optimal situation, lexicographically optimal situation, Nash equilibrium situation, stability radius.

ИНФОРМАЦИОННОЕ КВАЗИРАВНОВЕСИЕ В МОДЕЛИ ОБМЕНА С ЭКСТЕРНАЛИЯМИ ¹

В.А.Васильев

*Институт математики им. С.Л.Соболева СО РАН, Новосибирск
e-mail: vasilev@math.nsc.ru*

Аннотация. В работе анализируется понятие информационного квазиравновесия для модели экономического обмена с экстерналиями, обобщающее как традиционное определение квазиравновесия, так и понятие информационного равновесия, введенное В.Л.Макаровым. Устанавливаются условия существования информационных квазиравновесий и изучаются их экстремальные свойства, включая слабую Парето-оптимальность и неблокируемость относительно некоторых классов нечетких коалиций. Вводится понятие представительного нечеткого ядра и указываются достаточно общие предположения, гарантирующие совпадение этого ядра с множеством информационных квазиравновесий. Полученные теоремы существования и кооперативной характеристики информационных квазиравновесий дополняют и усиливают некоторые результаты, касающиеся существования и коалиционной стабильности информационных равновесий.

Ключевые слова: Информационное квазиравновесие, представительное нечеткое ядро.

Введение

В работе анализируется понятие информационного квазиравновесия для модели экономического обмена с экстерналиями, обобщающее как традиционное определение квазиравновесия (см., например, [1], [10]), так и понятие информационного равновесия, введенное В.Л.Макаровым [4] (см. также [5], [6] и [11]). Приводятся теоремы существования информационных квазиравновесий, аналогичные известным результатам, полученным для моделей обмена без экстерналий [10]. При стандартных предположениях устанавливаются достаточно сильные экстремальные свойства рассматриваемых квазиравновесий, включая их слабую Парето-оптимальность и неблокируемость относительно некоторых классов нечетких коалиций. Вводится понятие представительного нечеткого ядра и анализируются условия совпадения этого ядра с множеством информационных квазиравновесий. Полученные теоремы существования и кооперативной характеристики информационных квазиравновесий дополняют и усиливают более ранние результаты, касающиеся существования и коалиционной стабильности информационных равновесий (см. [2], [6], [14]).

Информационные квазиравновесия (кратко - I -квазиравновесия) порождаются стандартными квазиравновесиями в некотором неоклассическом расширении исходной модели с экстерналиями. Поэтому доказательство существования I -квазиравновесия, как и для стандартных моделей обмена, базируется на известной лемме Гейла-Никайдо-Дебре [7] и ее более общих аналогах (см., например, [3]). Основные трудности в реализации указанной схемы, как и в случае с информационными равновесиями [2], состоят в известной вырожденности рассматриваемого расширения. Что касается теоремы о совпадении представительного нечеткого ядра и множества I -квазиравновесий, то здесь, как и в

¹Работа выполнена при содействии Российского гуманитарного научного фонда (грант 02-02-00189а) и программы Государственной поддержки ведущих научных школ (грант 80.2003.6)

классическом случае (см., например, [8]), основным инструментом является теорема отделимости Минковского.

1. Информационное квазиравновесие: существование и оптимальность

Приведем необходимые определения и формулировки основных результатов, касающихся существования и слабой Парето-оптимальности информационных квазиравновесий. Моделью экономического обмена с экстерналиями (м.о.э.) будем называть систему

$$\mathcal{E} = \langle N, L, \{X_i, w^i, R_i\}_{i \in N} \rangle,$$

где $N = \{1, \dots, n\}$ - множество участников, $L = \{1, \dots, l\}$ - множество продуктов, $X_i \subseteq \mathcal{R}^l$ - потребительское множество участника $i \in N$, а $w^i \in \mathcal{R}^l$ - его начальный запас продуктов. Далее, символами R_i обозначены бинарные отношения предпочтения (кратко - предпочтения) участников $i \in N$, заданные на множестве X распределений м.о.э. \mathcal{E} , где

$$X = \prod_{i \in N} X_i$$

(напомним, для сравнения, что в классической модели обмена предпочтение каждого участника $i \in N$ определено на его потребительском множестве X_i и, следовательно, "нечувствительно" к уровню потребления других участников). Всюду в дальнейшем для простоты предполагается, что предпочтения $R_i \subseteq X \times X$ являются полными и транзитивными.

Переходя к определению информационного квазиравновесия, напомним, что для каждого $i \in N$ и $x \in X$ через $P_i(x)$ обозначается множество распределений, более предпочтительных для участника i , чем распределение x :

$$P_i(x) = \{z \in X \mid [(z, x) \in R_i] \& [(x, z) \notin R_i]\}$$

(включение $(z, x) \in R_i$ интерпретируется как " z не хуже, чем x в смысле предпочтения R_i "). Далее, распространим обозначение скалярного произведения $p^0 \cdot z^0 = \sum_{k \in L} p_k^0 z_k^0$ для векторов $p^0 = (p_1^0, \dots, p_l^0)$, $z^0 = (z_1^0, \dots, z_l^0)$ из \mathcal{R}^l на любые векторы $p = (p^1, \dots, p^n)$ и $z = (z^1, \dots, z^n)$ из векторного пространства $(\mathcal{R}^l)^N$, полагая, как обычно, $p \cdot z = \sum_{i \in N} p^i \cdot z^i$. Наконец, обозначим через $X(N)$ множество допустимых распределений модели \mathcal{E} , определяемое формулой

$$X(N) = \{x = (x^i)_{i \in N} \in X \mid \sum_{i \in N} x^i \leq \sum_{i \in N} w^i\}.$$

ОПРЕДЕЛЕНИЕ 1. Распределение $\bar{x} = (\bar{x}^i)_{i \in N} \in X(N)$ называется информационным квазиравновесием (I -квазиравновесием) м.о.э. \mathcal{E} , если существуют векторы $\bar{p}_0 \in \mathcal{R}_+^l$, $\bar{p}^i = (\bar{p}^{i1}, \dots, \bar{p}^{in}) \in (\mathcal{R}^l)^N$, $i \in N$, такие, что $\bar{p}^0 \neq 0$ и при этом выполняются условия

- 1) $\sum_{i \in N} \bar{p}^{ij} = \bar{p}^0$ для каждого $j \in N$,
- 2) $\bar{p}^i \cdot \bar{x} = \bar{p}^0 \cdot w^i$ и $\bar{p}^i \cdot x \geq \bar{p}^0 \cdot w^i$ для каждого $i \in N$ и $x \in P_i(\bar{x})$,
- 3) $\bar{p}^0 \cdot (\sum_{i \in N} w^i - \sum_{i \in N} \bar{x}^i) = 0$.

Содержательное истолкование I -квазиравновесия аналогично интерпретации информационного равновесия, предложенной в [4] (см. также [5] и [12]). Отметим лишь, что векторы \bar{p}^i играют роль индивидуальных цен, определяющих издержки участников $i \in N$ при

реализации того или иного распределения $x \in X$ (составляющие \bar{p}^{ij} вектора \bar{p}^i интерпретируются в [4] и [5] как цены за продукты, представляющие собой информацию для участника i о потреблении участника j).

Множество информационных квазиравновесий м.о.э. \mathcal{E} обозначим через $IW_q(\mathcal{E})$.

Для формулировки теорем существования I -квазиравновесий потребуются дополнительные обозначения. Для $i \in N$ и $x \in X$ под $P_i^{-1}(x)$ будем понимать множество распределений, менее предпочтительных для участника i , чем распределение x :

$$P_i^{-1}(x) = \{z \in X \mid [(x, z) \in R_i] \& [(z, x) \notin R_i]\}.$$

Кроме того, через $co Z$, как обычно, будем обозначать выпуклую оболочку множества Z .

ТЕОРЕМА 1. Если м.о.э. \mathcal{E} удовлетворяет предположениям:

- (1) X_i - непустые, выпуклые и компактные для всех $i \in N$,
- (2) $P_i^{-1}(x)$ - открытое в X_i для каждого $i \in N$ и $x \in X$,
- (3) $x \notin co P_i(x)$ для всех $x \in X$ и $i \in N$,
- (4) $P_i(x) \neq \emptyset$ для всех $x \in X(N)$ и $i \in N$,

то $IW_q(\mathcal{E}) \neq \emptyset$.

Усиленная версия теоремы 1, не предполагающая компактности множеств X_i , имеет следующий вид.

ТЕОРЕМА 1а. Если м.о.э. \mathcal{E} удовлетворяет предположениям (2) – (4) теоремы 1 и, кроме того, все множества X_i - непустые, выпуклые, замкнутые и ограниченные снизу, то $IW_q(\mathcal{E}) \neq \emptyset$.

Отметим, что приведенные теоремы существования не предполагают ни выпуклости, ни непрерывности предпочтений (не говоря уже о возможности их представления функциями полезности). Следует подчеркнуть также, что информационное расширение модели \mathcal{E} , используемое для доказательства теоремы 1а, не наследует, вообще-говоря, ограниченности снизу потребительских множеств исходной м.о.э. \mathcal{E} . Ключевую роль в преодолении указанных трудностей играет выбор подходящего вспомогательного конуса и учет специфики самого вложения модели \mathcal{E} в ее расширение (см. некоторые подробности в п.3).

Как уже отмечалось (см, например, [4], [5] и [12]), необходимость изучения информационных аналогов классических принципов оптимальности диктуется, главным образом, неэффективностью вальрасовских равновесий в моделях экономического обмена, учитывающих наличие экстерналий (зависимости предпочтений участников от уровня потребления других экономических агентов). Обращаясь к достаточно необременительным и естественным условиям, гарантирующим оптимальность I -квазиравновесий, будем считать, что всюду далее выполняется соотношение $0 \in X$ и предположение (3) теоремы 1. Напомним, что распределение $x \in X(N)$ называется слабо Парето-оптимальным, если для любого $z \in X$ справедлива импликация

$$\forall i \in N [z \in P_i(x)] \Rightarrow z \notin X(N).$$

Совокупность всех слабо Парето-оптимальных распределений м.о.э. \mathcal{E} будем обозначать через $\mathcal{P}(\mathcal{E})$. Как обычно, всякий вектор $\bar{p} = (\bar{p}^0, \bar{p}^1, \dots, \bar{p}^n)$, удовлетворяющий вместе с $\bar{x} \in IW_q(\mathcal{E})$ условиям определения 1, будем называть ценами, поддерживающими I -квазиравновесие \bar{x} . В случае, когда для всех $i \in N$ векторы \bar{p}^i принадлежат $(\mathcal{R}_+^l)^N$, такие цены будем называть неотрицательными. Напомним также, что через \mathcal{R}_{++}^l обозначается

совокупность положительных векторов из \mathcal{R}^l :

$$\mathcal{R}_{++}^l = \{x = (x_1, \dots, x_l) \in \mathcal{R}^l \mid x_k > 0, k = 1, \dots, l\}.$$

ТЕОРЕМА 2. Пусть м.о.э. \mathcal{E} удовлетворяет предположениям:

- (1) X_i - непустые и выпуклые для всех $i \in N$,
- (2) $\sum_{i \in N} w^i \in \mathcal{R}_{++}^l$ и $w^i \in \mathcal{R}_+^l$ для каждого $i \in N$,
- (3) $P_i(x)$ - открытые в X_i для всех $i \in N$ и $x \in X(N)$.

Тогда любое I -квазиравновесие модели \mathcal{E} , поддерживаемое неотрицательными ценами, является слабо Парето-оптимальным.

Простейшее условие, обеспечивающее в предположениях теоремы 2 неотрицательность *всех* (а не только некоторых) поддерживающих цен, можно назвать локальной монотонностью предпочтений. Оно формулируется следующим образом (ниже $[x, y] =: \{z \in (\mathcal{R}^l)^N \mid x \leq z \leq y\}$ - порядковый интервал с концами в точках $x, y \in (\mathcal{R}^l)^N$):

(M) Для каждого участника $i \in N$ и для любого распределения $x \in X(N)$ существуют элемент $a = a(x) \in (\mathcal{R}_{++}^l)^N$ и продукт $k = k(i) \in L$, удовлетворяющие соотношениям: $x + b \in P_i(x)$ для всех $b = (b^1, \dots, b^n) \in [0, a]$ таких, что $b_k^i > 0$.

Приводимая ниже лемма указывает широкую область применимости теоремы 2.

ЛЕММА 1. Пусть $x \in IW_q(\mathcal{E})$. Если выполняется условие (M) и предположение (2) теоремы 2, то любые цены p , поддерживающие x , являются неотрицательными.

2. Представительное нечеткое ядро и информационные квазиравновесия

Вложение $W_q(\mathcal{E}) \subseteq \mathcal{P}(\mathcal{E})$, установленное в предыдущем пункте, допускает существенное усиление. Ниже предлагается один из вариантов такого усиления, основанный на использовании нечеткого блокирования, введенного в работе [6] (см. также [12] и [14]). Напомним, прежде всего (см., например, [8]), что нечеткими коалициями называются ненулевые элементы единичного куба $[0, 1]^n$. Положим

$$\mathcal{T} = \{\tau = (\tau_1, \dots, \tau_n) \in [0, 1]^n \mid \tau_i > 0, i \in N\}.$$

Элементы множества \mathcal{T} (т.е. коалиции, в которых с ненулевой долей участия представлен каждый из экономических агентов) будем называть представительными нечеткими коалициями. Следуя [6] (см. также [14]), определим нечеткое блокирование с помощью коалиций из \mathcal{T} .

ОПРЕДЕЛЕНИЕ 2. Будем говорить, что нечеткая коалиция $\tau = (\tau_1, \dots, \tau_n) \in \mathcal{T}$ блокирует распределение $x \in X(N)$, если найдутся распределения $x^0 = (x^{01}, \dots, x^{0n}) \in X$ и $z^i = (z^{i1}, \dots, z^{in}) \in X$, $i \in N$, такие, что выполняются условия

- 1) $z^i \in P_i(x)$ для каждого $i \in N$,
- 2) $\tau_i z^i = (\tau_1 x^{01}, \dots, \tau_n x^{0n})$ для каждого $i \in N$,
- 3) $\sum_{i \in N} \tau_i x^{0i} \leq \sum_{i \in N} \tau_i w^i$.

Множество распределений $x \in X(N)$, не блокируемых никакой нечеткой коалицией из \mathcal{T} , будем обозначать через $IC_F^r(\mathcal{E})$. Множество $IC_F^r(\mathcal{E})$ будем называть представительным нечетким ядром (кратко - r -ядром) м.о.э. \mathcal{E} .

Справедливо следующие усиление теоремы 2.

ТЕОРЕМА 3. В условиях теоремы 2 справедливо вложение

$$IW_q(\mathcal{E}) \subseteq IC_F^r(\mathcal{E}).$$

Оказывается, что при достаточно слабых и естественных условиях справедливо и противоположное вложение (ниже используется стандартное обозначение: $o(x, y) =: \{x + \lambda(y - x) \mid \lambda \in (0, 1]\}$).

ТЕОРЕМА 4. Если множества $P_i(x)$ непустые и выпуклые для любого распределения $x \in X(N)$ и, кроме того, для каждого участника $i \in N$ и для любых $x \in X(N)$ и $y \in P_i(x)$ выполняются вложения $o(x, y) \subseteq P_i(x)$, то каждый элемент r -ядра $IC_F^r(\mathcal{E})$ является I -квазиравновесием м.о.э. \mathcal{E} .

Комбинируя теоремы 3,4 и лемму 1, получаем следующий результат об условиях совпадения представительного нечеткого ядра и множества информационных квазиравновесий в модели обмена с экстерналиями.

ТЕОРЕМА 5. Если для м.о.э. \mathcal{E} выполняется предположение (M) и условия теорем 3,4, то распределение $x \in X(N)$ является I -квазиравновесием тогда и только тогда, когда оно не блокируется никакой представительной нечеткой коалицией.

3. Информационное расширение модели обмена с экстерналиями

В заключение, следуя [5], [6], приведем краткое описание информационного расширения модели обмена с экстерналиями, играющего ключевую роль в доказательстве основных результатов работы. Пусть $\mathcal{E} = \langle N, L, \{X_i, w^i, R_i\}_{i \in N} \rangle$ - произвольная модель с экстерналиями. Положим

$$\mathcal{D} = \{(i, j) \in N \times N \mid i \neq j\}, \quad \mathcal{D}_0 = \{0\} \cup \mathcal{D}$$

и введем следующие обозначения:

$$X_i^0 = X_i, \quad i \in N; \quad (1)$$

$$X_i^{jk} = \begin{cases} X_k, & j = i, \\ -X_i, & k = i, \\ \{0\} & \text{в остальных случаях,} \end{cases} \quad (j, k) \in \mathcal{D}. \quad (2)$$

Для каждого $i \in N$ определим "информационное" расширение X_i^Δ его потребительского множества X_i :

$$X_i^\Delta = \{(x_d^i)_{d \in \mathcal{D}_0} \in X_i^0 \times \prod_{(j,k) \in \mathcal{D}} X_i^{jk} \mid x_{(j,i)}^i = -x_0^i \text{ для каждого } j \in N \setminus \{i\}\}, \quad (3)$$

и через $w_\Delta^i \in (\mathcal{R}^l)^{\mathcal{D}_0}$ обозначим вектор, компоненты которого вычисляются по формуле

$$(w_\Delta^i)_0 = w^i, \quad (w_\Delta^i)_{(j,k)} = 0, \quad (j, k) \in \mathcal{D}. \quad (4)$$

Далее, через R_i^Δ обозначим бинарное отношение предпочтения на X_i^Δ , определяемое формулой

$$((x_d^i)_{d \in \mathcal{D}_0}, (z_d^i)_{d \in \mathcal{D}_0}) \in R_i^\Delta \Leftrightarrow ((x_{(i,1)}^i, \dots, x_{(i,n)}^i), (z_{(i,1)}^i, \dots, z_{(i,n)}^i)) \in R_i, \quad (5)$$

где $x_{(i,i)}^i =: x_0^i$, $z_{(i,i)}^i =: z_0^i$. Наконец, определим (агрегированное) производственное множество Y^Δ согласно формуле $Y^\Delta = (-\mathcal{R}_+^l) \times \{0\} \times \dots \times \{0\}$, или, в более развернутой форме

$$Y^\Delta = \{y = (y_d)_{d \in \mathcal{D}_0} \in (-\mathcal{R}_+^l) \times (\mathcal{R}^l)^{\mathcal{D}} \mid y_{(i,j)} = 0, (i,j) \in \mathcal{D}\}, \quad (6)$$

и положим $L^\Delta = L \times \mathcal{D}_0$.

ОПРЕДЕЛЕНИЕ 3. Модель обмена с агрегированным производством

$$\mathcal{E}^\Delta = \langle N, L^\Delta, \{X_i^\Delta, w_\Delta^i, R_i^\Delta\}_{i \in N}, Y^\Delta \rangle,$$

потребительские множества, начальные запасы, предпочтения и производственное множество которой определены в соответствии с формулами (1) - (6), будем называть информационным расширением (или рынком информации) м.о.э. \mathcal{E} .

Отметим, что векторы $x_\Delta^i \in X_i^\Delta$ характеризуют возможное собственное потребление x_0^i участника $i \in N$ и желаемый им уровень потребления $x_{(i,k)}^i$ участников $k \in N \setminus \{i\}$ экономики \mathcal{E} . При этом компоненты $x_{(j,i)}^i = -x_0^i$ указывают объем собственного потребления участника $i \in N$, о котором ставятся в известность экономические агенты $j \in N \setminus \{i\}$. Важной особенностью модели \mathcal{E}^Δ , заложенной в самой её конструкции, является тот факт, что предпочтения R_i^Δ участников \mathcal{E}^Δ определены только на их собственных потребительских множествах X_i^Δ .

Вложение исходной м.о.э. \mathcal{E} в \mathcal{E}^Δ осуществляется с помощью линейного оператора $\gamma = (\gamma_1, \dots, \gamma_n)$, где отображения $\gamma_i : (\mathcal{R}^l)^N \rightarrow (\mathcal{R}^l)^{\mathcal{D}_0}$ для каждого $i \in N$ и $x = (x^1, \dots, x^n) \in (\mathcal{R}^l)^N$ определяются формулами:

$$(\gamma_i(x^1, \dots, x^n))_d = \begin{cases} x^i, & d = 0; \\ -x^i, & d = (k, i), k \in N \setminus \{i\}; \\ x^k, & d = (i, k), k \in N \setminus \{i\}; \\ 0 & \text{в остальных случаях.} \end{cases} \quad (7)$$

Ясно, что $\gamma_i(X) = X_i^\Delta$ и $(\gamma_i(x), \gamma_i(z)) \in R_i^\Delta \Leftrightarrow (x, z) \in R_i$ для всех $i \in N$ и $x, z \in X$. При этом, как нетрудно проверить, отображение $\gamma : x \mapsto (\gamma_1(x), \dots, \gamma_n(x))$, $x \in X$, осуществляет взаимно-однозначное соответствие между допустимыми распределениями моделей \mathcal{E} и \mathcal{E}^Δ . Более того, нетрудно проверить, что непосредственно из определения множества $W_q(\mathcal{E}^\Delta)$ классических квазиравновесий модели \mathcal{E}^Δ вытекает равенство

$$\gamma(IW_q(\mathcal{E})) = W_q(\mathcal{E}^\Delta). \quad (9)$$

Равенство (9) позволяет редуцировать вопрос об условиях непустоты $IW_q(\mathcal{E})$ к соответствующей задаче для стандартного квазиравновесия. Этот подход и используется в настоящей работе. Трудность его реализации, как уже отмечалось, состоит в том, что модель \mathcal{E}^Δ не удовлетворяет практически ни одному из типичных условий, так или иначе фигурирующих в известных теоремах существования. В частности, какова бы ни была исходная м.о.э. \mathcal{E} , начальные запасы w_Δ^i модели \mathcal{E}^Δ не принадлежат потребительским множествам X_i^Δ при $w^i \neq 0$. Далее, множества X_i^Δ не имеют внутренних точек ни при каких $i \in N$, а бюджетные отображения модели \mathcal{E}^Δ разрывны в стандартной области их определения. К этому списку можно добавить и неограниченность снизу потребительских множеств X_i^Δ в случае ограниченности снизу (но не сверху!) потребительских множеств

исходной модели \mathcal{E} . Таким образом, даже исследование не учитывающей производственных аспектов, относительно простой модели с экстерналиями требует привлечения достаточно тонких методов анализа, основанных на результатах работ [5], [9], [10], [13] и их современных обобщений.

Список литературы

- [1] К. Алипрантис, Д. Браун, О. Бёркеншо *Существование и оптимальность конкурентного равновесия*. - М.: Мир, 1995, 384 с.
- [2] В.А. Васильев *Существование информационного равновесия в экономике чистого обмена*. - Оптимизация, 1983, вып. 33(50), с. 79-94.
- [3] В.А. Васильев *Модели экономического обмена и кооперативные игры*. - Новосибирск: Изд-во НГУ, 1984, 97 с.
- [4] В.Л. Макаров *Модели согласования экономических интересов*. - Новосибирск: Изд-во НГУ, 1981, 67 с.
- [5] В.Л.Макаров *Экономическое равновесие: Существование и экстремальные свойства*. - В кн. Современные проблемы математики / Итоги науки и техники, т. 19. М.: ВИНТИ АН СССР, 1982, с. 23-58.
- [6] В.Л.Макаров, В.А.Васильев *Информационное равновесие и ядро в обобщенной модели обмена*. - Докл. АН СССР, 1984, т. 275, N3, с. 549-553.
- [7] Х. Никайдо *Выпуклые структуры и математическая экономика*. - М.: Мир, 1972, 519 с.
- [8] И. Экланд *Элементы математической экономики*. - М.: Мир, 1983, 248 с.
- [9] Bergstrom T.C. *How to discard "free disposability" - At no cost*. - J. Math. Econ., 1976, v. 3, N2, p. 131-134.
- [10] Florenzano M. *L'équilibre économique général transitif et intransitif: Problèmes d'existence*. - Paris: CNRS - CEPREMAP, 1981, 137 p.
- [11] Makarov V.L. *Some results on general assumptions about the existence of economic equilibrium*. - J. Math. Econ., 1981, v. 8, N1, p. 87-99.
- [12] Makarov V.L., Vasil'ev V.A., et al. *Equilibria, rationing and stability*. - Matekon, 1989, v. 25, N4, p. 4-95.
- [13] Shafer W. *Equilibrium in economies without ordered preferences or free disposal*. - J.Math. Econ., 1976, v. 3, N2, p. 135-138.
- [14] Vasil'ev V.A. *On Edgeworth equilibria for some types of nonclassic markets*. - Siberian Adv. Math., 1996, v. 6, N3, p. 96-150.

INFORMATION QUASI-EQUILIBRIUM IN EXCHANGE MODELS WITH EXTERNALITIES

V.A. Vasil'ev

*Sobolev Institute of Mathematics, Russian Academy of Sciences, Novosibirsk
e-mail: vasilev@math.nsc.ru*

Abstract. In this paper, we propose the notion of information quasi-equilibrium for an exchange economy with externalities. Being a direct generalization of the concept of information equilibrium, introduced by V.L.Makarov, the notion under consideration is a natural extension of the classic definition of quasi-equilibrium, as well. Some existence results similar to that obtained for the classic quasi-equilibrium are established, and coalitional stability with respect to several types of blocking is investigated. In particular, the concept of representative fuzzy core is introduced, and the coincidence of this core and the set of information quasi-equilibria is demonstrated under rather general assumptions. Main core equivalence and existence theorems, given in the paper, amplify previously obtained results on the information equilibrium.

Key words: information quasi-equilibrium, representative fuzzy core.

РАЗРЕШЕНИЕ ЗАДАЧИ ДЕЛЕЖА В КООПЕРАТИВНОЙ ИГРЕ С ПОМОЩЬЮ СИСТЕМЫ ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ

С.А. Осипов

Запорожский государственный университет, Запорожье
e-mail: webmaster@zsu.zp.ua

Аннотация. В статье описывается программа, представляющая собой систему поддержки принятия решений для задачи дележа в кооперативной игре. Выявляя наличие или отсутствие ядра в кооперативной игре, компьютеризированная система искусственно создает ядро (в случае его отсутствия), тем самым позволяя распределить дележ в кооперативной игре и предотвращая выход игроков из кооператива. Математическая модель системы поддержки принятия решений базируется на основах теории кооперативных игр, понятиях ядра, селектора ядра.

Ключевые слова: система, принятия, решений, кооперативная, игра, ядро, коалиция, дележ.

Введение

Разработана компьютеризированная система поддержки принятия решений в конфликтных ситуациях. Она позволяет оптимальным образом распределить дележ в кооперативной игре и предлагает механизм, который с помощью системы штрафов удержит участников кооператива от выхода из него при возникновении на то причин. На практике подобные задачи встречаются довольно часто, причем в основном в реальном секторе. Предположим, существует необходимость нескольких рядом находящихся поселков в газификации. Как распределить затраты на общественное строительство ветки газопровода в условиях объединения поселков в кооператив? Математическая интерпретация подобного рода задач решается предлагаемой системой поддержки принятия решений. Или например, франчайзинговая структура в силу своих объективных свойств может быть признана выгодной или невыгодной для франчайзера и/или франчайзи с помощью рассматриваемой программы.

1. Постановка задачи

Математическая модель данной компьютеризированной системы построена на базе теории кооперативных игр и понятия ядра в кооперативной игре [1]. Возможность эффективного решения задачи о дележе в кооперативной игре подразумевает наличие в ней ядра. Ядро существует, когда любая коалиция, состоящая из менее чем N участников кооперативной игры, будет в худших (менее выгодных) условиях вне кооператива, чем в его составе. Это условие выражается системой неравенств:

$$\text{для всех } S \subset N \quad v(S) \leq \sum_{i \in S} x_i \quad (1)$$

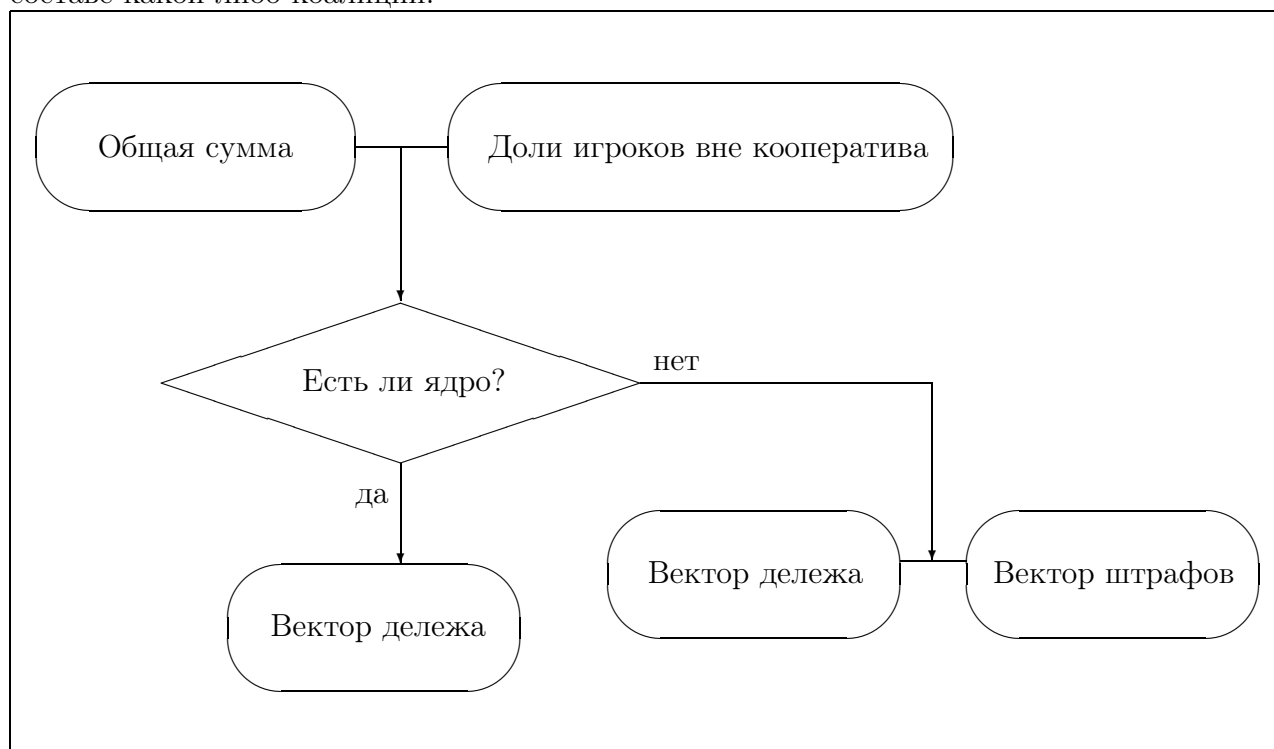
где N – множество игроков,

$v(S)$ – прибыль коалиции S из N за пределами кооператива,

x_i – прибыль i -го игрока в составе кооператива.

Когда ядра не существует, у некоторой коалиции, состоящей из одного или нескольких игроков, возникает интерес с выгодой для себя выйти из кооператива. Такое развитие событий можно предупредить, если при организации кооперативной игры ввести систему штрафов. Она должна быть минимально необходимой для того, чтобы нивелировать интерес коалиции выйти из кооператива. Иными словами такая система штрафов искусственно создаст ядро.

На вход в компьютеризированную систему поддержки принятия решений подается общая сумма игры, подлежащая дележу, а также значения доли каждой из возможных 2^{N-1} коалиций в случае их выхода из кооператива. В виде решения выдается вектор оптимального дележа общей прибыли (суммы) между игроками и, при отсутствии ядра, дополнительный вектор штрафов, назначаемых в случае выхода игрока из кооператива в составе какой-либо коалиции.



2. Метод решения

Ход решения выглядит следующим образом. Полученную выше систему неравенств (1) преобразуем к виду:

$$\text{для всех } S \subset N \quad v(S) - p(S) \leq \sum_{i \in S} x_i \quad (2)$$

$$p(S) \leq \sum_{i \in S} \alpha_i \quad (3)$$

где $p(S)$ – штраф, взимаемый с коалиции S при выходе из кооператива, α_i – штраф на игрока, выходящего из кооператива в составе коалиции S .

Систему неравенств (2) приводим к системе уравнений, путем добавления дополнительных неизвестных. Вводим целевую функцию суммы штрафов $p(S)$ на минимум:

$$\sum_{S \subset N} p(S) \rightarrow \min \quad (4)$$

Таким образом, мы приходим к задаче линейного программирования, которую решаем одним из известных методов. Затем аналогично решаем систему (3), устремляя к минимуму индивидуальные штрафы игроков α_i :

$$\sum_{i \in S} \alpha_i \rightarrow \min \quad (5)$$

Задачу, которую мы рассмотрели выше, можно видоизменять в зависимости от конкретных предпочтений. Например, оптимальным считать не такой вектор штрафов, где сумма штрафов всех участников минимальна (5), а где максимальный из штрафов игроков минимален (5'). Так мы будем стремиться к справедливости с другой точки зрения:

$$\max_{i \in S} \alpha_i \rightarrow \min \quad (5')$$

Список литературы

- [1] Мулен Э. *Кооперативное принятие решений: Аксиомы и модели*. Пер. с англ. – М.: Мир, 1991, - 464с., ил.
- [2] Чистяков С.В. *Динамический аспект решения классических кооперативных игр* Докл. РАН. 1993. Т.330, № 6. С.707-709.

SYSTEM OF DECISION-MAKING SUPPORT FOR THE PROBLEM OF THE SHARING IN COOPERATIVE GAME

S.A. Osypov

Zaporizhya State University, Zaporizhya

e-mail: webmaster@zsu.zp.ua

Abstract. The article represents the program which is a system of decision-making support for a problem of a sharing in cooperative game. Revealing presence or absence of a nucleus in the cooperative game, the computerized system artificially creates a nucleus, thus allowing to distribute a sharing in cooperative game and avoiding an ejecting of players from cooperative society. The mathematical model of system of decision-making support rests on the basis of the theory of cooperative games, concepts of a nucleus, the selector of a nucleus.

Key words: System, support, decision, cooperative, game, nucleus, coalition, sharing, distributing.

РАВНОВЕСНЫЙ ПОДХОД В ИГРОВЫХ МОДЕЛЯХ ПРОЦЕССА ОБУЧЕНИЯ

О.А.Попова

*Омский государственный университет, Омск
e-mail: olgaarc@yandex.ru*

Аннотация. В статье рассматривается игра двух лиц в смешанных стратегиях, а также метод ее решения. Решение игры представляет собой равновесие по Нэшу. Как игра так и метод ее решения интерпретируются как статическая и динамическая модель процесса обучения. Доказывается сходимость процесса обучения к его оптимальному (наилучшему) состоянию.

Ключевые слова: игра двух лиц, равновесие по Нэшу, процесс обучения, сходимость.

1. Постановка задачи

В данной работе рассматривается игра двух лиц в смешанных стратегиях как модель процесса обучения [1]. Поведение каждого из игроков в этой ситуации моделируется задачей параметрического квадратичного программирования с ограничениями типа равенств (симплексы). Любая точка этих симплексов представляет собой допустимую (смешанную) стратегию каждого из игроков. Предпочтения игроков описываются параметрическими квадратичными целевыми функциями, причем параметр одного из игроков является переменной для другого. Формально игра двух лиц определяет неподвижную (равновесную) точку или равновесие по Нэшу, которая представляет собой оптимальную стратегию процесса обучения, Один из игроков этого процесса представляет собой преподавателя, другой - группу студентов. Если допустимые стратегии обоих игроков бегают по вершинам вершинам симплексов, то мы получаем, как частный случай, биматричную игру в чистых стратегиях. В этой игре чистая стратегия, например для студентов, представляет собой конкретную подгруппу студентов, соответствующую некоторому номеру строки матрицы S , соответственно, чистая стратегия преподавателя отражает уровень сложности лекции, которая соответствует некоторому номеру столбца матрицы P . Если усреднить чистые стратегии и вместо номеров строк i и столбцов j матриц S и P ввести их смеси или распределения $x = (x_1, x_2, \dots, x_n)$ и $y = (y_1, y_2, \dots, y_n)$, где $\sum_1^n x_i = 1$ и $\sum_1^n y_i = 1$, то придем к биматричной игре обучения в смешанных стратегиях

$$\begin{aligned} x^* \in \operatorname{Argmax}\{\langle S^T x, y^* \rangle + (1/2)\langle Bx, x \rangle \mid \langle e, x \rangle = 1, x \geq 0\}, \\ y^* \in \operatorname{Argmax}\{\langle x^*, Py \rangle + (1/2)\langle Dy, y \rangle \mid \langle e, y \rangle = 1, y \geq 0\}, \end{aligned} \quad (1)$$

где $e = (1, 1, \dots, 1)$. Размерность e равна размерности x или y . Ограничения суть симплексы $\langle e, x \rangle = \sum_1^n x_j = 1$, где $x_j \geq 0$ и $\langle e, y \rangle = \sum_1^n y_i = 1$, где $y_i \geq 0$. В задаче (1) выражение Py представляет собой вектор-столбец, каждая компонента которого описывает смесь сложностей лекции преподавателя для всех подгрупп студентов, т.е. i -я подгруппа студентов имеет дело со смесью сложностей лекции вида $\sum_1^n p_{ij}y_j$. Соответственно смесь или распределение групп студентов будет иметь дело с суммарной сложностью лекции вида $\langle x, Py \rangle$. Аналогичные рассуждения справедливы и относительно другого участника игры. Действительно, выражение $S^T x$, где T - знак транспонирования матрицы, который позволяет нам иметь дело со строками матрицы S , вместо столбцов в отличие от матрицы

P описывает вектор эффективностей смеси подгрупп студентов для разных чистых стратегий преподавателя. Соответственно, величина характеризует $\langle S^T x, y \rangle$ эффективность распределения подгрупп студентов для некоторого выбранного распределения сложностей лекции преподавателя. Для учета ограничений каждого из игроков введем функции Лагранжа:

$$\begin{aligned} L_1(x, y^*, \lambda_1) &= \langle S^T x, y^* \rangle + (1/2)\langle Bx, x \rangle + \lambda_1(\langle e, x \rangle - 1), \\ L_2(x^*, y, \lambda_2) &= \langle x^*, Py \rangle + (1/2)\langle Dy, y \rangle + \lambda_2(\langle e, y \rangle - 1). \end{aligned}$$

В состоянии равновесия игры, т.е. при $x = x^*, y = y^*$, точки x^*, λ_1^* (y^* - фиксировано) и y^*, λ_2^* (x^* - фиксировано) назовем седловыми для функций Лагранжа для первого и второго игроков соответственно, если

$$\begin{aligned} L_1(x^*, y^*, \lambda_1) &\geq L_1(x^*, y^*, \lambda_1^*) \geq L_1(x, y^*, \lambda_1^*), \\ L_2(x^*, y^*, \lambda_2) &\geq L_2(x^*, y^*, \lambda_2^*) \geq L_2(x^*, y, \lambda_2^*) \end{aligned} \quad (2)$$

для всех $x \geq 0, y \geq 0, \lambda_1 \in R^1, \lambda_2 \in R^1$. Систему неравенств (2) можно записать в виде системы следующих задач:

$$\begin{aligned} x^* &\in \text{Argmax}\{\langle S^T x, y^* \rangle + (1/2)\langle Bx, x \rangle + \lambda_1^*(\langle e, x \rangle - 1) \mid x \geq 0\}, \\ &\langle e, x^* \rangle = 1, \\ y^* &\in \text{Argmax}\{\langle x^*, Py \rangle + (1/2)\langle Dy, y \rangle + \lambda_2^*(\langle e, y \rangle - 1) \mid y \geq 0\}, \\ &\langle e, y^* \rangle = 1. \end{aligned} \quad (3)$$

Таким образом, полученная система из четырех операторных уравнений представляет статическую модель ситуации обучения с учетом утилитарного (матрицы S и P) и рефлексивного (деонтологического) аспектов (матрицы B и D), характеризующие поведенческие аспекты игроков [2].

2. Динамическая модель процесса обучения

Обучение представляет собой протекающий во времени процесс, причем у каждого из участников этого процесса имеются свои задачи и функции. Так, например, на каждом шаге преподаватель выступает как процессор знаний (источник информации), который с помощью обучающих приемов и представлений об уровне знаний студентов на предшествующем шаге на основе вектора старых знаний студентов y^n вырабатывает вектор своих новых знаний x^{n+1} , предназначенных для усвоения студентами на следующем шаге. С другой стороны, в этом же процессе обучения студенты на основе знаний, представленных в лекции преподавателя x^n , приобретают вектор своих новых знаний y^{n+1} . Последовательность лекций определяет многошаговый процесс, в результате которого формируется последовательность векторов знаний студентов y^n , полученных из лекций преподавателя x^n . Последовательность векторов знаний $\{x^n\}, \{y^n\}$ должна завершиться освоением предметной области знания с максимальной эффективностью. Таким образом, если принять за основу высказанные выше предположения о процедуре передачи и усвоения знаний, то обучение можно представить в виде некоторого итерационного процесса, протекающего во времени, причем состояние игры на каждом итерационном шаге может быть описано системой операторных уравнений (3). Заметим, что модель (3) может быть представлена

также в виде системы операторных уравнений с использованием оператора проектирования или в виде вариационных неравенств. Модель в форме операторных уравнений имеет вид:

$$\begin{aligned}\lambda_1^* &= \lambda_1^* - \alpha(\langle e, x^* \rangle - 1), \\ \lambda_2^* &= \lambda_2^* - \alpha(\langle e, y^* \rangle - 1), \\ x^* &= \pi_+(x^* + \alpha(S^T y^* + Bx^* + \lambda_1^* e)), \\ y^* &= \pi_+(y^* + \alpha(Px^* + Dy^* + \lambda_2^* e)).\end{aligned}\tag{4}$$

где π_+ - оператор проектирования некоторого вектора на R_+^n или R_+^m . Модель в форме вариационных неравенств представляет систему вида:

$$\begin{aligned}\lambda_1^* &= \lambda_1^* - \alpha(\langle e, x^* \rangle - 1), \\ \lambda_2^* &= \lambda_2^* - \alpha(\langle e, y^* \rangle - 1), \\ \langle x - x^* - \alpha(S^T y^* + Bx^* + \lambda_1^* e), x - x^* \rangle &\geq 0 \\ \langle y - y^* - \alpha(Px^* + Dy^* + \lambda_2^* e), y - y^* \rangle &\geq 0.\end{aligned}\tag{5}$$

Для решения системы (4) естественно использовать идею простой итерации

$$\begin{aligned}x^{n+1} &\in \operatorname{Argmax}\{\langle S^T x, y^n \rangle + (1/2)\langle Bx, x \rangle + \lambda_1^n(\langle e, x \rangle - 1) \mid x \geq 0\}, \\ \lambda_1^{n+1} &= \lambda_1^n - \alpha(\langle e, x^n \rangle - 1), \\ y^{n+1} &\in \operatorname{Argmax}\{\langle x^n, Py \rangle + (1/2)\langle Dy, y \rangle + \lambda_2^n(\langle e, y \rangle - 1) \mid y \geq 0\}, \\ \lambda_2^{n+1} &= \lambda_2^n - \alpha(\langle e, y^n \rangle - 1).\end{aligned}\tag{6}$$

Если оператор этой системы строго сжимающий, то процесс (6) согласно принципу сжимающего отображения сходится к равновесию по Нэшу (x^*, y^*) . Однако, в общем случае оператор игры не является строго сжимающим и, следовательно, процесс (6) не сходится к решению игры. Для построения сходящегося процесса применим экстраградиентный подход [3]. Метод базируется на двух идеях: несжимаемость игрового оператора за счет регуляризации и расщепление основного шага на два полушага (прогнозный полушаг и основной полушаг). Итеративные формулы этого процесса имеют вид. Первый полушаг:

$$\begin{aligned}\bar{\lambda}_1^n &= \lambda_1^n - \alpha(\langle e, x^n \rangle - 1), \\ \bar{\lambda}_2^n &= \lambda_2^n - \alpha(\langle e, y^n \rangle - 1), \\ \bar{x}^n &= \pi_+(x^n + \alpha(S^T y^n + Bx^n + \bar{\lambda}_1^n e)), \\ \bar{y}^n &= \pi_+(y^n + \alpha(Px^n + Dy^n + \bar{\lambda}_2^n e)).\end{aligned}\tag{7}$$

Второй полушаг:

$$\begin{aligned}\lambda_1^{n+1} &= \lambda_1^n - \alpha(\langle e, \bar{x}^n \rangle - 1), \\ \lambda_2^{n+1} &= \lambda_2^n - \alpha(\langle e, \bar{y}^n \rangle - 1), \\ x^{n+1} &= \pi_+(x^n + \alpha(S^T \bar{y}^n + B\bar{x}^n + \bar{\lambda}_1^n e)), \\ y^{n+1} &= \pi_+(y^n + \alpha(P\bar{x}^n + D\bar{y}^n + \bar{\lambda}_2^n e)).\end{aligned}\tag{8}$$

Последовательность полушагов, соответствующая модели в форме вариационных неравенств, отличается от метода, представленного в форме операторных уравнений, только последними двумя выражениями в каждом полушаге. Первый полушаг:

$$\langle \bar{x}^n - x^n - \alpha(S^T y^n + Bx^n + \bar{\lambda}_1^n e), x - \bar{x}^n \rangle \geq 0, \quad (9)$$

$$\langle \bar{y}^n - y^n - \alpha(Px^n + Dy^n + \bar{\lambda}_2^n e), y - \bar{y}^n \rangle \geq 0,$$

Второй полушаг:

$$\langle x^{n+1} - x^n - \alpha(S^T \bar{y}^n + B\bar{x}^n + \bar{\lambda}_1^n e), x - x^{n+1} \rangle \geq 0, \quad (10)$$

$$\langle y^{n+1} - y^n - \alpha(P\bar{x}^n + D\bar{y}^n + \bar{\lambda}_2^n e), y - y^{n+1} \rangle \geq 0.$$

Перейдем к обсуждению вопросов сходимости этого подхода.

3. Сходимость метода

Заметим, что сходимость метода во многом будет определяться системными свойствами игровых матриц и длиной шага итерационного процесса. Прежде, чем перейти непосредственно к формулировке условий и доказательству сходимости метода, получим оценки отклонения векторов \bar{x}^n, \bar{y}^n и $x^{n+1}, y^{n+1}, \bar{\lambda}_1^n, \bar{\lambda}_2^n$ и $\lambda_1^{n+1}, \lambda_2^{n+1}$. Нетрудно показать, что верны оценки

$$\begin{aligned} |\bar{\lambda}_1^n - \lambda_1^{n+1}| &\leq \alpha|\bar{x}^n - x^n|, \quad |\bar{\lambda}_2^n - \lambda_2^{n+1}| \leq \alpha|\bar{x}^n - x^n|, \\ |\bar{x}^n - x^{n+1}| &\leq \alpha|S^T(\bar{y}^n - y^n) + B(\bar{x}^n - x^n)|, \\ |\bar{y}^n - y^{n+1}| &\leq \alpha|P(\bar{x}^n - x^n) + D(\bar{y}^n - y^n)|. \end{aligned} \quad (11)$$

Используя эти оценки, можно доказать теорему о сходимости процесса.

Теорема. Если множество решений задачи (1) не пусто, матрица $C + B^T \leq 0$ отрицательно полуопределенная, длина итерационного шага $\alpha > 0$ выбирается из условия $0 \leq \alpha < 1/\sqrt{2|B^* + C|^2 + 1}$, то последовательность x^n, y^n , порожденная методом (7), (8) сходится монотонно по норме к одному из игровых решений, т.е. $x^n, y^n \rightarrow x^*, y^*$ при $n \rightarrow \infty$.

В силу ограниченности объема статьи наметим краткую схему доказательства сходимости метода. Положим $x = x^*, y = y^*$ в (10) и $x = x^{n+1}, y = y^{n+1}$ в (9). Первые пару уравнений относительно переменной λ из (7), (8) умножим на величины $(\lambda_i^* - \lambda_i^{n+1})$ и $(\lambda_i^{n+1} - \lambda_i^n)$, $i = 1, 2$, тогда после несложных преобразований можно получить оценку

$$\begin{aligned} &\langle \lambda_1^{n+1} - \lambda_1^n, \lambda_1^* - \lambda_1^{n+1} \rangle + \langle \lambda_2^{n+1} - \lambda_2^n, \lambda_2^* - \lambda_2^{n+1} \rangle + \\ &\quad + \langle \bar{\lambda}_1^n - \lambda_1^n, \lambda_1^{n+1} - \bar{\lambda}_1^n \rangle + \langle \bar{\lambda}_2^n - \lambda_2^n, \lambda_2^{n+1} - \bar{\lambda}_2^n \rangle + \\ &\quad + \langle x^{n+1} - x^n, x^* - x^{n+1} \rangle + \langle y^{n+1} - y^n, y^* - y^{n+1} \rangle + \\ &\quad + \langle \bar{x}^n - x^n, x^{n+1} - \bar{x}^n \rangle + \langle \bar{y}^n - y^n, y^{n+1} - \bar{y}^n \rangle + \\ &\quad + \alpha \langle S^T(y^* - \bar{y}^n) + B(x^* - \bar{x}^n), x^* - \bar{x}^n \rangle + \alpha \langle D(y^* - \bar{y}^n) + P(x^* - \bar{x}^n), y^* - \bar{y}^n \rangle + \\ &\quad + \alpha^2 |S^T(y^n - \bar{y}^n) + B(x^n - \bar{x}^n)|^2 + \alpha^2 |D(y^n - \bar{y}^n) + P(x^n - \bar{x}^n)|^2 \geq 0. \end{aligned} \quad (12)$$

Полученное неравенство перепишем в векторно-матричном виде. Введем обозначения: $C = \begin{pmatrix} 0 & S \\ P^T & 0 \end{pmatrix}$, $B^* = \begin{pmatrix} B & 0 \\ 0 & D \end{pmatrix}$, и $v^* = \begin{pmatrix} x^* \\ y^* \end{pmatrix}$, $v = \begin{pmatrix} x \\ y \end{pmatrix}$. $\lambda^* = \begin{pmatrix} \lambda_1^* \\ \lambda_2^* \end{pmatrix}$. $\lambda = \begin{pmatrix} \lambda_1 \\ \lambda_2 \end{pmatrix}$ и перепишем оценку (12) в форме

$$\begin{aligned} & \langle v^{n+1} - v^n, v^* - v^{n+1} \rangle + \langle \bar{v}^n - v^n, v^{n+1} - \bar{v}^n \rangle + \\ & + \langle \lambda^{n+1} - \lambda^n, \lambda^* - \lambda^{n+1} \rangle + \langle \bar{\lambda}^n - \lambda^n, \lambda^{n+1} - \bar{\lambda}^n \rangle + \\ & + \alpha^2 |(C + B^*)(v^n - \bar{v}^n, v^* - \bar{v}^n)|^2 + \alpha^2 (C + B^*) \langle v^* - \bar{v}^n, v^* - \bar{v}^n \rangle \geq 0. \end{aligned}$$

Разлагая скалярные произведения в суммы квадратов, последнее неравенство приведем к виду

$$\begin{aligned} & |v^{n+1} - v^*|^2 + |\bar{v}^n - v^{n+1}|^2 + (1 - 2\alpha^2(|B^* + C|^2 + 1))|v^n - \bar{v}^n|^2 + \\ & + |\lambda^{n+1} - \lambda^*|^2 + |\lambda^n - \bar{\lambda}^*|^2 + |\lambda^{n+1} - \bar{\lambda}^n|^2 \leq |v^n - v^*|^2 + |\lambda^n - \lambda^*|^2. \end{aligned}$$

Просуммируем полученное неравенство от $n = 0$ до $n = N$:

$$\begin{aligned} & |v^{N+1} - v^*|^2 + \sum_{k=0}^{k=N} |v^{k+1} - \bar{v}^k|^2 + d \sum_{k=0}^{k=N} |\bar{v}^k - v^k|^2 + \\ & + |\lambda^{N+1} - \lambda^*|^2 + \sum_{k=0}^{k=N} |\lambda^{k+1} - \bar{\lambda}^k|^2 + d \sum_{k=0}^{k=N} |\bar{\lambda}^k - \lambda^k|^2 \leq |v^0 - v^*|^2 + |\lambda^0 - \lambda^*|^2, \end{aligned}$$

где $d = 1 - 2\alpha^2(|B^* + C|^2 + 1)$. Из полученного неравенства следует ограниченность траектории:

$$|v^{N+1} - v^*|^2 + |\lambda^{N+1} - \lambda^*|^2 \leq |v^0 - v^*|^2 + |\lambda^0 - \lambda^*|^2,$$

а также, сходимость рядов: $\sum_{k=0}^{\infty} |v^{k+1} - \bar{v}^k|^2 < \infty$, $\sum_{k=0}^{\infty} |\bar{v}^k - v^k|^2 < \infty$, $\sum_{k=0}^{\infty} |\lambda^{k+1} - \bar{\lambda}^k|^2 < \infty$, $\sum_{k=0}^{\infty} |\bar{\lambda}^k - \lambda^k|^2 < \infty$, и, следовательно, стремление к нулю величин $|v^{n+1} - \bar{v}^n|^2 \rightarrow 0$, $|\bar{v}^n - v^n|^2 \rightarrow 0$, $n \rightarrow \infty$, $|\lambda^{n+1} - \bar{\lambda}^n|^2 \rightarrow 0$, $|\bar{\lambda}^n - \lambda^n|^2 \rightarrow 0$, $n \rightarrow \infty$. Так как последовательности v^n, λ^n ограничены, то существуют элементы v', λ' , такие, что $v^{n_i} = (x^{n_i}, y^{n_i}) \rightarrow v' = (x', y')$ и $\lambda^{n_i} = (\lambda_1^{n_i}, \lambda_2^{n_i}) \rightarrow \lambda' = (\lambda_1', \lambda_2')$ при $n_i \rightarrow \infty$, и при этом $|v^{n_i+1} - v^{n_i}|^2 = |x^{n_i+1} - x^{n_i}|^2 + |y^{n_i+1} - y^{n_i}|^2 \rightarrow 0$, $|\bar{v}^{n_i} - v^{n_i}|^2 = |\bar{x}^{n_i} - x^{n_i}|^2 + |\bar{y}^{n_i} - y^{n_i}|^2 \rightarrow 0$. В силу доказанного $\lambda_1^{n+1} \rightarrow \lambda_1', \lambda_1^n \rightarrow \lambda_1', \bar{x}^n \rightarrow x'$ и $\lambda_2^{n+1} \rightarrow \lambda_2', \lambda_2^n \rightarrow \lambda_2', \bar{y}^n \rightarrow y'$. Тогда

$$\lambda_1' = \lambda_1' - \alpha(\langle e, x' \rangle - 1), \quad \lambda_2' = \lambda_2' - \alpha(\langle e, y' \rangle - 1),$$

$$x' = x' + \alpha(S^T y' + Bx' + \lambda_1' e), \quad y' = y' + \alpha(Px' + Dy' + \lambda_2' e).$$

Полученная предельная точка x', y', λ' совпадает с x^*, y^*, λ^* и в силу монотонности убывания величин $|x^n - x^*|^2, |y^n - y^*|^2, |\lambda_1^n - \lambda_1^*|^2$ предельная точка может быть только одна. Теорема доказана.

Автор выражает глубокую благодарность А.С.Антипину за постоянное внимание к работе.

Список литературы

- [1] Попова О.А. *Игровая рефлексивная модель процесса обучения.* - Алгоритмический анализ неустойчивых задач. Тез.Докл. Всерос. Конф., Екатеринбург, 2-6 февр. 2004. - С.295 -296.
- [2] Лефевр В.А. *Алгебра совести.* М., "Когито-Центр", 2003.-411
- [3] Антипин А.С. *Градиентный и экстраградиентный подходы в билинейном равновесном программировании.* Москва ВЦ РАН, -2002.-132.

EQUILIBRIUM APPROACH WITH GAME MODELS OF TEACHING PROCESS

O.A. Popova

*Omsk state universiry, Omsk
e-mail: olgaarc@yandex.ru*

Abstract. The paper consider a two-person game with mixed strategies and a solution method for it. The Nash equilibrium is its solution. The game and its solution method are interpreted as static and dynamic model of teaching process. The convergence of training process to optimal solution is proved.

Key words: two-person game, Nash equilibrium, teaching process, convergence.

ДИСКРЕТНАЯ ОПТИМИЗАЦИЯ

ЭФФЕКТИВНЫЕ АЛГОРИТМЫ ДЛЯ ЗАДАЧ О РАЗМЕЩЕНИИ ЦЕНТРОВ С НЕЗАМКНУТЫМ И ЗАМКНУТЫМ МАРШРУТАМИ ОБСЛУЖИВАНИЯ¹

А. А. Агеев

*Институт математики СО РАН, Новосибирск
e-mail: ageev@math.nsc.ru*

Аннотация. Рассматриваются оптимизационные модели, обобщающие классическую задачу о p -центре, — задача о размещении центров с незамкнутым маршрутом обслуживания и задача о размещении центров с замкнутым маршрутом обслуживания. Предлагаются полиномиальные приближенные алгоритмы с оценками точности 3 для обеих задач на произвольных сетях, что значительно улучшает полученные ранее результаты. Предлагается также точный полиномиальный алгоритм для решения задачи о размещении центров с незамкнутым маршрутом обслуживания на цепи при условии одного и того же множества допустимых депо для всех клиентов (известно, что в случае произвольных множеств депо последняя задача NP-трудна).

Ключевые слова: полиномиальный алгоритм, метрическое пространство, сеть, оценка точности, трудоемкость.

Введение

В классической задаче о p -центре задано множество точек (клиентов) в метрическом пространстве и требуется разместить p центров таким образом, чтобы максимальное расстояние между клиентом и ближайшим к нему центром было минимально. В данной работе рассматриваются обобщения этой задачи, в которых с каждым клиентом связано множество депо или распределительных станций, которые ему разрешается использовать. В задаче о размещении центров с незамкнутым маршрутом обслуживания обслуживание клиента состоит в перемещении сервера, находящегося в одном из (сервисных) центров, в допустимое депо, погрузка необходимого материала (например, запчасти) и доставка его клиенту. В задаче о размещении центров с замкнутым маршрутом обслуживания обслуживание клиента также включает обратное перемещение сервера в центр его базирования. В обеих задачах стоимость обслуживания клиента есть линейная функция от расстояния, пройденного обслуживающим сервером, и требуется разместить p центров таким образом, чтобы максимальная стоимость обслуживания была минимальной.

Перейдем теперь к формальным постановкам задач. В рассматриваемых задачах предполагается, что задан неориентированный связный граф $G = (V, E)$, каждому ребру $e \in E$ которого приписана положительная длина l_e . Под ребром, вообще говоря, понимается образ замкнутого интервала при непрерывном взаимнооднозначном отображении, т. е. ребро - жорданова дуга. Для наших целей достаточно считать, что каждое ребро $e = (u, v)$ есть интервал длины l_e с доступными внутренними точками. Обозначим через $A(G)$ континуум точек на ребрах G (заметим, что $A(G)$ - объединение $|E|$ замкнутых интервалов). Длины ребер порождают функцию расстояний на $A(G)$: для любых двух точек $x, y \in A(G)$ положим $d(x, y)$ равным длине кратчайшего пути, соединяющего x и y . Вместе с определенной

¹Работа выполнена при поддержке Российского Фонда Фундаментальных Исследований (проект 05-01-00961-а)

выше функцией расстояний $A(G)$ является метрическим пространством. Назовем $A(G)$ сетью, порожденной G и набором длин ребер l_e . Множество вершин $V = \{v_1, \dots, v_n\}$ рассматривается как множество клиентов. Каждому клиенту v_i приписан неотрицательный вес w_i . Задано также множество $X = \{x_1, \dots, x_m\}$ точек множества $A(G)$, представляющих имеющиеся депо. Каждому клиенту соответствует подмножество доступных депо $X^i \subseteq X$, которые ему позволено использовать.

Требуется выбрать множество p центров $Y = \{y_1, \dots, y_p\}$ в $A(G)$, доставляющее минимальное значение целевой функции $F^1(Y) = \max_{i=1, \dots, n} S_i^1(Y)$ в случае задачи с незамкнутым маршрутом обслуживания и целевой функции $F^2(Y) = \max_{i=1, \dots, n} S_i^2(Y)$ в случае задачи с замкнутым маршрутом обслуживания, где

$$S_i^1(Y) = \min_{y_j \in Y, x_k \in X^i} \{d(y_j, x_k) + d(x_k, v_i)\},$$

$$S_i^2(Y) = \min_{y_j \in Y, x_k \in X^i} \{d(y_j, x_k) + d(x_k, v_i) + d(v_i, y_j)\}.$$

Наряду с сформулированными выше непрерывными постановками задач мы рассматриваем также их дискретные варианты, в которых выбираемое множество центров обязано быть подмножеством множества вершин V . Первые мы будем называть непрерывными моделями, а вторые дискретными. Очевидно, что обе модели совпадают с классической задачей о p -центре (соответственно, в непрерывной и дискретной постановке), если $v_i \in X^i$ для каждого $v_i \in V$.

Модель с замкнутым маршрутом обслуживания в случае, когда $X^i = X$ для всех $v_i \in V$, впервые рассматривалась в работе [1]. В этой работе изучается только случай $p = 1$ и обсуждаются некоторые приложения модели. Более сильные результаты для обеих моделей были получены в [5]. В последней работе, в частности, установлено, что задача с незамкнутым маршрутом обслуживания NP-трудна даже в случае, когда граф G - цепь, в то время как задача с замкнутым маршрутом обслуживания в случае, когда G - дерево, может быть решена полиномиальным алгоритмом. В [5] также построены полиномиальные приближенные алгоритмы для обеих задач на произвольных сетях — с оценками точности 9 и 12 для незамкнутой и замкнутой моделей соответственно. В настоящей работе предлагаются полиномиальные алгоритмы с оценками точности 3 для решения обеих задач на произвольных сетях. Кроме того, предлагается алгоритм с оценкой времени работы $O((m+n) \log^2(m+n))$ для модели с незамкнутыми маршрутами обслуживания в случае, когда G — цепь и $X^i = X$ для всех $v_i \in V$.

1. Приближенные алгоритмы

В этом разделе исследуемые задачи рассматриваются в общей постановке. Ввиду их труднорешаемости мы опишем общий метод построения приближенных алгоритмов с оценками точности 3 для обеих задач. А именно, мы построим сводимости этих задач к задаче о p поставщиках.

Задача о p поставщиках: Пусть $H = (V \cup U, F)$ — полный неориентированный двудольный граф с долями $V = \{v_1, \dots, v_n\}$ и $U = \{u_1, \dots, u_t\}$ и множеством ребер F . Ребрам графа H приписаны положительные длины, индуцированные некоторой метрикой на множестве $V \cup U$. Для каждой пары вершин $x, y \in V \cup U$, через $d'(x, y)$ обозначим длину кратчайшего пути, связывающего x и y . Каждой вершине $v_i \in V$ приписан неотрицательный вес w_i . Требуется найти множество $S \subseteq U$, $|S| = p$, минимизирующее $\max_{v_i \in V} \{w_i \min_{u_j \in S} d'(v_i, u_j)\}$.

Полиномиальные приближенные алгоритмы с оценками точности 3 для различных вариантов этой задачи можно найти в [2], [6] (в этих работах такие алгоритмы предлагаются даже для более общего случая, когда заданы стоимости размещения центров и верхняя граница на суммарную стоимость).

Мы вначале рассмотрим дискретные варианты моделей ($Y \subseteq V$, $|Y| = p$) в общей постановке. В этом случае без ограничения общности можно считать, что длины ребер удовлетворяю неравенству треугольника. В частности, для любой пары вершин $x, y \in V$ длина ребра (x, y) равна $d(x, y)$, кратчайшему расстоянию между x и y .

Определим двудольный граф $H = (V \cup U, F)$ следующим образом: V — множество вершин графа G , а $U = \{u_1, \dots, u_n\}$ — его копия, причем u_i соответствует v_i , $i = 1, \dots, n$. Каждой вершине $v_i \in V$ припишем тот же вес w_i , что она имеет в G . Далее, для каждой пары $v_i \in V$, $u_j \in U$ определим длину ребра (v_i, u_j) в зависимости от рассматриваемой модели следующим образом. Для задачи с незамкнутым маршрутом обслуживания длина ребра (v_i, u_j) определяется как $d_1(v_i, u_j) = \min_{x_k \in X^i} \{d(v_i, x_k) + d(x_k, v_j)\}$, а для задачи с замкнутым маршрутом как

$$d_2(v_i, u_j) = \min_{x_k \in X^i} \{d(v_i, x_k) + d(x_k, v_j)\} + d(v_j, v_i) = d_1(v_i, u_j) + d(v_j, v_i).$$

Ясно, что исходные индивидуальные задачи эквивалентны построенным выше индивидуальным задачам о p поставщиках. Для получения 3-приближенных алгоритмов остается показать, что длины ребер $d_1(v_i, u_j)$ и $d_2(v_i, u_j)$ удовлетворяют неравенству треугольника.

Лемма 1. Длины ребер $\{d_1(v_i, u_j)\}$ и $\{d_2(v_i, u_j)\}$, определенные на полном двудольном графе $H = (V \cup U, F)$ удовлетворяют неравенству треугольника.

Доказательство. Рассмотрим сначала набор длин $\{d_1(v_i, u_j)\}$. Для каждой пары $v_r \in V$, $u_q \in U$ определим $x_{k(r,q)} \in X^r$ следующим равенством:

$$d_1(v_r, u_q) = \min_{x_k \in X^r} \{d(v_r, x_k) + d(x_k, v_q)\} = d(v_r, x_{k(r,q)}) + d(x_{k(r,q)}, v_q).$$

Пусть $P = (v_i, u_{j(1)}, v_{j(1)}, u_{j(2)}, v_{j(2)}, \dots, u_{j(t)}, v_{j(t)}, u_j)$ — путь в H . Покажем, что $d_1(v_i, u_j)$ не превосходит $d_1(P)$, суммарной длины P в H (относительно метрики d_1). Рассмотрим путь

$$P^* = (x_{k(i,j(1))}, u_{j(1)}, x_{k(i(1),j(1))}, v_{i(1)}, \dots, u_{j(t)}, x_{k(i(t),j(t))}, v_{i(t)}, x_{k(i(t),j)}, u_j)$$

в G и пусть $d(P^*)$ обозначает его длину в G (для того, чтобы увидеть, что P^* — действительно путь в G достаточно заменить каждую вершину из U ее оригиналом в G). По определению $d_1(P) = d(v_i, x_{k(i,j(1))}) + d(P^*)$. Так как d удовлетворяет неравенству треугольника в G , мы имеем $d(x_{k(i,j(1))}, u_j) \leq d(P^*)$, откуда и получаем требуемое

$$d_1(P) \geq d(v_i, x_{k(i,j(1))}) + d(x_{k(i,j(1))}, v_j) \geq d(v_i, v_j) = d_1(v_i, u_j).$$

Теперь рассмотрим набор длин $\{d_2(v_i, v_j)\}$. Как и выше нам нужно показать, что для любого пути $P = (v_i, u_{j(1)}, v_{j(1)}, u_{j(2)}, v_{j(2)}, \dots, u_{j(t)}, v_{j(t)}, u_j)$ в H $d_2(v_i, u_j)$ не превосходит $d_2(P)$, длину P в H (относительно метрики d_2). Действительно, выше мы установили, что $d_1(P) \geq d_1(v_i, u_j)$ и, кроме того, $d(P) \geq d(v_i, u_j)$. Согласно определениям d_1 и d_2 , $d_2(v_i, u_j) = d_1(v_i, u_j) + d(v_i, u_j)$, откуда $d_2(P) = d_1(P) + d(P)$. Следовательно,

$$d_2(v_i, u_j) = d_1(v_i, u_j) + d(v_i, u_j) \leq d_1(P) + d(P) = d_2(P).$$

Те же результаты имеют место также и для взвешенных непрерывных вариантов обеих задач, поскольку непрерывные модели нетрудно дискретизировать аналогично тому, как это делается в классической задаче о p -центре.

2. Полиномиальный алгоритм для задачи о размещении центров с незамкнутыми маршрутами обслуживания на цепи

Пусть $G = (V, E)$ — цепь (путь) с $V = \{v_1, \dots, v_n\}$, $E = \{(v_1, v_2), (v_2, v_3), \dots, (v_{n-1}, v_n)\}$ и пусть $X = \{x_1, \dots, x_m\} \subseteq A(G)$ — множество депо (мы рассматриваем непрерывную модель, но излагаемый результат, очевидно, справедлив и для дискретной). Без ограничения общности можно рассматривать V , X , $Y = \{y_1, \dots, y_p\}$ как множества точек на вещественной прямой, причем $v_1 < \dots < v_n$, $x_1 < \dots < x_m$ и $y_1 \leq \dots \leq y_p$. При фиксированном положительном r , рассмотрим задачу, в которой требуется определить, существует ли множество точек Y такое, что $|Y| \leq p$ и $S_i^1(Y) \leq r_i = r/w_i$.

Описание алгоритма. Для простоты записи к исходному множеству X добавим пару несущественных дополнительных точек $x_0 = -\infty$ and $x_{m+1} = +\infty$. Тогда при любом i $v_i \in [x_{k(i)}, x_{k(i)+1}]$, где $k(i) \in \{0, 1, \dots, m+1\}$ — некоторая однозначно определенная точка. Заметим, что задача эквивалентна случаю, когда $X^i = \{x_{k(i)}, x_{k(i)+1}\}$, поскольку использование клиентом i других депо может только увеличить стоимость его обслуживания. Рассматриваемая задача сводится к задаче о минимальном точечном покрытии множеств $\{T_i(r)\}$ следующего вида: $T_i(r) = T_i^-(r) \cup T_i^+(r)$, где

$$T_i^-(r) = \{y \mid |x_{k(i)} - v_i| + |y - x_{k(i)}| \leq r/w_i\},$$

$$T_i^+(r) = \{y \mid |x_{k(i)+1} - v_i| + |y - x_{k(i)+1}| \leq r/w_i\},$$

— интервалы с центральными точками $x_{k(i)}$ и $x_{k(i)+1}$ соответственно.

Мы представим полиномиальный алгоритм для более общей задачи о покрытии множеств: задано множество точек $x_1 < \dots < x_m$ (назовем их депо) на вещественной прямой и совокупность непустых множеств $T_i = T_i^- \cup T_i^+$ ($i = 1, \dots, n$) (назовем их 2-интервалами), где T_i^- и T_i^+ — интервалы (возможно, один из них пустой) с центральными точками $x_{k(i)}$ и $x_{k(i)+1}$ соответственно. Найти множество точек Y минимальной мощности, покрывающее $\{T_i\}$.

Описываемый алгоритм находит множество $Y = \{y_1, \dots, y_q\}$ ($y_1 < \dots < y_q$) и состоит из $|X| = m$ однотипных шагов. На первом шаге алгоритм рассматривает подмножество 2-интервалов T_i таких, что x_1 является центром T_i^- . Определим $I_1 = \{i \mid x_{k(i)} = x_1, x_1 \in T_i^-\}$ и $I_1^* = \{i \mid i \in I_1, T_i^+ = \emptyset\}$. Если $I_1^* \neq \emptyset$, то мы можем удалить все интервалы T_i , $i \in I_1^*$ за исключением кратчайшего. Далее через $|T|$ будет обозначаться длина интервала T .

Случай 1: $I_1^* = \emptyset$. Определим $i' \in I_1$ равенством $|T_{i'}^+| = \min\{|T_i^+| : i \in I_1\}$. Удаляем x_1 и все 2-интервалы T_i , $i \in I_1 \setminus \{i'\}$. Полагаем $T_{i'} = T_{i'}^+$ и переходим к следующему шагу, где роль x_1 играет x_2 .

Предположим, что $I_1^* = \{i^*\}$. В этом случае мы должны выбрать y_1 в $T_{i^*}^-$.

Случай 2: $x_2 \in T_{i^*}^-$. Заменяем интервал $T_{i^*}^-$ интервалом $T_{i^*}^+$ с центром в x_2 и длиной $2(|I_{i^*}^-|/2 - x_2 + x_1)$. Далее делаем то же, что в случае 2 и переходим к следующему шагу.

Случай 3: $x_2 \notin T_{i^*}^-(x_1)$. Полагаем y_1 равным наибольшей точке, покрывающей все 2-интервалы T_i , $i \in I_1$, т. е. $y_1 := \max\{y : y \in T_i, i \in I_1\}$. Удаляем все 2-интервалы из I_1 . Определяем I_2 как множество оставшихся 2-интервалов, одна из компонент которых имеет центром x_2 . Удаляем x_1 и все 2-интервалы из I_2 , покрытые y_1 . Переходим к следующему шагу.

Заметим, что в предположении, что множество X заранее упорядочено, т. е. $x_1 < \dots < x_m$, описанный алгоритм может быть реализован с временной сложностью $O(m + n \log n)$. Перейдем теперь к обоснованию корректности алгоритма.

Лемма 2. Множество Y - минимальное покрытие.

Доказательство. Случай 1-2. Если $I_1^* = \emptyset$ или $I_1^* = \{i^*\}$ и $x_2 \in T_{i^*}^-$, то, очевидно, существует минимальное покрытие с наименьшей точкой y_1^* , удовлетворяющей $y_1^* \geq x_2$. В этих случаях задача сводится к задаче с множеством депо $\{x_2, \dots, x_m\}$.

Случай 3. Пусть $Y^* = \{y_1^*, \dots, y_q^*\}$ — минимальное покрытие с наименьшей точкой y_1^* . Покажем, что точка y_1^* может быть выбрана как точка y_1 , определенная в рассматриваемом случае. Предположим сначала, что $y_1^* < y_1 < x_2$. Тогда для любого $i \notin I_1$ $i^* \in T_i$ влечет $y_1 \in T_i$. Поскольку $y_1 \in T_i$ для всех $i \in I_1$, то отсюда следует, что $\{y_1, y_2^*, \dots, y_q^*\}$ также является минимальным покрытием. Пусть теперь $y_1 < y_1^*$. Из максимальной y_1 следует, что существует $i' \in I_1 \setminus \{i^*\}$ такой, что $y_1^* \notin T_{i'}$ и тем самым найдется $y_j^* > y_1^*$ такой, что $y_j^* \in T_{i'}^+$. Кроме того, поскольку $T_{i'}^+$ имеет центром x_2 , то для каждого $i \notin I_1$ такого, что $y_1^* \in T_i$, имеем $T_{i'}^+ \subseteq T_i$; в частности, $y_j^* \in T_i$. Таким образом, множество $\{y_1, y_2^*, \dots, y_q^*\}$ покрывает все 2-интервалы $\{T_i\}$, $i = 1, \dots, n$.

Нетрудно показать, что с использованием описанного выше алгоритма и поисковых алгоритмов из [3], [4] исходная задача может быть решена за время $O((m+n) \log^2(m+n))$.

Список литературы

- [1] O. Berman, Z. Drezdner, G. O. Wesolowsky *The collection depots location problems on networks*. - Naval Research Logistics **49** (2002), 15–24.
- [2] R. Bhatia, S. Guha, S. Khuller, Y. J. Sussmann *Facility location with dynamic distance function*. - J. of Combinatorial Optimization **2** (1998), 199–217.
- [3] G. N. Frederickson, D. B. Johnson *The complexity of selection and ranking in $X + Y$ and matrices with sorted columns*. - J. Comput. System Science **24** (1982), 197–208.
- [4] N. Megiddo, A. Tamir, E. Zemel, R. Chandrasekaran *An $O(n \log^2 n)$ algorithm for the k -th longest path in a tree with applications to location problems*. - SIAM J. Computing **10**, 1981, 328–337.
- [5] A. Tamir, N. Halman *One-way and round-trip center location problems*. - to appear in Discrete Applied Mathematics.
- [6] S. Khuller, R. Pless, Y. J. Sussmann *Fault tolerant K -center problems*. - Theoretical Computer Science **242** (2000), 237–245.

EFFICIENT ALGORITHMS FOR THE ONE-WAY AND ROUND-TRIP CENTER LOCATION PROBLEMS

A. A. Ageev

Sobolev Institute of Mathematics, Novosibirsk

e-mail: ageev@math.nsc.ru

Abstract. We consider the one-way and round-trip center location problems, which are extensions of the classical p -center problem. We present polynomial-time 3-approximation algorithms for both problems on general networks, which significantly improves the known results. We also develop a polynomial algorithm for solving the one-way problem on path networks in the case when each client has the same set of feasible depots (this problem is known to be NP-hard when the sets of feasible depots are arbitrary).

Key words: polynomial-time algorithm, metric space, network, approximation ratio, running time.

ПОВЕДЕНИЕ В СРЕДНЕМ АЛГОРИТМА ЛОКАЛЬНОГО СПУСКА ДЛЯ ЗАДАЧИ О P -МЕДИАНЕ¹

Е.В. Алексеева

Институт математики им. С.Л. Соболева СО РАН, Новосибирск
e-mail: ekaterina2@math.nsc.ru

Аннотация. Для задачи о p -медиане исследовано поведение в среднем стандартного алгоритма локального спуска. Показано, что среднее число шагов для достижения локального минимума растет не быстрее линейной функции от n , (n — число предприятий) при $p = const$ и не быстрее квадратичной функции при $p = [\alpha n]$, $0 < \alpha < 1$. Полученные оценки не зависят от применяемых правил выбора. Приведены результаты численных экспериментов, показывающие влияние правил выбора на относительную погрешность получаемых локальных минимумов, число шагов алгоритма и время счета.

Ключевые слова: Задачи размещения, PLS-полные задачи, локальный спуск, поведение в среднем.

Введение

В задаче о p -медиане заданы множество предприятий $I = \{1, \dots, n\}$, множество клиентов $J = \{1, \dots, m\}$, матрица затрат на обслуживание клиентов (g_{ij}) , $i \in I$, $j \in J$ и натуральное число $p < n$. Требуется найти подмножество $S \subset I$ мощности p , на котором достигается минимум целевой функции

$$F(S) = \sum_{j \in J} \min_{i \in S} g_{ij}.$$

Эта задача является NP -трудной в сильном смысле. Более того, если для нее существует приближенный полиномиальный алгоритм, относительная погрешность которого может быть ограничена константой, то $P = NP$. Эти наблюдения стимулируют разработку неполномиальных алгоритмов, среди которых наибольшей эффективностью на сегодняшний день отличаются алгоритмы локального поиска [3]. Многие из них используют процедуры локального спуска по одной или нескольким окрестностям, концентрируя внимание только на локальных оптимумах.

Под окрестностью $Swap(S)$ будем понимать множество допустимых решений задачи, получающихся из S удалением одного элемента и добавлением другого из дополнения $I \setminus S$. Мощность окрестности равна $p(n - p)$, то есть она линейна при фиксированном p и квадратична при $p = [\alpha n]$, $0 < \alpha < 1$.

Стандартный алгоритм локального спуска начинает свою работу с произвольного допустимого решения S_0 . На i -м шаге алгоритма в окрестности решения S_i выбирается решение S' с меньшим значением целевой функции, $F(S') < F(S_i)$. Если такого решения нет, то S_i является локальным минимумом и алгоритм заканчивает свою работу. В противном случае выполняется следующий шаг с $S_{i+1} = S'$. Число шагов такой процедуры и отклонение получаемых локальных минимумов от глобального является предметом данного исследования.

¹Работа выполнена при поддержке РФФИ (проект 04-07-90096)

Известно [1], что задача о p -медиане с окрестностью $Swap$ является PLS -полной. Это значит, что существование полиномиального алгоритма нахождения локального оптимума для данной окрестности ведет к существованию аналогичного алгоритма для целого ряда задач локального поиска [6], что маловероятно. Однако, численные эксперименты показывают, что в среднем, число шагов стандартного алгоритма локального спуска растет линейно или квадратично с ростом n [5]. Оценки поведения в худшем случае и в среднем разительно отличаются друг от друга. Данная работа посвящена исследованию этого феномена для задачи о p -медиане.

1. Оценка в среднем числа шагов алгоритма

Предположим, что элементы (g_{ij}) , $i \in I$, $j \in J$ являются действительными числами и выбираются с равномерным распределением из некоторого отрезка $[a, b]$, $a < b$. Простым путем длины k будем называть последовательность (S_0, \dots, S_k) попарно различных решений, где $S_{i+1} \in Swap(S_i)$, $i = 0, \dots, k - 1$. Простой путь будем называть путем с улучшением, если $F(S_{i+1}) < F(S_i)$, для всех $i = 0, \dots, k - 1$.

Поскольку элементы g_{ij} — действительные числа, выбранные случайным образом, то можно считать, что все значения целевой функции различны. Если их упорядочить по возрастанию, то эти порядки будут равновероятны.

Теорема 1. *Для данного правила выбора элементов g_{ij} , $i \in I$, $j \in J$ и $p = [\alpha n]$, $0 < \alpha < 1$, среднее число шагов стандартного алгоритма локального спуска с окрестностью $Swap$ не превосходит $\frac{3}{2}en^2$, где e — основание натурального логарифма.*

Условие равномерного распределения на отрезке $[a, b]$ не является существенным. Теорема верна для любого распределения, гарантирующего равновероятные упорядочения допустимых решений по неубыванию целевой функции [4].

Теорема 2. *Для данного правила выбора элементов g_{ij} , $i \in I$, $j \in J$ и фиксированном значении параметра p , среднее число шагов стандартного алгоритма локального спуска с окрестностью $Swap$ не превосходит $\frac{3}{2}en$.*

Изменим правило формирования матрицы (g_{ij}) . Будем считать, что с вероятностью q каждый ее элемент становится равным $+\infty$, а с вероятностью $(1 - q)$ — выбирается по старым правилам. Отношение числа небесконечных элементов к общему числу элементов матрицы называют плотностью. Обозначим ее через ρ . При $\rho = 1$ получаем старое правило выбора элементов g_{ij} .

Матрицы с малой плотностью называют разреженными. Такие матрицы приводят к большому разрыву двойственности и требуют от метода ветвей и границ большого числа итераций.

Следствие 3. *При $\rho < 1$ теоремы 1 и 2 остаются справедливыми.*

Утверждение остается справедливым и при других способах порождения разреженных матриц. Например, в [5] рассматривались матрицы, в которых число небесконечных элементов в каждой строке и(или) каждом столбце было одинаковым.

2. Выбор направления спуска

Выделим в множестве $Swap(S)$ подмножество $Swap^*(S) = \{S' \in Swap(S) \mid F(S') < F(S)\}$ решений с меньшим чем в S значением целевой функции. Если на каждом шаге локального спуска в множестве $Swap^*(S)$ выбирается решение с наименьшим значением целевой функции, то такое правило называют *наискорейшим* спуском. Такой выбор кажется наиболее естественным. Однако существуют и другие правила. Их применение сильно влияет на поведение алгоритма.

В некотором смысле противоположным правилом является выбор *наихудшего* элемента в множестве $Swap^*(S)$. Такая стратегия дает наиболее пологий спуск к локальному минимуму, но может потребовать значительно больше шагов, чем в предыдущем случае. При этом мы получим, вообще говоря, разные локальные минимумы. Логично предположить, что более пологий спуск приведет к меньшему значению целевой функции. Как мы увидим ниже, разница в этих значениях может оказаться достаточно большой.

Правило *случайный* элемент предполагает выбор элемента из множества $Swap^*(S)$ случайным образом, например, с равномерным распределением. Для реализации этого и двух предшествующих правил требуется просмотр всей окрестности. Правило *первый* подходящий лишено этого недостатка. Оно предписывает остановку просмотра как только обнаружен первый элемент из $Swap^*(S)$. По смыслу это правило близко к правилу *случайный* элемент, если просмотр окрестности начинается со случайной точки. Однако чаще всего процедуры просмотра окрестности используют один и тот же порядок, например, лексикографический, и начинают с наименьшего элемента. Если же начинать просмотр с того места, где остановились на предшествующем шаге и заканчивать просмотр на первом же элементе из множества $Swap^*(S)$, то получим *круговое* правило [2]. В его пользу можно привести следующие соображения. При переходе от одной окрестности к другой, решения меняются незначительно. Скорее всего, невыгодные решения в одной окрестности будут невыгодными и в следующей окрестности. Поэтому лучше продолжить просмотр, чем повторять его сначала. Конечно, данные соображения нужно проверять для каждой задачи отдельно.

Заметим, что сформулированные в предыдущем параграфе утверждения касались любого правила выбора элемента из множества $Swap^*(S)$. В следующем параграфе будет показана разница между этими правилами и даны практические рекомендации по их использованию.

3. Эмпирические исследования

Численные эксперименты проводились на примерах с $a = 0$, $b = 1000$, $n = m$, $p = \lfloor n/2 \rfloor$ и $p = 15$. Цель экспериментов состояла в выяснении реальной зависимости числа шагов локального спуска от размерности задачи при различных правилах выбора. Кроме того, подсчитывалась средняя относительная погрешность получаемых локальных оптимумов и среднее время счета. Так как трудоемкость одного шага меняется при смене правил замещения, то прямой корреляции между числом шагов и временем счета нет.

На рисунке 1 показана зависимость числа шагов локального спуска от размерности задачи. Для каждого n случайным образом генерировались 10 тестовых примеров и для каждого из них порождалось 1000 стартовых точек. На графике каждая точка показывает среднее значение для 10000 испытаний. При любом p правило *наихудший* дает значительно большее число шагов локального спуска. Правило *наискорейший* спуск

дает наименьшее число шагов. *Первый, случайный* и *круговой* дают примерно одинаковые результаты. Их число шагов немного больше, чем у *наискорейшего* спуска. Интересно отметить, что для всех правил число шагов растет не быстрее линейной функции, что дает надежду на усиление теоремы 1.

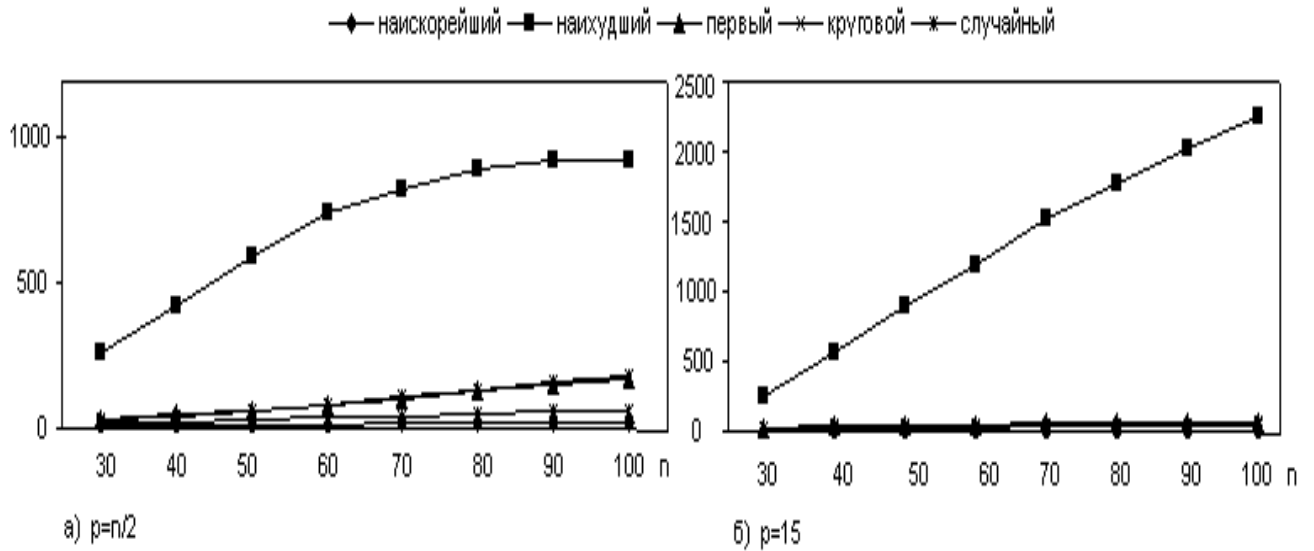


Рис. 1: Среднее число шагов

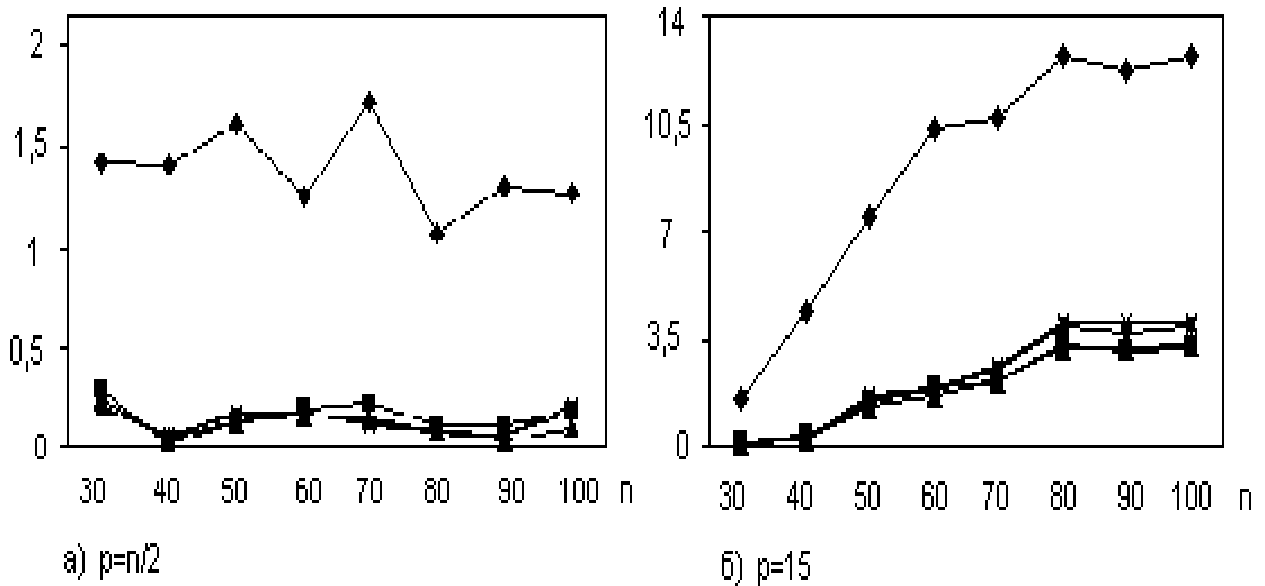


Рис. 2: Средняя относительная погрешность

На рисунке 2 показана средняя относительная погрешность для разных правил замещения. *Наискорейший* спуск дает значительно худшие результаты в обоих случаях, $p = [n/2]$, $p = 15$. Погрешность остальных алгоритмов примерно одинаковая. Таким образом, применение *наискорейшего* спуска дает наибольшую погрешность, но требует наименьшего числа шагов. Правило *наихудший* несмотря на значительно большее число

шагов приводит в среднем к тем же погрешностям, что и правила *первый*, *случайный*, *круговой*. Так как *первый* и *круговой* имеют наименьшую трудоемкость одного шага, то именно эти правила целесообразно использовать на практике. Они мало отличаются друг от друга: чуть большее число шагов и меньшая погрешность у правила *первый* подходящий.

Суммарное время на получение локального оптимума меньше всего у правила *наискорейшего* спуска. Разница составляет 4–5 раз по сравнению с правилами *первый* подходящий или *круговой*. Таким образом, для сравнения этих правил по относительной погрешности нужно брать лучшее решение из 5-ти для *наискорейшего* спуска против одного решения, полученного по двум другим правилам. Однако даже в этих условиях правило *наискорейшего* спуска приводит к большей относительной погрешности.

Список литературы

- [1] Ю.А. Кочетов, М.Г. Пащенко, А.В. Плясунов *Локальный поиск в задаче о p-медиане* - Дискретный анализ и исследование операций, Материалы конференции, Новосибирск, ИМ СО РАН, 2004, с. 186.
- [2] Х. Пападимитриу, К. Стайглиц *Комбинаторная оптимизация. Алгоритмы и сложность* - Москва, Мир, 1985.
- [3] С. С. Ribeiro, P. Hansen *Essays and surveys in metaheuristics* - Kluwer Academic Publishers, 2002.
- [4] С. А. Tovey *Local improvement on discrete structures* - E. Aarts and J.K. Lenstra (eds.) *Local Search in Combinatorial Optimization*, 1997, Chichester: Wiley, p.19-55.
- [5] Yu. Kochetov, E. Alexeeva *Large Neighborhood Search for the p-Median Problem* - Proceedings SYM-OP-IS 2003, Herceg-Noví, Montenegro, 2003, p. 19-21.
- [6] M. Yannakakis *Computational complexity* - E. Aarts and J.K. Lenstra (eds.) *Local Search in Combinatorial Optimization*, 1997, Chichester: Wiley, p.19-55.
- [7] <http://math.nsc.ru/AP/benchmarks/index.html>

AVERAGE BEHAVIOR OF THE LOCAL IMPROVEMENT ALGORITHM FOR THE P -MEDIAN PROBLEM

E. Alekseeva

Sobolev Institute of Mathematics, Novosibirsk

e-mail: ekaterina2@math.nsc.ru

Abstract. For the p -median problem we study behavior of the standard local improvement algorithm in average case. It is shown that average number of steps grows at most as linear function of n (n is number of facilities) if $p = \text{const}$ and at most as a quadratic function if $p = [\alpha n]$, $0 < \alpha < 1$. The results are true for arbitrary pivoting rules. We present computational results which show the influence of the pivoting rules on the average relative error, number of steps, and running time of the algorithm.

Key words: Facility location problems, PLS-complete problem, local search, average behavior.

КРИТИЧЕСКИЕ ПАРАМЕТРЫ ПРОСТЕЙШЕЙ ЗАДАЧИ РАЗМЕЩЕНИЯ¹

Е.В. Алексеева, Ю.А. Кочетов, Н.А. Кочетова

*Институт математики им. С.Л. Соболева СО РАН, Новосибирск
e-mail: ekaterina2@math.nsc.ru, jkochet@math.nsc.ru, nkochet@math.nsc.ru*

Аннотация. Для простейшей задачи размещения производства проведен анализ вычислительной сложности в зависимости от плотности матрицы производственно-транспортных затрат и разброса в затратах на открытие предприятий. Показано, что трудные в вычислительном отношении примеры появляются только при малой плотности и равных затратах на открытие предприятий.

Ключевые слова: простейшая задача размещения, метод ветвей и границ, локальный поиск, разрыв двойственности.

Введение

В области целочисленной и комбинаторной оптимизации наблюдается быстрый рост числа работ, посвященных методам локального поиска. Все более и более сложные модели удается анализировать этими методами в различных разделах исследования операций. Однако авторы часто предлагают метод решения задачи и результаты численных экспериментов, но не объясняют, почему этот метод хорошо работает, всегда ли он будет давать хорошие результаты, каковы границы его применения и как следует менять управляющие параметры при переходе от одного класса исходных данных к другому. Как правило, в задаче имеется один или несколько критических параметров, изменение которых сильно влияет на поведение численных методов. Например, в задаче MAX-2SAT [6] таким критическим параметром является отношение числа конъюнкций к числу переменных. Увеличение этого параметра приводит сначала к росту трудоемкости получения точного решения, а затем к его падению. Максимум достигается при значении 4,5. При больших и меньших значениях параметра задача становится значительно проще. Таким образом, имеется возможность еще до решения задачи предсказать ее трудоемкость.

В настоящей работе с этих позиций исследуется простейшая задача размещения производства. Показано, что плотность матрицы транспортных затрат и разброс в стоимостях открытия предприятий являются критическими параметрами задачи. Малая плотность и отсутствие разброса приводят к наиболее трудным примерам. Огромная трудоемкость при плотности 5% объясняется большим разрывом двойственности, малыми областями обслуживания предприятий и случайной структурой матрицы производственно-транспортных затрат. Методы локального поиска, например, поиск с запретами, часто ошибаются в этом случае, и для повышения их эффективности требуются специальные приемы: рандомизация окрестности, процедуры интенсификации и диверсификации и др. Визуализация процесса поиска помогает понять слабые стороны локальных методов и добиться их большей эффективности даже для трудных в вычислительном отношении примеров.

¹Работа выполнена при поддержке РФФИ (проект 04-07-90096)

1. Постановка задачи

В простейшей задаче размещения производства (ПЗРП) задано множество клиентов $J = \{1, \dots, n\}$ и множество предприятий $I = \{1, \dots, m\}$. Для каждого предприятия $i \in I$ известны стоимость его открытия $f_i \geq 0$ и производственно-транспортные затраты g_{ij} на обслуживание клиентов $j \in J$. Требуется найти непустое подмножество предприятий $S \subseteq I$, которое позволит обслужить всех клиентов с минимальными суммарными затратами. Исследованию этой комбинаторной задачи посвящены десятки статей. Подкупающая простота постановки задачи не мешает ей, как впрочем и многим другим классическим комбинаторным задачам, принадлежать классу задач NP-трудных в сильном смысле. Задача о p -медиане, задача о покрытии множествами, задача минимизации псевдодобулевых полиномов тесно связаны с ПЗРП. Установлено, что существование приближенных полиномиальных алгоритмов с гарантированной оценкой точности меньше чем $0,5 \ln n$ влечет $NP \subset TIME(n^{\log \log n})$, что скорее всего не верно. Таким образом, относительная погрешность в худшем случае любого полиномиального алгоритма для ПЗРП растет не медленнее логарифма от n .

Исключением является метрический частный случай, когда матрица производственно-транспортных затрат является неотрицательной, симметричной и удовлетворяет неравенству треугольника. В этом случае задача остается NP-трудной, но погрешность приближенных полиномиальных алгоритмов может быть ограничена константой. Наилучшие результаты в этой области связаны с применением жадных алгоритмов и округлением оптимального решения линейной релаксации [7]. Значение 1,52 является рекордным на сегодняшний день. Заметим, что величина 1,463 является порогом, ниже которого эта оценка опуститься не может. Складывается впечатление, что даже в метрическом случае решить задачу с относительной погрешностью менее 50 % представляется почти невозможным. К счастью, это не так. Оценки поведения алгоритма в худшем случае дают излишне пессимистическую картину. Например, в [3] приводятся результаты численных экспериментов для метрических задач большой размерности. Предлагаемые авторами полиномиальные алгоритмы дают в среднем около 0,5 % относительной погрешности. Более того, точное решение задачи можно получать модифицированным методом ветвей и границ [4] при $n = m \approx 3000 - 5000$, что покрывает большую часть приложений.

Метрический случай является узким подклассом ПЗРП. Во-первых, матрица (g_{ij}) кроме транспортной составляющей, которая может удовлетворять неравенству треугольника, имеет еще и производственную составляющую, связанную с затратами на выпуск продукции предприятия. Предполагать, что эти затраты удовлетворяют неравенству треугольника, было бы не логичным. Во-вторых, в метрическом случае неявно считается, что каждое предприятие может обслужить любого клиента. Это не всегда так. Например, в задачах стандартизации [1] каждое предприятие имеет свое подмножество потенциальных клиентов, что порождает запрещенные элементы в матрице (g_{ij}) и приводит к понятию плотности. Ниже будет показано, что этот параметр играет ключевую роль в исследовании трудоемкости решения ПЗРП.

2. Легкие и трудные примеры ПЗРП

Рассмотрим частный случай метрической задачи, когда элементы матрицы (g_{ij}) порождаются евклидовыми расстояниями на двумерной плоскости. На квадрате со стороной a выберем n точек случайным образом независимо друг от друга с равномерным распре-

делением. Положим $I = J = \{1, \dots, n\}$ и определим величину g_{ij} как евклидово расстояние между точками $i, j \in I$. Начальные затраты f_i на открытие предприятия будем выбирать из отрезка $[b, c]$ случайным образом независимо с равномерным распределением. Далее, для каждого предприятия упорядочим клиентов по неубыванию производственно-транспортных затрат и будем считать, что d процентов от общего числа клиентов, имеющих минимальные значения g_{ij} , составляют подмножество потенциальных клиентов данного предприятия. Остальные клиенты не могут обслуживаться этим предприятием. Для них положим $g_{ij} = +\infty$. Величину d будем называть плотностью матрицы (g_{ij}) .

В [5] показано, что значение $d = 100\%$ соответствует простым примерам, которые легко решаются как точными, так и приближенными методами. Там же приводятся классы трудных примеров ПЗРП. Они имеют малую плотность, $d \approx 10\% - 16\%$. На рис. 1 показано изменение среднего числа шагов метода ветвей и границ как функции от d . Интервал от

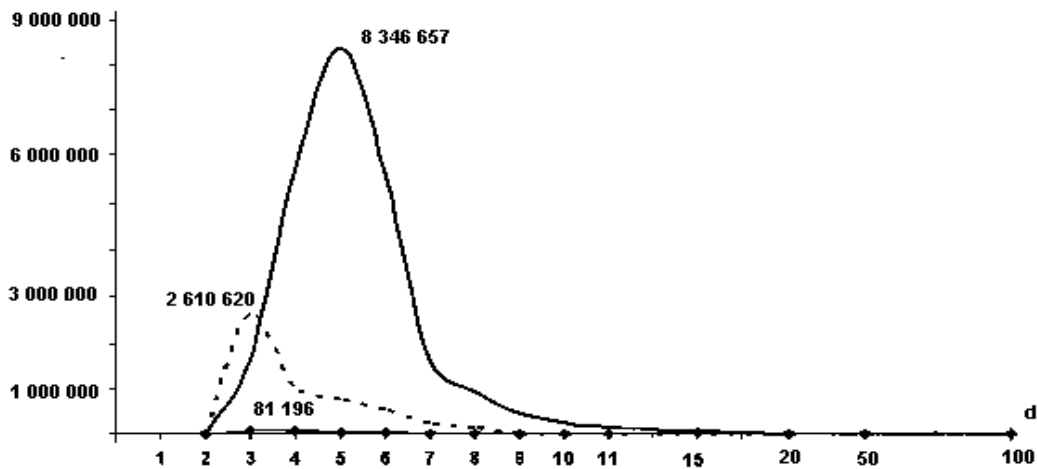


Рис. 1: Среднее число шагов метода ветвей и границ

4% до 6% соответствует наиболее трудным примерам при $a = 500, b = c = 1000, n = 100$. При $d > 20\%$ примеры оказываются легкими и не представляют проблем для численных методов. Пунктирная кривая на рис.1 также соответствует среднему числу шагов метода ветвей и границ но при другом способе порождения запретов в матрице (g_{ij}) . Новый способ аналогичен предыдущему, но выставляет запреты не по строкам, а по столбцам матрицы. Для каждого клиента все предприятия упорядочиваются по неубыванию производственно-транспортных затрат, выделяется группа *ближайших* предприятий, составляющая d процентов от n , остальные предприятия запрещаются. Пунктирная кривая идет значительно ниже предыдущей и имеет максимум при меньшей плотности $d = 3\%$. Столь сильное различие имеет простое объяснение. Первый способ порождает предприятия с равными по мощности потенциальными областями обслуживания. Предприятия находятся в равных условиях и выбрать из них оптимальный набор достаточно сложно. Второй способ позволяет предприятиям иметь разные мощности областей обслуживания. При малых d предприятия с большими областями получают преимущества, что значительно упрощает задачу. Наконец, последняя кривая соответствует первому способу порождению запретов, но с выбором начальных затрат f_i из интервала $[500, 1000]$. Эта кривая лежит ниже предыдущих. Ее максимум достигается при $d = 3\%$. Разброс в начальных затратах заметно облегчает процесс решения задачи методом ветвей

и границ. При $d = 5\%$ разрыв двойственности $\delta = 100\%(F^* - F_{LP}^*)/F^*$, где F^* — оптимум, F_{LP}^* — оптимум линейной релаксации, составляет в среднем $\delta = 0,7\%$ против $\delta = 4,5\%$ для $b = c = 1000$. Таким образом, трудные примеры ПЗРП возникают только при малых плотностях и одинаковых начальных затратах.

3. Поведение методов локального поиска

Высокая плотность матрицы (g_{ij}) соответствует простым примерам как с точки зрения точных методов типа ветвей и границ, так и с точки зрения методов локального поиска. Интересно понять, что меняется в поведении алгоритмов при переходе в зону критических значений плотности. На рисунках 2а,б,в,г показаны парные замены, выполняемые методом поиска с запретами [2] за 5000 итераций. Они изображены линиями. Крестиками помечены выбранные на плоскости случайные точки. Каждая из них соответствует одновременно некоторому клиенту и месту возможного открытия предприятия. Большие крестики соответствуют оптимальному решению задачи. Кружками помечено начальное решение, выбранное рандомизированным жадным алгоритмом.

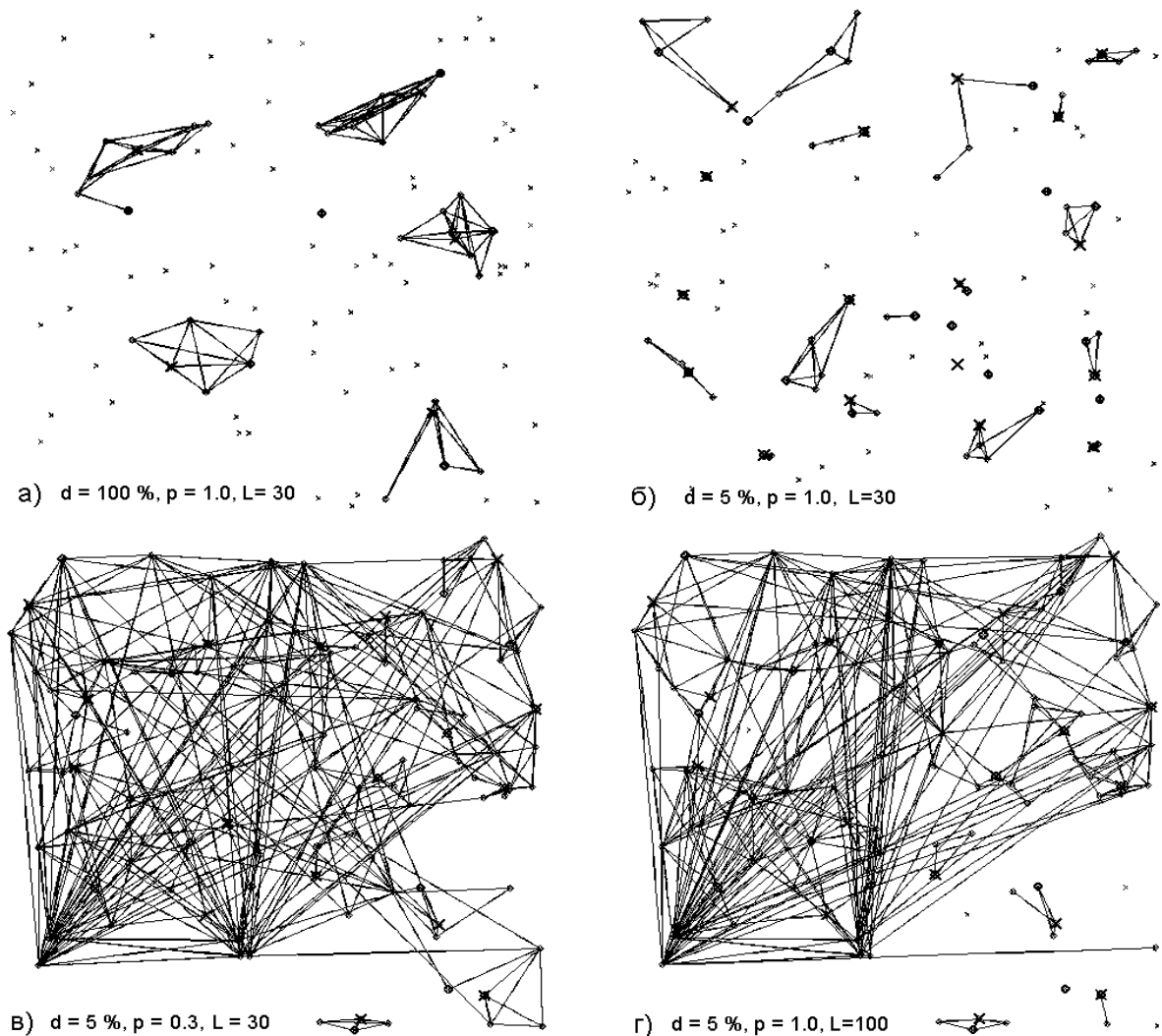


Рис. 2: Поведение алгоритма поиска с запретами

Малое число парных замен на рис. 2а свидетельствует о том, что при $d = 100\%$ алгоритм быстро нашел *хорошее* решение и фактически от него не уходит. При высокой плотности это решение как правило оказывается оптимальным. На рисунке 2б показано поведение алгоритма при $d = 5\%$. Оптимальное решение содержит теперь 18 предприятий против 5-ти в предыдущем случае. Алгоритм по-прежнему детально исследует район найденного им *хорошего* решения, но теперь это решение является только локальным оптимумом. Алгоритм не циклит, но рост числа итераций не приводит к снижению погрешности. Для повышения эффективности нужны специальные приемы, способные внести необходимое разнообразие. Рисунки 2в и 2г показывают влияние рандомизации окрестности и увеличения списка запретов на поведение алгоритма. Величина p задает вероятность включения соседних решений в рандомизированную окрестность. Параметр L задает длину списка запрещаемых парных замен [2]. Оба приема повышают эффективность поиска: растет число просмотренных локальных оптимумов, повышается вероятность найти оптимальное решение задачи. Таким образом, при решении практических задач следует особое внимание обращать на разреженные матрицы и тщательно исследовать области, близкие к критическим значениям плотности.

Список литературы

- [1] В.Л. Береснев, Э.Х. Гимади, В.Т. Дементьев *Экстремальные задачи стандартизации*. Новосибирск: Наука, 1978.
- [2] Е.Н. Гончаров, Ю.А. Кочетов *Вероятностный поиск с запретами для дискретных задач безусловной оптимизации* - Дискретный анализ и исследование операций, Серия 2, 2002. т. 9, N 2, с. 13-30.
- [3] F. Barahona, F. Chudak *Solving large scale uncapacitated facility location problems* in: Approximation and complexity in numerical optimization. 2000, Kluwer Academic Publishers.
- [4] P. Hansen, J. Brimberg, D. Urošević, N. Mladenović *Primal-dual variable neighborhood search for bounded heuristic and exact solution of the simple plant location problem* - Les Cahiers du GERAD, G-2003-64, 2003.
- [5] Yu. Kochetov, D. Ivanenko *Computationally difficult instances for the uncapacitated facility location problem* in: Metaheuristics: Progress as Real Problem Solvers. Springer. 2005. p. 349–365.
- [6] D. Mitchell, B. Selman, H.J. Levesque *Generating hard satisfiability problems* - Artificial Intelligence, 1996, v. 81, p. 17–29.
- [7] M. Sviridenko *An improved approximation algorithm for the metric uncapacitated facility location problem* - Integer Programming and Combinatorial Optimization - IPCO 2002. Springer. 2002. p. 240-257 (Lecture Notes in Comput. Sci.; v. 2337).

CRUCIAL PARAMETERS FOR THE UNCAPACITATED FACILITY LOCATION PROBLEM

E. Alekseeva, Yu. Kochetov, N. Kochetova

Sobolev Institute of Mathematics, Novosibirsk

e-mail: ekaterina2@math.nsc.ru, jkochet@math.nsc.ru, nkochet@math.nsc.ru

Abstract. For the uncapacitated facility location problem we analyse the computational efforts for exact solution with different values of density of the production and transportation matrices and the scattering of the fixed costs for opening facilities. It is shown that the most difficult instances can be generated by small density and identical fixed costs.

Key words: uncapacitated facility location problem, branch-and-bound method, local search, duality gap.

МОДЕЛИРОВАНИЕ ВЗАИМООТНОШЕНИЙ НА ГРАФАХ ¹

С.Н. Астраков

Кемеровский институт (филиал) РГТЭУ, Кемерово
e-mail: astra@kemcity.ru

А.И. Ерзин

Институт математики СО РАН, Новосибирск
e-mail: adil@math.nsc.ru

Аннотация. В работе рассматривается модель, в которой каждый элемент (участник) некоторой системы оценивает отношения со своими соседями и независимо от других элементов принимает решение по изменению значений своих параметров. Состояние системы зависит от значений всех параметров элементов. Начальное состояние системы предполагается известным, затем состояния изменяются по заданному правилу (стратегии) в течение дискретного времени.

Показано существование предельных состояний системы и найдено их аналитическое представление. Приведены примеры задач прикладного характера.

Ключевые слова: модель, развивающаяся система, оптимизация, предельные и равновесные состояния.

1. Постановка задачи

Пусть в полном графе $G = (V, E)$, $|V| = n$, который представляет некоторую систему, отношения вершин (элементов системы) i и j определяются переменными: x_{ij} – степенью воздействия элемента i на элемент j и x_{ji} – степенью воздействия элемента j на элемент i . Будем считать, что набор $X = (x_{ij})$, $i, j \in V$, где x_{ij} можно назвать весом дуги $(i, j) \in E$, определяет состояние системы.

Предположим, что состояния X^k , $k = 0, 1, \dots$, системы меняются дискретно и начальное распределение весов X^0 известно. Будем считать также, что каждая вершина $i \in V$ имеет потенциал $q_i \geq 0$, который она на каждом шаге $k = 1, 2, \dots$ распределяет по инцидентным ребрам,

$$q_i = \sum_{j=1}^n x_{ij}^k.$$

С другой стороны, каждая вершина $i \in V$ в произвольный момент времени $k = 1, 2, \dots$ испытывает воздействие со стороны смежных вершин величины

$$p_i^k = \sum_{j=1}^n x_{ji}^k.$$

При рекуррентном изменении состояний $X^{k+1} = \Phi(X^k)$, представляет интерес существование *предельных* (циклических) и *устойчивых* (не меняющихся от шага к шагу)

¹Работа выполнена при поддержке РФФИ (проект 05-01-00953)

состояний (решений) системы. В данной работе рассматривается класс взаимодействий элементов системы (вершин графа), определяемый стратегиями участников, и изучается характер изменения состояний системы (в частности, при $k \rightarrow \infty$).

Качество отношений элементов i и j по оценке элемента i характеризуется функцией полезности $c_{ij}(x_{ij}, x_{ji})$, которая может быть, например, вида:

1) $c_{ij}(x_{ij}, x_{ji}) = x_{ij} + x_{ji}$ (сотрудничество) или

2) $c_{ij}(x_{ij}, x_{ji}) = x_{ij} - x_{ji}$ (противостояние).

Функции вида 1) и 2) были исследованы в работах [1], [2].

В данной работе рассмотрим более общий линейный случай оценки отношений между элементами системы $c_{ij}(x_{ij}, x_{ji}) = ax_{ij} + bx_{ji}$ с произвольными константами a и b .

Если предположить, что для каждого элемента системы i отношения его с соседями одинаково важны, то можно определить стратегию перераспределения ресурса (определенные величины x_{ij}^{k+1}) на шаге $k+1$ (зная x_{ji}^k , $j = 1, \dots, n$) как решение задачи

$$\min_j (ax_{ij}^{k+1} + bx_{ji}^k) \longrightarrow \max_{\{x_{ij}^{k+1}\}}; \quad (1)$$

$$\sum_{j=1}^n x_{ij}^{k+1} = q_i; \quad (2)$$

Одно из решений задачи (1)-(2) получается из равенств

$$ax_{ij}^{k+1} + bx_{ji}^k = \frac{1}{n-1} \left(\sum_{j=1}^n ax_{ij}^k + bx_{ji}^k \right), j = 1, \dots, n,$$

откуда

$$x_{ij}^{k+1} = -\frac{b}{a}x_{ji}^k + \frac{1}{n-1} \left(\sum_{j=1}^n x_{ij}^k + \frac{b}{a}x_{ji}^k \right).$$

Или

$$x_{ij}^{k+1} = -\frac{b}{a}x_{ji}^k + \frac{1}{n-1} \left(q_i + \frac{b}{a}p_i^k \right) = -\frac{b}{a}x_{ji}^k + f_i^k,$$

где

$$f_i^k = \frac{1}{n-1} \left(q_i + \frac{b}{a}p_i^k \right).$$

Вектор $F^k = (f_1^k, \dots, f_n^k)$ назовем *вектором стабилизации*. По его изменению можно определить характер состояний X^k .

2. Основные результаты

Пусть известно начальное распределение (состояние) потенциалов вершин X^0 и заданы рекуррентные преобразования весов

$$x_{ij}^{k+1} = -\frac{b}{a}x_{ji}^k + f_i^k, \quad f_i^k = \frac{1}{n-1} \left(q_i + \frac{b}{a}p_i^k \right), \quad i \neq j; \quad x_{ii}^{k+1} = 0, \quad i, j = 1, \dots, n. \quad (3)$$

Требуется определить свойства и характер изменения состояний X^k при различных значениях параметром a и b , а также найти равновесное (устойчивое) состояние X^* , не меняющееся в результате преобразований (3).

Отметим, что преобразования (3) не меняют потенциалы вершин q_i . Действительно,

$$\sum_{j=1}^n x_{ij}^{k+1} = \sum_{j \neq i} \left[-\frac{b}{a} x_{ji}^k + \frac{1}{n-1} \left(q_i + \frac{b}{a} p_i^k \right) \right] = -\frac{b}{a} p_i^k + \frac{n-1}{n-1} \left(q_i + \frac{b}{a} p_i^k \right) = q_i.$$

Опуская доказательства, приведем основные полученные результаты.

Лемма 1. Для элементов вектора стабилизации имеют место соотношения

$$\sum_{i=1}^n f_i^k = \frac{Q}{n-1} \left(1 + \frac{b}{a} \right), \quad k = 0, 1, \dots,$$

где $Q = \sum_{i=1}^n q_i$ – суммарный потенциал системы.

Лемма 2. Для каждого элемента $i \in V$ имеют место соотношения

$$f_i^{k+1} = -\frac{1}{n-1} \frac{b}{a} f_i^k + c_i, \quad k = 0, 1, \dots,$$

где

$$c_i = \frac{1}{n-1} \left(1 + \frac{b}{a} \right) \left[q_i \left(1 - \frac{b}{a} \right) + \frac{b}{a} \frac{Q}{n-1} \right].$$

Лемма 3. Для любого шага $k = 1, 2, \dots$ выполняются следующие соотношения

$$f_i^k = z^k f_i^0 + c_i (1 + z + z^2 + \dots + z^{k-1}), \quad i = 1, \dots, n,$$

где $z = -\frac{1}{n-1} \frac{b}{a}$. Или, с учетом суммы геометрической прогрессии,

$$f_i^k = \left(-\frac{1}{n-1} \frac{b}{a} \right)^k f_i^0 + c_i \frac{a(n-1)}{b+a(n-1)} \left[1 - \left(-\frac{1}{n-1} \frac{b}{a} \right)^k \right].$$

Теорема 1. Для сходимости вектора стабилизации необходимо и достаточно выполнение условия

$$\left| \frac{b}{a} \right| < n-1 \quad (4)$$

При этом предельные значения координат вектора стабилизации равны

$$f_i^* = \lim_{k \rightarrow \infty} f_i^k = c_i \frac{n-1}{b+a(n-1)} = \frac{a+b}{b+a(n-1)} \left[q_i \left(1 - \frac{b}{a} \right) + \frac{b}{a} \frac{Q}{n-1} \right].$$

Теорема 0. А. При выполнении условия (4) решение (состояние)

$$x_{ij}^* = \frac{1}{b+a(n-1)} \left(a q_i - b q_j + \frac{b Q}{n-1} \right) \quad (5)$$

является стационарным (равновесным), т.е. не изменяется в результате преобразований (3).

В. При выполнении неравенства $|b/a| < 1$ любое начальное решение системы в результате преобразований (3) стремится к состоянию (5) при $k \rightarrow \infty$.

Замечание 1. При некоторых значениях a и b (в частности, при $|a| = |b|$) решение системы сходится к двум предельным циклическим (четному и нечетному), зависящим от начальных условий, среднее арифметическое которых дает устойчивое состояние (5).

Как показывают расчеты, цикличности можно избежать, если в стратегию принятия решений ввести свойство *преемственности*:

$$y_{ij}^{k+1} = -\frac{b}{a}x_{ji}^k + f_i^k; \quad x_{ij}^{k+1} = \alpha y_{ij}^{k+1} + (1 - \alpha)x_{ij}^k, \quad i \neq j, \quad 0 < \alpha < 1; \quad x_{ii}^{k+1} = 0. \quad (6)$$

Замечание 2. В случае $1 < |b/a| < n - 1$ решение системы в общем случае расходится при сходящемся векторе стабилизации.

Для иллюстрации подхода ниже приведены два практических примера.

3. Конкурентные отношения

Предположим три фирмы (или политические партии) распределяют свои ресурсы $q_1 = 80$, $q_2 = 50$, $q_3 = 40$ друг против друга с целью улучшения своего конкурентного положения. Требуется определить устойчивое состояние, устраивающее всех участников.

В данном случае функции $c_{ij}(x_{ij}, x_{ji}) = x_{ij} - x_{ji}$, $i, j = 1, 2, 3$. Согласно Лемме 2, величины $c_i = 0$, $i = 1, 2, 3$, поэтому $f_i^{k+1} = 0.5f_i^k$, $i = 1, 2, 3$.

При начальном распределении ресурсов

$$X^0 = \begin{pmatrix} 0 & 50 & 30 \\ 40 & 0 & 10 \\ 20 & 20 & 0 \end{pmatrix}$$

вектор стабилизации $F^0 = (10, -10, 0)$.

После первого шага получаем

$$X^1 = \begin{pmatrix} 0 & 50 & 30 \\ 40 & 0 & 10 \\ 30 & 10 & 0 \end{pmatrix}, \quad F^1 = (5, -5, 0).$$

При $k \rightarrow \infty$ решение стремится к устойчивому состоянию

$$X^* = \begin{pmatrix} 0 & 45 & 35 \\ 45 & 0 & 5 \\ 35 & 5 & 0 \end{pmatrix}$$

с нулевым вектором стабилизации $F^* = (0, 0, 0)$. Это состояние устраивает всех участников, т.к. $c_{ij} = 0$, $i, j = 1, 2, 3$.

4. Кооперативные отношения

Рассмотрим распределенную коммуникационную сеть, в которой каждая вершина обслуживает только инцидентные ей ребра (линии связи). В примере возьмем четырехвершинную сеть, заданную матрицей смежности

$$\begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix},$$

с ресурсами вершин $q = (50, 40, 30, 20)$. Здесь x_{ij} – величина ресурса, направленного вершиной i на поддержание линии (i, j) . Так как смежные вершин i и j обслуживают линию (i, j) совместно, можно считать $c_{ij} = x_{ij} + x_{ji}$. Каждая вершина старается, чтобы все инцидентные ей ребра обслуживались одинаково хорошо, поэтому выбор новых значений весов происходит также по критерию (1)-(2). В примере рассматривается неполный граф, поэтому следует считать $x_{ij}^k = x_{ji}^k = 0$, $(i, j) \notin E$ и сделать корректировку при вычислении сил стабилизации [1], положив:

$$f_i^k = \frac{q_i + p_i^k}{d_i}, \quad k = 0, 1, \dots, \quad i = 1, \dots, 4,$$

где d_i – степень вершины i .

Расчеты показали, что при начальном состоянии

$$X^0 = \begin{pmatrix} 0 & 20 & 20 & 10 \\ 10 & 0 & 20 & 10 \\ 20 & 10 & 0 & 0 \\ 10 & 10 & 0 & 0 \end{pmatrix}, \quad C^0 = \begin{pmatrix} 0 & 30 & 40 & 20 \\ 30 & 0 & 30 & 20 \\ 40 & 30 & 0 & 0 \\ 20 & 20 & 0 & 0 \end{pmatrix},$$

на шаге $k = 20$ имеем равномерное обслуживание линий связи:

$$X^{20} = \begin{pmatrix} 0 & 20.25 & 11.125 & 18.625 \\ 7.75 & 0 & 14.875 & 17.375 \\ 16.875 & 13.125 & 0 & 0 \\ 19.375 & 10.625 & 0 & 0 \end{pmatrix}, \quad C^{20} = \begin{pmatrix} 0 & 28 & 28 & 28 \\ 28 & 0 & 28 & 28 \\ 28 & 28 & 0 & 0 \\ 28 & 28 & 0 & 0 \end{pmatrix}.$$

Исследования в данном направлении продолжаются как для различных графов, так и для разных функций оценки отношений между элементами систем, которые имеют прикладное значение.

Список литературы

- [1] А.И. Ерзин, С.Н. Астраков, И.И. Тахонов, О.А. Гадяцкая *Одна задача функционирования распределенной сети.* - В кн.: Материалы международного семинара "Вычислительные методы и решения оптимизационных задач". Бишкек, 2004, с. 77-82.

- [2] С.Н. Астраков, А.И. Ерзин *Одна модель саморегулирующейся системы.* - Математические структуры и моделирование, 2004, N13, с. 30-38.

THE MODELLING OF MUTUAL RELATIONS ON GRAPHS

S.N. Astrakov

Kemerovo Institute (Branch) of RSTEU, Kemerovo
e-mail: astra@kemcity.ru

A.I. Erzin

Sobolev Institute of Mathematics, SD of RAS, Novosibirsk
e-mail: adil@math.nsc.ru

Abstract. Considered a model in which each element (competitor) of some system evaluates the relations with its neighbors and independently of other elements makes a decision how to change the values of its parameters. The state of a system depends on the values of all parameters. An initial state of a system is supposed to be known, then the states are change according to a given rule (strategy) during a discrete time.

The existence of the limiting states of a system is shown and their analytical presentations are found. The examples of applied problems are presented as well.

Key words: model, evolutionary system, optimization, limiting and equilibrium states

АЛГОРИТМЫ С ОЦЕНКАМИ ДЛЯ НЕКОТОРЫХ ОБОБЩЕНИЙ ЗАДАЧИ КОММИВОЯЖЕРА ¹

А.Е. Бабурин, Э.Х. Гимади

Институт Математики им. С.Л. Соболева СО РАН, Новосибирск
e-mail: gimadi@math.nsc.ru

Аннотация. В статье приводятся результаты исследований авторов по разработке приближенных алгоритмов с оценками для решения некоторых задач, связанных с отысканием в полном взвешенном графе подграфов специального вида. Рассмотрены задача коммивояжера на максимум в евклидовом пространстве, задача отыскания двух реберно непересекающихся гамильтоновых циклов экстремального суммарного веса (на минимум или на максимум), задача отыскания связного подграфа максимального веса с заданными степенями вершин.

Ключевые слова: Задача коммивояжера, граф, приближенный алгоритм, оценка точности, асимптотически точный, трудоемкость.

Введение

Результаты, кратко представленные в данной работе, относятся к построению приближенных полиномиальных алгоритмов решения задач, которые являются определенными модификациями классической задачи коммивояжера, либо ее обобщениями, связанными с отысканием в полном взвешенном графе подграфов специального вида.

Задача коммивояжера MAX SNP-трудна, т. е. из существования полиномиальной аппроксимационной схемы для ее решения следует $P = NP$. Это обстоятельство стимулировало исследования приближенных полиномиальных алгоритмов с оценками точности их решения.

Наиболее полная информация о современных достижениях по задаче коммивояжера до 2002 года содержится в книге [11]. Представленные ниже результаты получены в основном в последнее время.

1. Евклидова задача коммивояжера на максимум (ETSPmax)

ETSPmax в R^k NP-трудна при $k \geq 3$ [11]. Вопрос о сложностном статусе ETSPmax при $k = 2$ остается открытым.

В [5] для решения ETSPmax в R^k представлен асимптотически точный алгоритм с оценкой точности $1 - 3 n^{-2/(k+2)}$. Здесь параметр t алгоритма — натуральное число, $2 \leq t \leq n/4$.

Ниже приведены краткие описания алгоритмов из работ [3] и [6], представляющих собой более упрощенные версии по сравнению с [5]. Временная сложность алгоритмов, равная $O(n^{5/2})$, определяется трудоемкостью отыскания паросочетания максимального веса [9].

¹Исследование выполнено при поддержке Российского фонда фундаментальных исследований (проект 05-01-00395а), программы поддержки ведущих научных школ РФ (проект НШ-313.2003.1) и INTAS (проект 04-77-7173)

1.1. Алгоритм $\tilde{A}(t)$ [3]

Находим паросочетание M максимального веса. Разобьем M на подмножество из $t - 1$ "легких" ребер и подмножество из оставшихся "тяжелых" ребер, $2 \leq t \leq n/4$.

Назовем *набором* либо ребро, либо любое множество ребер, образующих цикл в графе G . Выделим в нем *ведущее* и *ведомое* ребра. (Если набор состоит из одного ребра, то оно одновременно является ведущим и ведомым).

С множеством одноэлементных наборов из тяжелых ребер повторяем следующую процедуру \mathcal{P} , пока число наборов не станет равным $t - 1$.

Процедура \mathcal{P} :

P_1 : Выберем два набора с минимальным углом между ведущими ребрами;
 P_2 : Объединим выбранные наборы в один, заменив ведущие ребра на пару соединяющих их ребер максимального суммарного веса, и в качестве ведущего ребра нового набора объявляется одно из его ведомых ребер.

Множество наборов, полученных в результате работы процедуры \mathcal{P} , перемежаем одноэлементными наборами, состоящими из оставшихся $t - 1$ легких ребер. Образованную таким образом последовательность наборов склеиваем в один набор-цикл, используя $t - 1$ раз операцию P_2 процедуры \mathcal{P} .

В случае нечетного n вершину x_0 , не попавшую в паросочетание M , добавляем в построенный цикл заменой одного из его произвольных ребер, скажем (x, y) , на пару новых ребер $(x, x_0), (x_0, y)$.

В результате работы алгоритма $\tilde{A}(t)$ построен гамильтонов цикл, дающий приближенное решение задачи ETSPmax в R^k .

Наилучшее приближение при фиксированном n можно достичь определенным выбором параметра t . Более того, при $t^* = \lceil n^{(k-1)/(k+1)} \rceil$, алгоритм $\tilde{A}(t^*)$ гарантирует получение асимптотически точного решения задачи (когда относительная погрешность получаемого решения стремится к нулю при $n \rightarrow \infty$).

1.2. Модификация алгоритма без деления ребер на легкие и тяжелые [6]

Одним из основных моментов в алгоритмах [5], [3] является операция разбиения максимального паросочетания на множество легких и тяжелых ребер. Процесс соединения пары ребер парой соединяющих (процедура \mathcal{P}) производится только над подмножеством тяжелых ребер M . Невозможность распространить этот процесс на все ребра паросочетания обусловлена тем, что при уменьшении количества склеиваемых ребер достигаются все меньшие и меньшие оценки суммарного веса получаемых соединяющих ребер, что в итоге приводит к ухудшению эффективности алгоритма.

Оказывается, что при наложении некоторых достаточно слабых ограничений на исходный граф, данный процесс допускает продолжение на все ребра паросочетания M . Эта идея была реализована в работе [6] для отыскания приближенного решения ETSPmax на графах, у которых максимальное паросочетание характеризуется ограниченным отношением весов максимального и минимального ребер. К этому классу относится, например, ETSPmax на графе, имеющем ограниченный диаметр и вершины в узлах целочисленной решетки. Получены условия на длины ребер, при которых алгоритм является асимптотически точным и имеет лучшую скорость сходимости в асимптотике по сравнению с алгоритмами в [5] и [3].

2. Задачи нахождения двух непересекающихся гамильтоновых циклов

Дан полный n -вершинный неориентированный взвешенный граф $G = (V, E)$ с весовыми функциями $w_1 : E \rightarrow R$, $w_2 : E \rightarrow R$. Далее под $W(H)$ будем понимать $W(H) = \sum_{e \in H} w(e)$. Требуется найти такие реберно непересекающиеся гамильтоновы циклы $H_1, H_2 \subset G$, при которых их суммарный вес $W_1(H_1) + W_2(H_2)$ достигает экстремума.

В [2] показано, что задача отыскания двух реберно непересекающихся гамильтоновых циклов в полном взвешенном графе NP-трудна в сильном смысле, как в случае двух, так и в случае одной весовой функции.

2.1. Метрическая задача на минимум [2]

В метрической задаче для весовых функций выполняется неравенство треугольника. Рассматриваемая задача возникает, например, при планировании одновременной работы двух роботов, обрабатывающих n деталей на плате. Каждый из роботов совершает замкнутый обход всех деталей. В интересах технологической безопасности прохождение роботов по одному и тому же ребру запрещается. При этом требуется минимизировать суммарное время, затрачиваемое роботами на обработку деталей.

При построении приближенного решения задачи в [2] используется классический алгоритм КС (Кристофидеса - Сердюкова), который находит решение метрической задачи коммивояжера с точностью $3/2$. Время работы алгоритмов в следующих двух подразделах определено трудоемкостью отыскания паросочетания минимального веса и равно $O(n^{5/2})$, как и в разделе 1.

2.1.1. Алгоритм для задачи с одной весовой функцией

С помощью алгоритма КС строится первый гамильтонов цикл $H_1 \subset G$. Без ограничения общности считаем, что вершины графа занумерованы так, что $H_1 = \{1, 2, \dots, n, 1\}$. Построение второго гамильтонова цикла H_2 , непересекающегося с первым, проводится по-разному, в зависимости от четности n .

В случае нечетного n в качестве второго гамильтонова цикла H_2 берется последовательность вершин $\{1, 3, 5, \dots, n, 2, 4, \dots, n-1, 1\}$, в которой в возрастающем порядке сначала расположены все нечетные вершины, затем все четные вершины; на этом алгоритм заканчивает свою работу.

В случае четного n второй гамильтонов цикл H_2 строится посредством соответствующего склеивания пары циклов $\{1, 3, \dots, n-1, 1\}$ и $\{2, 4, \dots, n-2, n, 2\}$, составляющих 2-фактор в графе G .

В результате пара полученных непересекающихся гамильтоновых циклов H_1, H_2 дает решение задачи с точностью, асимптотически (с ростом n) равной $9/4$ [2].

2.1.2. Алгоритм для задачи с двумя весовыми функциями

С помощью алгоритма КС строим два гамильтоновых цикла H_1 и H_2 в графе G с весовыми функциями w_1 и w_2 , соответственно. Если найденные гамильтоновы циклы H_1 и H_2 не пересекаются по ребрам, то мы получили приближенное решение с оценкой точности $3/2$ и алгоритм на этом заканчивает свою работу.

В противном случае один из гамильтоновых циклов, скажем H_1 , оставляем без изменения, а другой перестраиваем в гамильтонов цикл \tilde{H}_2 так, чтобы его ребра не пересекались с циклом H_1 . (Детали и корректность перестройки см. в [2].) Поменяв ролями циклы H_1 и H_2 , получаем другое решение: (\tilde{H}_1, H_2) . Приближенное решение составляется из пары непересекающихся гамильтоновых циклов наименьшего веса. Алгоритм имеет гарантированную точность $12/5$ [2].

2.2. Задача на максимум с одной весовой функцией [1]

Дан полный n -вершинный неориентированный граф $G = (V, E)$ с весовой функцией на ребрах $w : E \rightarrow R$. Требуется найти такие реберно непересекающиеся гамильтоновы циклы $H_1, H_2 \subset G$, суммарный вес которых $W(H_1) + W(H_2)$ максимален.

Отправная идея построения алгоритма [1] навеяна работой [4], в которой ребра двух максимальных по весу подграфов — паросочетания и два-фактора — распределяются по двум частичным турам, дополняемым далее до гамильтоновых циклов посредством оставшихся ребер. За решение принимается тот гамильтонов цикл, вес которого больше. Большой по весу гамильтонов цикл дает решение с константной оценкой точности, равной $3/4$. Однако прямое применение алгоритма [4] к задаче с двумя непересекающимися гамильтоновыми циклами оказывается неприемлемым, поскольку частичные туры, формируемые алгоритмом [4], могут содержать пересекающиеся ребра.

Предлагаемый алгоритм корректен для $n > 13$ (иначе задача решается любым точным переборным алгоритмом).

Сначала в графе G строится кубический (при четном n) или почти кубический (при нечетном n) подграф $K \subset G$ максимального реберного веса. (Подграф почти кубический, если в нем содержится только одна вершина степени 4, а остальные имеют степень 3).

Подграф K разбиваем на частичный тур и 2-сочетание. Частичный тур и 2-сочетание модифицируются путем переброски части ребер из одного подграфа в другой. В итоге, становится возможным, добавляя ребра, достроить эти частичные туры до гамильтоновых циклов, не пересекающихся по ребрам. Ребра графа G' , образованного этими циклами, имеют суммарный вес, не меньший $(3/4)W^*$, где W^* — оптимум задачи.

Оценка точности $3/4$ следует из того, что вес графа G' не меньше $(3/4)W^*$, и в ходе выполнения алгоритма этот вес полностью распределяется между двумя реберно непересекающимися гамильтоновыми циклами. Временная сложность алгоритма, определяемая трудоемкостью отыскания 2-фактора максимального веса, равна $O(n^3)$ (см., например, [8]).

3. Задача нахождения в полном графе связного подграфа с максимальным суммарным весом ребер и заданными степенями вершин

Задан $G(V, E)$ — полный неориентированный граф без петель с n вершинами. Определена функция $w : E \rightarrow R^+$ и натуральные числа d_i ($i = 1, \dots, n$), $1 < d_i < n$. Задача состоит в отыскании в графе связного подграфа максимального реберного веса с заданными степенями вершин. В частном случае, когда степени всех вершин искомого подграфа равны 2, мы имеем задачу коммивояжера на максимум. Задача изучалась в [10], где на случай четных степеней d_i был предложен приближенный алгоритм решения метрической задачи. В [7] предложены новые алгоритмы, работа которых корректна независимо от четности степеней вершин выбираемого подграфа и применима не только

для метрической задачи.

3.1. Алгоритм A для задачи с детерминированными входными данными.

С использованием алгоритма из [8] находится подграф графа G с заданными степенями вершин, с максимальным суммарным весом ребер. Выделяются компоненты связности этого подграфа. В каждой компоненте специальным образом выбирается одно ребро. Выбранные ребра удаляются, а компоненты связности последовательно соединяются новыми ребрами, компенсирующими степени вершин до требуемых значений.

Утверждение. Алгоритм A находит решения задачи за время $O(n^3)$ с оценкой относительной погрешности, не большей $2/(d^2 + d)$, где $d = \min\{d_i | i = 1, \dots, n\}$. В метрическом случае погрешность в два раза меньше.

Замечание. При $\min\{d_i | i = 1, \dots, n\} + \max\{d_i | i = 1, \dots, n\} > n - 2$ алгоритм A строит точное решение задачи.

3.2. Алгоритм \tilde{A} для задачи со случайными входными данными.

Для задачи отыскания d -регулярного подграфа в полном неориентированном графе со случайными входными данными построен новый алгоритм с квадратичной временной сложностью. Алгоритм основан на блочном представлении матрицы расстояний. В некоторых блоках жадным алгоритмом решается обобщенная задача о назначении на максимум, в других — задача коммивояжера на максимум (например, алгоритмом "Иди в самый удаленный город"). В результате, выбранные элементы всех блоков вместе представляют допустимое решение исходной задачи.

Изучен случай, когда веса ребер задаются случайно и независимо друг от друга из некоторого сегмента. При распределении минорирующего типа (частным случаем которого является равномерное распределение) алгоритм \tilde{A} за время $O(n^2)$ почти всегда дает решение с относительной погрешностью в пределах $O(\ln(n/d)/(n/d))$. Т.е. при $d = o(n)$ алгоритм асимптотически точен.

Список литературы

- [1] А.А. Агеев, А.Е. Бабурин, Э.Х. Гимади, Н.М. Коркишко. Алгоритмы с константными оценками точности для отыскания двух реберно непересекающихся гамильтоновых циклов экстремального веса – Материалы Всероссийской конференции "Проблемы оптимизации и экономические приложения". 1-5 июля 2003 г., Омск. С. 9-12.
- [2] А.Е. Бабурин, Э.Х. Гимади, Н.М. Коркишко. Приближенные алгоритмы для нахождения двух реберно непересекающихся гамильтоновых циклов минимального веса – Дискрет. анализ и исслед. операций. Сер 2, Том. 11, N 1. Новосибирск, 2004, С. 11-25.
- [3] Э.Х. Гимади. Новая версия асимптотически точного алгоритма решения евклидовой задачи коммивояжера – Труды Байкальской конференции по математическому программированию. Иркутск. 2002.
- [4] А.И. Сердюков. Алгоритм с оценкой для задачи коммивояжера на максимум – Управляемые системы. Сб. науч. тр. Вып. 25. Новосибирск: Ин-т математики СО АН СССР. 1984. С. 80–86.

- [5] А.И. Сердюков. *Асимптотически точный алгоритм для задачи коммивояжера на максимум в евклидовом пространстве* – Управляемые системы. Сб. науч. тр. Вып. 27. Новосибирск: Ин-т математики СО АН СССР. 1987. С. 79–87.
- [6] А.Е. Бабурин, Э.Х. Гимади. *Об асимптотической точности одного алгоритма решения задачи коммивояжера на максимум в евклидовом пространстве* – Дискретный анализ и исследование операций. Серия 1, Новосибирск, 2002. Том 9, No 4, С. 23–32.
- [7] А.Е. Baburin, E.Kh. Gimadi. *Algorithms for finding a maximum-weight spanning connected subgraph with given vertex degrees* – Operations Research Proceedings 2004, Springer Verlag (2005).
- [8] H.N. Gabow. *An efficient reduction technique for degree-constrained subgraph and bidirected network flow problems* – Proceedings of the 15th Annual ACM Symposium on Theory of Computing, 1983, P. 448–456.
- [9] S. Micali, V.V. Vazirani. *An $O(\sqrt{|V|} \cdot |E|)$ algorithm for finding maximal matching in general graphs* – Proceedings of the 21st Annual IEEE Symposium on Foundations of Computer Science. IEEE, New York, P. 17–27.
- [10] E. Kh. Gimadi, A.I. Serdyukov. *A problem of finding the maximal spanning connected subgraph with given vertex degrees* – Oper. Res. Proc. 2000, Springer Verlag (2001), 55–59.
- [11] *The Traveling Salesman Problem and its variations* (ed. by A. Punnen and G. Gutin). Kluwer Academic Publishers. Dordrecht/Boston/London. 2002.

ALGORITHMS WITH PERFORMANCE RATIOS FOR SOME GENERALIZATIONS OF THE TRAVELING SALESMAN PROBLEM

Alexei Y. Baburin, Edward Kh. Gimadi

*Sobolev Institute of Mathematics SB RAS, Novosibirsk
alebab@mail.ru; gimadi@math.nsc.ru*

Abstract. We suggest some results in designing approximation algorithms with performance ratios for solving problems that concern finding special subgraphs in complete weighted graph. The problems considered are the Traveling Salesman Problem in Euclidean space, the problem of finding two edge-disjoint Hamiltonian circuits of extremal total weight (on minimum or on maximum), the problem of finding a connected degree-constrained subgraph of maximum weight.

Key words: Traveling Salesman Problem, graph, approximation algorithm, performance ratio, asymptotically optimal, time complexity.

АЛГОРИТМ МУРАВЬИНОЙ КОЛОНИИ ДЛЯ ЗАДАЧ ДВУХМЕРНОЙ УПАКОВКИ: РЕЗУЛЬТАТЫ ВЫЧИСЛИТЕЛЬНОГО ЭКСПЕРИМЕНТА¹

А.Ф. Валеева, М.Н. Аглиуллин

Уфимский Государственный Авиационный Технический Университет, Уфа
e-mail: griph@mail.ru

Аннотация. В статье рассматривается метод муравьиной колонии для решения задач упаковки заданного набора прямоугольников в прямоугольные листы или полубесконечную полосу. Приводятся результаты вычислительного эксперимента на тестовых задачах из библиотеки OR-library.

Ключевые слова: прямоугольная упаковка, метаэвристика, алгоритм муравьиной колонии.

Введение

Во многих приложениях возникает задача упаковки конечного набора прямоугольников в прямоугольные листы одинакового размера (2DBPP) или полубесконечную полосу фиксированной ширины (1.5BPP) таким образом, чтобы упакованные прямоугольники не пересекались и количество использованных листов, либо длина занятой части полосы были минимальны. Эта задача NP-трудна, так как её частный случай — задача одномерной упаковки — является NP-трудной. В литературе рассматривается несколько различных вариантов 2DBPP и 1.5DBPP, отличающихся допустимостью поворота прямоугольников на 90° и требованием гильотинности резов [1]. В данной работе представлен алгоритм на базе метаэвристики муравьиной колонии для решения задач 2DBPP и 1.5DBPP с разрешенными поворотами прямоугольников на 90° и отсутствием требования гильотинности.

1. Постановка задачи

Задача упаковки конечного набора прямоугольников в полубесконечную полосу заданной ширины (1.5DBPP). Имеется прямоугольная полоса заданной ширины W и неограниченной длины, а также набор прямоугольных предметов заданных размеров $w_i, l_i, i = 1, \dots, n$. (w_i — ширина i -го предмета, l_i — длина). Найти упаковку предметов в полосу, занимающую минимальную длину при условиях:

1. ребра упакованных предметов параллельны ребрам полосы;
2. упакованные предметы не перекрывают друг друга;
3. упакованные предметы не выходят за границы полосы.

Задача упаковки конечного набора прямоугольников в листы заданного размера (2DBPP). Имеется неограниченное количество прямоугольных листов заданной ширины W и длины L , а также набор прямоугольных предметов заданных размеров $w_i, l_i, i = 1, \dots, n$. Найти упаковку предметов с минимальным расходом листов при условиях:

¹Работа поддержана Российским фондом фундаментальных исследований, грант № 01-01-00510

1. ребра упакованных предметов параллельны ребрам листа;
2. упакованные предметы не перекрывают друг друга;
3. упакованные предметы не выходят за границы листа.

2. Метод решения

Для рассматриваемых задач предлагается эвристический метод (Ant Colony Packing, ACP), основанный на метаэвристике муравьиной колонии (Ant Colony, AC) [2]. В табл. 1 приведены основные характеристики элементов метаэвристики AC, в табл. 2 даны основные процедуры, а на рис. 1 показана общая схема AC.

Элемент AC	Описание элемента AC
Фрагмент	Составная часть решения задачи
Феромон	Численная характеристика фрагмента i или перехода между фрагментами (i, j) (обозначается τ_i либо τ_{ij} соответственно). Показывает, насколько часто данный фрагмент (переход) входил в лучшие решения на предыдущих итерациях алгоритма.
Локальная информация	Численная характеристика фрагмента i или перехода между фрагментами (i, j) (обозначается η_i либо η_{ij} соответственно). Показывает полезность фрагмента для построения хорошего решения.
Агент	Алгоритм, который итеративно строит из множества фрагментов допустимое решение задачи

Табл. 1. Характеристики элементов AC

Процедура 1	Процедура построения начального частичного решения задачи
Процедура 2	Процедура, находящая множество I допустимых фрагментов, т.е. фрагментов, удовлетворяющих условиям задачи
Процедура 3	Процедура, выбирающая из множества I один фрагмент, который будет добавлен к решению
Процедура 4	Процедура, добавляющая фрагмент к решению
Процедура 5	Процедура, улучшающая полученное агентом решение с использованием алгоритма локального поиска
Процедура 6	Процедура глобального обновления феромона
Процедура 7	Процедура обновления феромона на шаге
Процедура 8	Процедура декодирования, преобразующая последовательность фрагментов в решение задачи оптимизации

Табл. 2. Список процедур, используемых в алгоритме AC

Инициализация параметров алгоритма, феромона Пока не выполнены условия останова Построение решений Применить к полученным решениям процедуру 5 (не обязательная) Если среди полученных решений есть рекордные, применить процедуру 8 Выполнить процедуру 6

Оптимизационный цикл

Выполнить процедуру 1 Для всех агентов, не завершивших построение решения $I \leftarrow$ Построить множество допустимых фрагментов по процедуре 2 $s \leftarrow$ Выбрать фрагмент из I согласно процедуре 3 Добавить s в решение при помощи процедуры 4 Выполнить процедуру 7
--

Построение решений

Рис. 1. Общая схема метаэвристики АС

Приведем интерпретацию элементов метаэвристики АС для задач 2DBPP и 1.5DBPP. Так, фрагментом в алгоритме АСР является заданный прямоугольник с шириной w и длиной l , а феромон наносится на переходы, т.е. последовательности из двух прямоугольников (i, j) , $i, j = 1, \dots, n$ (n — заданное количество прямоугольников). Пусть i — последний фрагмент, добавленный некоторым агентом в свое решение, а j — один из фрагментов-кандидатов на добавление в решение, тогда локальная информация перехода (i, j) равна площади фрагмента j и определяется как: $\eta_j = w_j \cdot l_j$.

Такой выбор локальной информации способствует тому, что в первую очередь будут упаковываться прямоугольники больших размеров.

В отличие от принятых в алгоритмах АСО правилах, в алгоритме АСР в процедурах 6 и 7 применяется комплексная стратегия изменения феромона, являющаяся синтезом известных стратегий ACS и MMAS [2]. В MMAS введено два параметра τ_{max} , τ_{min} — максимальное и минимальное значения феромона соответственно, ограничивающие возможные значения феромона интервалом $[\tau_{min}, \tau_{max}]$. При этом начальное значение феромона τ_{init} устанавливается равным максимальному значению (т.е. $\tau_{init} \leftarrow \tau_{max}$). В процедуре 6 вводится параметр p_{gb} , контролирующий нанесение феромона либо на фрагменты рекордных решений, либо на фрагменты лучших решений итерации, $p_{gb} \in [0, 1]$. Комплексная стратегия изменения феромона для алгоритма АСР представлена на рис. 2.

Пусть X — равномерно распределенная на интервале $(0, 1)$ случайная величина. Если наступит событие $\{(X = x) \leq p_{gb}\}$, то уровень феромона на переходах рекордных решений изменяется по формуле: $\tau_{ij}(t+1) \leftarrow (1 - \rho) \cdot \tau_{ij}(t) + \Delta\tau^1$, где $\Delta\tau^1 = (2 - CC_r)^{-1}$; CC_r — коэффициент раскрытия для рекордного решения; ρ — коэффициент, задаваемый при инициализации алгоритма, $\rho \in (0, 1)$ иначе новый уровень феромона на переходах лучших решений итерации определяется по формуле: $\tau_{ij}(t+1) \leftarrow (1 - \rho) \cdot \tau_{ij}(t) + \Delta\tau^2$, где $\Delta\tau^2 = (2 - CC_i)^{-1}$; CC_i — коэффициент раскрытия для лучшего решения на данной итерации; t — номер текущей итерации; Значение $\tau_{ij}(t+1)$ ограничивается диапазоном $[\tau_{min}, \tau_{max}]$.

Процедура 6. Глобальное испарение феромона с применением стратегии MMAS

При добавлении фрагмента в решение, уровень феромона на переходах понижается по формуле $\tau_{ij}(t+1) \leftarrow (1 - \xi) \cdot \tau_{ij}(t) + \xi \cdot \tau_0$, где $\tau_{ij}(t)$ — уровень феромона на переходе (i, j) на итерации t ; ξ — коэффициент, задаваемый при инициализации алгоритма, $\xi \in (0, 1)$. Значение $\tau_{ij}(t+1)$ ограничивается диапазоном $[\tau_{min}, \tau_{max}]$.

Процедура 7. Локальное испарение феромона по стратегии ACS

Рис. 2. Комплексная стратегия обновления феромона

3. Численный эксперимент

Были проведены численные эксперименты на тестовых примерах из библиотеки OR-library (задачи S.P. Fekete и J. Schepers [3]), расположенной в Интернет по адресу: <http://mscmga.ms.ic.ac.uk/jeb/orlib/ngcutinfo.html>.

Решалась задача 2DBPP. Длина и ширина листа во всех задачах равнялась 100. Из каждого класса задач с размерностями $n = 20, 50, 100, 150, 250, 500$ были решены по 10 задач. Использовались следующие значения параметров АСР: $p_{gb} = 0.5, q_0 = 0.5, \tau_{min} = 0.1, \tau_{max} = 10, \tau_{init} = 0.1$. В табл. 3 и 4 приведены результаты эксперимента для $n = 40, n = 500$.

Тестовые наборы данных	n	Лучший CC	Средний CC	Худший CC	Среднее время, с
Среднее значение задач I типа (ngcutfs1_11–ngcutfs1_20)	40	0,8912	0,885635	0,879966	3
Среднее значение задач II типа (ngcutfs2_11–ngcutfs2_20)	40	0,889265	0,882416	0,875481	3
Среднее значение задач III типа (ngcutfs3_11–ngcutfs3_20)	40	0,912253	0,906232	0,902178	3

Табл. 3. Результаты эксперимента на задачах S.P. Fekete и J. Schepers ($n = 40$)

²Для задачи 1.5DBPP $CC = \frac{\sum_{i=1}^n w_{il_i}}{W}$, а для задачи 2DBPP $CC = \frac{\sum_{i=1}^n w_{il_i}}{W \cdot L}$

Тестовые наборы данных	n	Лучший CC	Средний CC	Худший CC	Среднее время, с
Среднее значение задач I типа (ngcutfs1_151–ngcutfs1_160)	500	0,921929	0,919619	0,917309	633
Среднее значение задач II типа (ngcutfs2_151–ngcutfs2_160)	500	0,925380	0,921201	0,917021	705
Среднее значение задач III типа (ngcutfs3_151–ngcutfs3_160)	500	0,935854	0,934267	0,932679	704,1

Табл. 4. Результаты эксперимента на задачах S.P. Fekete и J. Schepers ($n = 500$)

Как показал численный эксперимент, алгоритм АСР дает лучшие результаты по мере роста размерности решаемых задач (для $n = 100, 150, 250, 500$).

Список литературы

- [1] A. Lodi, S. Martello, M. Monaci *Two-dimensional packing problems: A survey*. — EJOR, 2002, 141, P. 241–252
- [2] M. Dorigo, G. Di Caro, L.M. Gambardella *Ant Algorithms for Discrete Optimization*. — Artificial Life, 1999, Vol. 5, No.3, P. 137–172
- [3] S.P. Fekete, J. Schepers *A new exact algorithm for general orthogonal d-dimensional knapsack problems*. — В кн.: Algorithms-ESA'97, Springer Lecture Notes in Computer Science, Berlin, 1997 (rev. 2000)

ANT COLONY ALGORITHM FOR 2D BIN PACKING PROBLEMS: NUMERICAL RESULTS

A.F. Valeyeva, M.N. Agliullin

Ufa State Aviation Technical University, Ufa
e-mail: griph@mail.ru

Abstract. Ant Colony algorithm for packing a given set of rectangles into rectangular bins or strip of given width is devised. Numerical results are reported for some tasks in the OR-library.

Key words: ractanglular packing, ACO

НЕРЕГУЛЯРНОЕ РАЗМЕЩЕНИЕ НЕВЫПУКЛЫХ ОРИЕНТИРОВАННЫХ МНОГОУГОЛЬНИКОВ В ОДНОСВЯЗНОЙ НЕВЫПУКЛОЙ ОБЛАСТИ

М.А.Верхотуров, С.В.Петренко

Уфимский государственный авиационный технический университет, Уфа
e-mail: verhotur@vmk.ugatu.ac.ru, simonp@inbox.ru

Аннотация. В данной статье рассматривается решение задачи нерегулярного размещения ориентированных невыпуклых многоугольников в невыпуклую многоугольную область, основанное на двухступенчатом подходе: применении "жадного" алгоритма для нахождения первоначального решения и метода градиентного спуска для последующего его улучшения.

Ключевые слова: размещение, невыпуклый, упаковка.

1. Постановка задачи

Пусть задана область размещения в виде односвязного невыпуклого многоугольника S_0 и множество невыпуклых ориентированных многоугольников $J_0 = \{S_1, S_2, \dots, S_n\}$. Многоугольник S_0 задается в глобальной системе координат XOY , многоугольники S_1, S_2, \dots, S_n - в локальных системах XO_iY ($i=1, n$). Положение S_i в глобальной системе координат XOY определяется положением контрольной точки $v_i = (x_i, y_i)$. Пусть J - произвольное подмножество множества J_0 . Многоугольник, полученный размещением многоугольника S_i в точке v_i обозначим $S_i(v_i)$. Требуется найти подмножество J^* множества J_0 и размещение v^* многоугольников этого подмножества, удовлетворяющие условиям: $\forall S_i \in J^* S_i(v_i^*) \cap S_0 = S_i(v_i^*)$ и $\forall S_i, S_j \in J^* (j \neq i) \text{int}S_i(v_i^*) \cap \text{int}S_j(v_j^*) = \emptyset$. Для которых справедливо:

$$\sum_{J^*} \text{area}(S_i) = \max_{\forall J \subset J_0} \left\{ \sum_J \text{area}(S_i) \right\}$$

где $\text{area}(S_i)$ - площадь, занимаемая многоугольником S_i .

2. Метод решения

Процесс решения разбит на ряд этапов: построение модели области размещения; построение годографов функции плотного размещения многоугольников в области; укладка многоугольников; улучшение полученного решения методом градиентного спуска.

3. Построение модели области размещения

Цель этого этапа представить область размещения в виде пересечения определенных областей, ограниченных выпуклыми многоугольниками. Для этого для области размещения строится выпуклая оболочка, что позволяет представить область S_0 в виде: $S_0 = T_0 \cap \bar{T}_1 \cap \bar{T}_2 \cap \dots \cap \bar{T}_k$,

где T_0 - выпуклая оболочка многоугольника S_0 ;

T_1, T_2, \dots, T_k - дополнения области S_0 до выпуклой оболочки T_0 ;

k - количество этих областей.



Границы областей $\bar{T}_1, \bar{T}_2, \dots, \bar{T}_k$ могут быть заданы в виде в общем случае невыпуклых многоугольников. Такое представление соответствует определенному размещению невыпуклых многоугольников T_1, T_2, \dots, T_k внутри выпуклой области T_0 . Таким образом, задачу можно свести к задаче размещения многоугольников $T_1, T_2, \dots, T_k, S_1, S_2, \dots, S_n$ внутри выпуклой области T_0 , причем многоугольники T_1, T_2, \dots, T_k уже размещены.

Рис. 1. Модель невыпуклой области S_0 .

4. Построение годографа функции плотного размещения многоугольников в области

Для получения годографа (ф-функции) плотного размещения многоугольника S_i ($i=1, n$) в области $T_0 - D_{0i}$, необходимо сдвинуть ребра области T_0 на определенные вектора (рис. 2). Эти вектора получаются следующим образом: через точку отсчета v_i ($i=1, n$) многоугольника S_i проводится прямая, параллельная ребру области.



Среди всех вершин многоугольника S_i ищется точка, расположенная на минимальном расстоянии r (по направлению внутреннего вектора нормали к ребру - n) от данной прямой. Вектор смещения определяется как $-rn$. На этом этапе можно отсеять те многоугольники из набора S_1, S_2, \dots, S_n , которые не могут быть размещены внутри области T_0 , а, следовательно, в S_0 . Для них множество точек, принадлежащих D_{0i} , будет \emptyset . Пусть многоугольники S'_1, S'_2, \dots, S'_m те многоугольники из набора S_1, S_2, \dots, S_n , для которых $D_{0i} \neq \emptyset$.

Рис. 2. Построение ребра годографа для области.

5. Укладка многоугольников

Для укладки многоугольников в область T_0 для каждого многоугольника S'_1, S'_2, \dots, S'_m строится оценка. В качестве оценки многоугольника S'_i ($i=1, m$) берется площадь выпуклой оболочки этого многоугольника. Пусть набор $S''_1, S''_2, \dots, S''_m$ - упорядоченный по убыванию оценок набор многоугольников S'_1, S'_2, \dots, S'_m . Жадный алгоритм предполагает поочередное размещение многоугольников - одного за другим.

Перед размещением многоугольники $T_1, T_2, \dots, T_k, S''_1, S''_2, \dots, S''_m$ разбиваются на выпуклые, следовательно, получается набор $T_{11}, \dots, T_{1k_1}, T_{21}, \dots, T_{2k_2}, \dots, S''_{11}, \dots, S''_{1m_1}, \dots, S''_{m1}, \dots, S''_{mm_m}$ выпуклых многоугольников. Для каждого многоугольника S''_i ($i=1, m$) используется следующий алгоритм размещения.

При размещении многоугольника S''_i необходимо учитывать, что многоугольники $T_1, T_2, \dots, T_k, S''_1, S''_2, \dots, S''_{i-1}$ уже размещены в области. Получение допустимой области размещения для многоугольника S''_i и области $T_0 - D_{0i}$, было описано выше. Эта область может быть представлена в виде системы линейных неравенств. Допустимая область размещения многоугольника относительно многоугольников $T_1, T_2, \dots, T_k, S''_1, S''_2, \dots, S''_{i-1}$

можно записать как:

$$D_i^* = D'_{1i} \cap D'_{2i} \cap \dots \cap D'_{ki} \cap D_{1i} \cap D_{2i} \cap \dots \cap D_{i-1i},$$

где D'_{ji} и D_{qi} - допустимые области размещения многоугольника S_i'' относительно T_j ($j=1, k$) и S_q'' ($q=1, i-1$) соответственно.

Допустимая область некоторого невыпуклого многоугольника A относительно невыпуклого многоугольника B - D_{AB} , может быть получена как:

$$D_{AB} = \bigcap_{i=1}^{n_A} \left(\bigcap_{j=1}^{n_B} D_{A_i B_j} \right),$$

где n_A и n_B - число выпуклых многоугольников, на которые разбиваются многоугольники A и B соответственно;

A_i, B_j - выпуклые многоугольники разбиения многоугольников A и B соответственно ($i=1, n_A; j=1, n_B$);

$D_{A_i B_j}$ - годограф (ф-функция) плотного размещения выпуклого многоугольника B_j относительно выпуклого многоугольника A_i ($i=1, n_A; j=1, n_B$) [2].

Область ограничена выпуклым многоугольником, что позволяет представить ее в виде объединения линейных неравенств.

Таким образом, допустимую область размещения S_i'' относительно $T_1, T_2, \dots, T_k, S_1'', S_2'', \dots, S_{i-1}''$ можно представить в виде:

$$D_i^* = \bigcap_{j=1}^{K_i} D'_j,$$

где $K_i = m_i \left(\sum_{q=1}^k k_q + \sum_{q=1}^{i-1} m_q \right)$;

D'_j - объединение линейных неравенств ($j=1, K_i$).

Следовательно, допустимая область размещения S_i'' будет

$$D_i = D_{0i} \cap \left(\bigcap_{j=1}^{K_i} D'_j \right).$$

В результате допустимая область была представлена в виде конъюнкции системы и объединений неравенств. Каждое неравенство содержит две переменные - x_i'' и y_i'' , координаты точки отсчета многоугольника S_i'' - v_i'' .

Для области D_{0i} можно получить минимальное и максимальное значение y - y_{min} и y_{max} соответственно, а также максимальное значение x - x_{max} . Для нахождения координат точки размещения S_i'' выполняется перебор точек (M, y) , где $y_{min} \leq y \leq y_{max}$, $M > x_{max}$, с некоторым дискретным шагом по y . Для каждой такой точки ищется минимальное значение x , такое что $(x, y) \in D_i$. Наглядно показать процедуру поиска этой точки на структуре неравенств D_i можно, разместив \underline{x}_j и \bar{x}_j на оси MX . Здесь \underline{x}_j и \bar{x}_j - минимальное и максимальное расстояние от точки (M, y) до границ D_{0i} , если $j = 0$; D'_j , если $j=1, K_i$ по направлению вектора $(-1, 0)$. Среди всех значений \bar{x}_j выбирается максимальное допустимое, которое определяет точку (x, y) (рис. 3). Если такого допустимого значения нет, то для точки (M, y) невозможно получить точку (x, y) .

Среди всех точек (x, y) выбирается точка с минимальной абсциссой и считается точкой размещения многоугольника S_i'' . Если не найдена ни одна допустимая точка (x, y) , то считается, что многоугольник S_i'' не может быть размещен в области. Таким образом, используя данный подход можно получить допустимое размещение многоугольников в области S_0 .

Рис. 3. Поиск абсциссы точки размещения многоугольника.

6. Улучшение полученного решения методом градиентного спуска

Рассматриваемый в данной статье подход поиска локального экстремума разработан на основе метода Ю.Г.Стояна [1].

Получив некоторое решение жадным алгоритмом - $v = (v_1''', v_2''', \dots, v_l''')$, где v_i''' - координаты размещения в области S_0 уместившихся многоугольников S_i''' ($i=1, l$), можно его улучшить с точки зрения занятой части области S_0 , применяя алгоритм поиска локального экстремума. Для этого строится обобщенная допустимая область в виде структуры неравенств:

$$D = \bigcap_{i=1}^l D_i,$$

где D_i соответствует допустимой области размещения многоугольника S_i''' в области S_0 (построение такой области было выполнено на предыдущем этапе решения задачи). В каждую систему неравенств D_{0i} ($i=1, l$) добавляется дополнительное неравенство: $x_i''' - z + x_{imax}''' \leq 0$, где x_i''' - абсцисса точки S_i''' ($i=1, l$), z - дополнительная переменная, x_{imax}''' - максимальное значение абсциссы для многоугольника S_i''' ($i=1, l$). Все системы неравенств D_{0i} ($i=1, l$) можно объединить в систему

$$D_0 = \bigcap_{i=1}^l D_{0i}.$$

Таким образом, для минимизации занятой части области S_0 необходимо решить задачу: найти минимум z на допустимой области

$$D = D_0 \left(\bigcap_{i=1}^K D_i \right), \text{ где } K = \sum_{j=1}^l K_j$$

Задачу будем решать методом градиентного спуска, для этого перепишем ее в виде: требуется найти минимум линейной функции цели $C = (cx)$, $c=(0, 0, \dots, -1) \in R^{2l+1}$ и соответствующий ему вектор $x = (x_1''', y_1''', x_2''', y_2''', \dots, x_l''', y_l''', z) = (x_1, x_2, \dots, x_{2l+1}) \in R^{2l+1}$ на области допустимых решений

$$D = D_0 \left(\bigcap_{i=1}^K D_i \right), \text{ где } D_0 = \bigcap_{j=1}^{k_0} (n_j^0 x + b_j^0 \leq 0)$$

$$D_i = \bigcup_{j=1}^{k_i} (n_j^0 x + b_j^0 \leq 0)$$

При этом задана точка $v = (v_1, v_2, \dots, v_{2l+1}) \in D$ - начальное приближение.

Идея метода: на каждом шаге выбирается вектор движения $g \in R^{2l+1}$ и его длина α . Новое приближение берется как $v' = v + \alpha g$. Основная задача - выбрать направление движения так, чтобы оно максимально уменьшало функцию цели.

Пусть в точке v активны q неравенств (активным будем называть то неравенство (гиперплоскость в R^{2l+1}), которое в точке v превращается в равенство, $-(cn) \geq 0$ и точка v принадлежит границе области D). В этом случае задачу нахождения оптимального вектора g можно записать в виде:

найти $\hat{g} = \min(gc)$ при условиях:

1. $|g| = |c|$ (удобно привести $|c|$ к 1 $|g| = 1 \Leftrightarrow g^2 = 1$).
2. для любого активного неравенства j ($j=1, q$) $(v + g)n_j + b_j \leq 0$, (т.к. в точке v эти неравенства активные, то $vn_j + b_j = 0 \Rightarrow$ условие сведется к $gn_j \leq 0$).

Замечание: для каждой дизъюнкции неравенств число активных неравенств не может быть более одного. При этом активными не могут быть неравенства (гиперплоскости), соответствующие тем ребрам графов, которые были получены при делении исходных невыпуклых многоугольников на выпуклые. Т.е., если ребро выпуклого многоугольника A_i - отрезок, по которому было выполнено разбиение невыпуклого многоугольника A , то это ребро не должно участвовать в поиске активных ребер.

Условие 1 заменим более мягкими условиями $g_i \leq 1$ и $g_i \geq -1$. В этом случае, заменив вектор g на разность векторов γ и β , получим задачу линейного программирования:

$$\min(c\gamma - c\beta)$$

$$x_j = -n^j\gamma + n^j\beta, j=1, 2, \dots, q$$

$$y_i = -\gamma_i + \beta_i + 1, i=1, 2, \dots, 2l+1$$

$$z_i = \gamma_i - \beta_i + 1, i=1, 2, \dots, 2l+1$$

$$\gamma_i \geq 0, \beta_i \geq 0, i=1, 2, \dots, 2l+1$$

Начальное базисное решение $x=0, y=1, z=1, \gamma=0$ и $\beta=0$ следует из формы записи задачи. Следовательно, базисными переменными будут (x, y, z) . Решая данную задачу, найдем вектор направления движения g , имеющий минимальное расхождение с вектором уменьшения функции цели.

Для нахождения длины шага - α , выполняются следующие действия. Для конъюнкции D_0 ищется величина:

$$\alpha_0 = \min_{j=1, k_0} \left\{ -\frac{(vn^j) + b^j}{(gn^j)} \right\}$$

для каждой дизъюнкции D_i ($i=1, 2, \dots, K$):

$$\alpha'_i = \max_{j=1, k_i; (gn^j > 0)} \left\{ -\frac{(vn^j) + b^j}{(gn^j)} \right\}, \alpha''_i = \min_{j=1, k_i; (gn^j < 0)} \left\{ -\frac{(vn^j) + b^j}{(gn^j)} \right\},$$

$$J = \{j = 1, k_i : (vn^j) + b^j \leq 0, (gn^j) = 0\}, \alpha = \begin{cases} +\infty, & J \neq \emptyset \text{ или } \alpha'_i \geq \alpha''_i \\ \alpha'_i, & \text{иначе} \end{cases}$$

Это позволяет отсеять те точки пересечения, которые принадлежат гиперплоскостям ограничений, но не принадлежат непосредственно границе допустимых областей.

$$\alpha = \min_{i=0, K} \{\alpha_i\}$$

В результате получаем следующее приближение $v' = v + \alpha g$. Локальный экстремум будет достигнут тогда, когда в результате определения направления движения будет получен вектор g , такой, что $(gc) \geq 0$ (т.к. в результате выше описанных действий полученный вектор g имеет максимальное скалярное произведение с вектором уменьшения целевой функции, и если, $(gc) \geq 0$, то изменение v в любом допустимом направлении приведет к увеличению значения целевой функции).

На каждом шаге поиска локального экстремума проверяется возможность заполнения свободных областей не попавшими в область многоугольниками по алгоритму, описанному в пункте 5. В том случае, если некоторый многоугольник S_j'' может быть размещен, он добавляется к набору $S_1''', S_2''', \dots, S_l'''$ и дальнейший поиск локального экстремума ведется уже по набору $S_1''', S_2''', \dots, S_l''', S_j''$.

7. Заключение

Метод отличается простой реализации и надежностью работы, т.к. при решении используются только отношения между выпуклыми многоугольниками. Особенностью алгоритма является то, что полученное решение является точкой локального экстремума задачи.

Данный подход может быть применен при заполнении свободных полостей в уже готовых картах раскроя, составлении карт раскроя с "нуля". Метод может быть применен (с небольшими изменениями) к любому промежуточному размещению, что позволяет использовать его в совокупности с другими методами.

Список литературы

- [1] Ю.Г.Стоян, М.В.Новожилова, А.В.Карташов *Математическая модель и оптимизация линейных $E_k(R^2)$ - задач размещения* - Препринт-353, ИПМАШ АН УССР, Харьков 1991.
- [2] Julia A. Bennell, Kathryn A. Dowsland, William B. Dowsland *The irregular cutting-stock problem - a new procedure for deriving the no-fit polygon.* - Computers & Operation Research 28(2001), p.271-287.

AN ALLOCATION PROBLEM OF NON-CONVEX ORIENTED POLYGONS IN A SINGLE-CONNECTED NON-CONVEX SURFACE

M.A.Verkhoturov, S.V.Petrenko

Ufa State Aviation Technical University

e-mail: verhotur@vmk.ugatu.ac.ru, simonp@inbox.ru

Abstract. We consider the decision of an allocation problem of non-convex oriented polygons in a single-connected non-convex surface, based on the two-level approach: application "greedy" algorithm for a finding of the initial decision and a gradient descend method for its subsequent improvement.

Key words: irregular nesting, ICSP

О ЗАДАЧЕ ВЫБОРА КОМПЛЕКТАЦИИ СИСТЕМЫ МАШИН И УЗЛОВ С НЕЛИНЕЙНОЙ ФУНКЦИЕЙ ПРОИЗВОДСТВЕННЫХ ЗАТРАТ¹

Э.Х.Гимади, Е.Н. Гончаров

Институт математики СО РАН им. С.Л.Соболева, Новосибирск
e-mail: gimadi@math.nsc.ru, gon@math.nsc.ru

Аннотация. В работе рассматривается задача выбора комплектов оборудования, состоящего из системы машин (комплектов) и составляющих их узлов (изделий). Целевая функция состоит в минимизации суммы начальных затрат, связанных с вводом в действие выбранных изделий и комплектов, и затрат, связанных с удовлетворением спроса потребителей. Рассматривается обобщение задачи на случай нелинейной функции затрат на производство узлов. Предлагается алгоритмическая реализация этой обобщенной задачи, приводится результат численного эксперимента.

Ключевые слова: дискретная оптимизация, задача стандартизации, метод ветвей и границ, нелинейная функция затрат.

Введение

Задача стандартизации, или, как ее еще называют, задача выбора типажа изделий, является сравнительно хорошо изученной задачей. Интерес к этой задаче связан со значительными приложениями этой задачи на практике. Широко известная и также популярная в практических приложениях задача размещения предприятий имеет точно такую же формулировку. Задача NP–трудна даже в так называемой одноуровневой постановке. Тем не менее для решения такой задачи для линейной производственной функции хорошо зарекомендовала себя техника ветвей и границ с использованием нижних оценок, вычисляемых с помощью так называемых тупиковых и квазитупиковых решений двойственных релаксаций рассматриваемых задач [1], [3]. В [1] показано, что для нелинейной функции затрат имеет место сведение к расширенной линейной модели, для которой успешно применяется упомянутая выше техника.

С особой остротой вопросы стандартизации возникают при создании систем изделий (машин, механизмов) для выполнения тех или иных функций (под системой понимается набор изделий различного функционального назначения для полного оснащения всего цикла производства). В качестве примера такой задачи можно привести планирование комплектации тракторов и входящих в них узлов, предназначенных для выполнения различных сельскохозяйственных, строительных, дорожных и других работ в разных ландшафтно-климатических условиях. Другим примером может служить проблема оснащения крупного вычислительного комплекса на корабле, летательном аппарате, автоматизированной производственной линии и др. промышленных объектах. Проблему можно представить себе в виде двухуровневой задачи стандартизации, которая имеет дело с выбором элементов двух уровней. Непосредственную работу выполняют элементы (машины, комплекты) второго уровня, а элементы первого уровня (узлы, модули) используются для формирования этих комплектов.

¹Исследование выполнено при финансовой поддержке Российского фонда фундаментальных исследований (проект 02-01-01153) и INTAS (проект 04-77-7173)

Приведем математическую формулировку двухуровневой задачи. Пусть известны множества: J — видов работ, M — возможных типов комплектов и I — возможных видов составных частей (модулей, узлов), из которых формируются комплекты. Для работ каждого вида $j \in J$ указан объем b_j . Каждый комплект $k \in M$ состоит из определенного набора $I_k \subseteq I$ составных частей (модулей, узлов); μ_{ik} — число узлов i в комплекте k . Заданы также стоимостные характеристики: g'_i — стоимость производства одного узла i ; g_i — фиксированные затраты на проектирование и ввод в действие узла i ; $c_k = \sum_{i \in I_k} g'_i \mu_{ik}$ — стоимость узлов, входящих в комплект k ; f_k — фиксированные затраты на проектирование и ввод в действие комплекта k . Для комплекта определенного типа $k \in M$ известны фиксированные затраты c_k , связанные с вводом его в действие, стоимость \tilde{c}_k сборки комплекта k , а также затраты c_{kj} на эксплуатацию одного такого комплекта в условиях использования на работе определенного вида $j \in J$. Требуется выбрать такую систему узлов $I' \subset I$ и комплектов $M' \subset M$ и такое назначение комплектов на выполняемые работы, чтобы все работы были выполнены, и суммарные затраты, связанные с производством узлов, комплектацией и выполнением работ, были бы минимальными.

С использованием введенных обозначений задача записывается следующим образом

$$\sum_{i \in I(M')} g_i + \sum_{k \in M'} f_k + \sum_{j \in J} b_j \min_{k \in M'} (c_k + c_{kj} + \tilde{c}_k) \rightarrow \min_{M' \subset M}, \quad (1)$$

где $I(M') = \bigcup \{I_k, k \in M'\}$.

В работе [3] показано, что использование упомянутой техники также приводит к хорошим вычислительным результатам.

1. Задача выбора комплектов оборудования в случае нелинейной производственной функции

Рассматривается нелинейная (кусочно-линейная, неубывающая, вогнутая, в нуле равная нулю) производственная функция

$$g_i(v_i) = \begin{cases} 0, & \text{if } v_i = 0, \\ g_i^r + g_{ir} v_i, & \text{if } v_i \in \mathcal{V}_i^r, r \in \mathcal{R}_i, \end{cases} \quad (2)$$

где

R_i — число диапазонов линейности функции $g_i(v_i)$;

$\mathcal{R}_i = \{1, 2, \dots, R_i\}$, $i \in I$ — совокупность диапазонов линейности функции $g_i(v_i)$;

v_i^r — границы диапазонов линейности, причем $v_i^0 = 0$; $v_i^{R_i} = \infty$;

$\mathcal{V}_i^r = (v_i^{r-1}, v_i^r]$, $r \in \mathcal{R}_i$ — совокупность границ диапазонов линейности функции $g_i(v_i)$;

g_i^r и g_{ir} — фиксированные затраты и стоимость производства одного узла i в диапазоне линейности r , соответственно.

К сожалению, прямое сведение нелинейной задачи к расширенной линейной, показанное в работе [1] применительно к нелинейной одноуровневой задаче, в общем случае оказывается неприемлимым, поскольку для каждой машины k составляющие ее узлы могут производиться с разными диапазонами серийности.

Ниже мы остановимся на подклассе нелинейной двухуровневой задачи, для которого справедливо

Допущение 1. Мощности множеств I_k , $k \in M$, и \mathcal{R}_i , $i \in I$ ограничены сверху мощностями множеств M и I соответственно.

Обозначим через S_k множество всех $|I_k|$ -наборов $(r_i)(i \in I_k)$ в декартовом произведении $\prod_{i \in I_k} \mathcal{R}_i$.

Для всех $s \in S_k, k \in M, j \in J$ определим числовые коэффициенты:

$$d_{kj}^s = b_j(c_{kj} + c_k^s + \tilde{c}_k),$$

где c_k^s — стоимость узлов $i \in I_k$, необходимых для комплекта k , если эти узлы производятся в соответствующих диапазонах линейности производственной функции (r_i^s) , $i \in I_k, s \in S_k$:

$$c_k^s = \sum_{i \in I_k} \mu_{ik} g_{ir_i^s}.$$

В принятых предположениях нелинейная двухуровневая модель может быть представлена в виде, аналогичном преобразованной нелинейной одноуровневой модели.

Определим новые переменные:

- y_i^r равно 1, если узел i производится в диапазоне линейности $r \in \mathcal{R}_i$, и 0, иначе;
- $x_{kj}^s = 1$, если работа j выполняется машиной k , состоящей из узлов из множества I_k в диапазонах линейности (r_i^s) , $i \in I_k, s \in S_k$, и $x_{kj}^s = 0$, иначе.

Тогда решение нелинейной модели может быть получено из решения следующей задачи линейного целочисленного программирования:

$$\min_{x,y,z} \left\{ \sum_{i \in I} \sum_{r \in \mathcal{R}_i} g_i^r y_i^r + \sum_{k \in M} f_k z_k + \sum_{k \in M} \sum_{j \in J_k} \sum_{s \in S_k} d_{kj}^s x_{kj}^s \right\}, \quad (3)$$

$$\sum_{k \in M} \sum_{s \in S_k} x_{kj}^s = 1, \quad j \in J; \quad (4)$$

$$z_k \geq x_{kj}^s, \quad j \in J_k, \quad s \in S_k, \quad k \in M; \quad (5)$$

$$y_i^r \geq \sum_{k \in M_i} \sum_{s \in S_k} x_{kj}^s, \quad r \in \mathcal{R}_i, \quad \forall i, j; \quad (6)$$

$$y_i^r \in \{0, 1\}, \quad r \in \mathcal{R}_i, \quad i \in I; \quad (7)$$

$$z_k, x_{kj}^s \in \{0, 1\}, \quad s \in S_k, \quad \forall k, j. \quad (8)$$

Утверждение. В условиях Допущения 1 существует оптимальное решение расширенной двухуровневой задачи (3)–(8), в которой для каждого узла выбирается только один диапазон линейности производственной функции (2).

Из утверждения следует, что в принятых предположениях на входные данные для решения двухуровневой задачи с нелинейной производственной функцией может быть использована техника, успешно опробованная для решения линейной двухуровневой модели [3].

2. Пример решения задачи выбора комплектов оборудования в случае нелинейной производственной функции

Приведем численный пример задачи выбора комплектов оборудования и поиска серийности производства изделий, входящих в комплекты.

Таблица 1: Список работ

J	1	2	3	4	5	6	7	8	9	10	11	12
d_j	6	6	4	3	7	7	5	9	9	4	3	9

Таблица 2: Список уровней производства продукции для каждого изделия

	Изделия									
	1	2	3	4	5	6	7	8	9	10
1 серия	0-10	0-29	0-27	0-25	0-17	0-29	0-20	0-14	0-23	0-14
2 серия	10-42	29-114	27-73	25-98	17-55	29-72	20-50	14-58	23-93	14-60
3 серия	> 42	> 114	> 73	> 98	> 55	> 72	> 50	> 58	> 93	> 60

Пусть требуется выполнить 12 различных типов работ, количество работ каждого типа приведено в таблице 1. Имеется 10 типов изделий, каждое из которых может производиться по одному из трех уровней серийности. В таблице 2 приводится список уровней производства продукции для каждого изделия, в таблице 3 — начальные затраты на производство изделий $g_i^0(v)$ в зависимости от объемов их производства, в таблице 4 — стоимость производства изделий $g_{ir}(v)$ в зависимости от объемов их производства.

Таблица 3: Начальные затраты на производство изделий $g_i^0(v)$ в зависимости от объемов их производства

	Изделия									
	1	2	3	4	5	6	7	8	9	10
1 серия	74	98	99	86	87	83	75	96	75	77
2 серия	174	388	369	386	308	402	295	236	282	203
3 серия	678	1870	1099	1660	803	1194	1045	932	1119	743

Из этих изделий могут быть собраны комплекты (машины).

Задано 20 комплектов, каждый из которых содержит ровно по 4 изделия,

Список этих комплектов, их состав, а также список выполняемых работ для каждого комплекта приведены в таблице 5.

Эксплуатационные затраты c_{kj} , $k \in M$, $j \in J$ на использование комплектов при выполнении работы были приняты равными нулю. Результаты вычислений приведены в таблице 6. Мы видим, что были выбраны 2 комплекта, 11-й и 17-й, а для их комплектования выбраны 6 типов изделий, каждый со своим объемом производства. Оптимальное значение целевой функции при этом составило 25682.

Таблица 4: Стоимость производства изделий $g_{ir}(v)$ в зависимости от объемов их производства

	Изделия									
	1	2	3	4	5	6	7	8	9	10
1 серия	139	74	119	131	149	111	91	125	90	56
2 серия	129	64	109	119	136	100	80	115	81	47
3 серия	117	51	99	106	127	89	65	103	72	38

Таблица 5: Список комплектов (машин)

Комплекты	Состав комплектов	Допустимые работы
1	4,6,7,8	1,2,3,4,5,6,7,8,10,11,12
2	2,3,4,5	1,3,4,5,6,8,9,10,11
3	2,6,7,8	1,2,3,4,5,6,8,10,11,12
4	1,5,6,10	1,3,4,5,6,8,9
5	2,4,7,9	1,2,3,4,5,6,7,8,9,10,11,12
6	1,3,8,10	1,3,4,5,6,7,9,11,12
7	1,2,6,8	1,2,4,5,6,7,8,9,10,11
8	2,3,5,6	1,2,3,4,5,6,7,8,9,10,11,12
9	1,2,7,8	1,2,3,4,5,6,7,8,9,10,11
10	2,3,7,9	1,2,3,4,6,7,8,9,11
11	2,4,6,10	1,3,4,5,6,7,8,9,10,12
12	5,7,8,9	1,2,3,4,5,6,7,8,9,11
13	2,5,8,9	1,2,3,4,5,8,10,11,12
14	4,5,8,10	1,2,3,5,8,9,10,12
15	1,2,3,9	2,4,5,6,7,8,10
16	1,5,7,10	1,2,3,4,5,6,7,8,9,10,11,12
17	2,3,6,7	1,2,3,4,5,8,9,10,11,12
18	3,7,8,10	1,2,3,7,8,9,10,11,12
19	1,4,5,7	1,2,3,5,6,9,11,12
20	1,2,3,5	2,3,4,5,7,8,9,10,11,12

Таблица 6: Оптимальное решение

Оптимальное множество изделий	Серийность производства изделий	Объем производства	Оптимальное множество машин	Потребители
2	2	72	11	1,3,4,5,6,7,8,9,10,12
3	1	9	17	2, 11
4	2	63		
6	3	72		
7	1	9		
10	3	63		

Список литературы

- [1] Береснев В. Л., Гимади Э. Х., Дементьев В. Т. Экстремальные задачи стандартизации. Новосибирск: Наука, 1978.
- [2] Cornuejols G., Nemhauser G. L. and Wolsey L. A. The Uncapacitated Facility Location Problem, In book edited by P. B. Mirchandani and R. L. Francis, Wiley&Sons, 1990.
- [3] Гончаров Е. Н. Метод ветвей и границ для простейшей двухуровневой задачи размещения предприятий // Дискрет. анализ и исслед. операций. Сер. 2. 1998. Т. 5, № 1. С. 19-39.
- [4] V.B. Algin, A. Dolgui, E.Kh. Gimadi, E.N. Goncharov, Optimal Design of Machines Dedicated to a Given Set of Jobs, Proc. Intern. Conf. DOM 2004, Omsk, 2004.

АНАЛИЗ УСТОЙЧИВОСТИ ПО ЦЕЛЕВОЙ ФУНКЦИИ НЕКОТОРЫХ АЛГОРИТМОВ ДИСКРЕТНОЙ ОПТИМИЗАЦИИ

М.В. Девятерикова, А.А. Колоколов

Омский государственный технический университет, Омск
e-mail: devy@omgtu.ru

Омский филиал Института математики СО РАН, Омск
e-mail: kolo@iitam.omsk.net.ru

Аннотация. Вводится определение устойчивости по целевой функции для достаточно широкого класса алгоритмов дискретной оптимизации. Изучается устойчивость некоторых алгоритмов целочисленного линейного программирования при малых колебаниях целевых функций решаемых задач. Показывается, что имеются как устойчивые, так и неустойчивые варианты алгоритмов перебора L -классов. Установлена неустойчивость ряда алгоритмов ветвей и границ.

Ключевые слова: дискретная оптимизация, целочисленное программирование, устойчивость алгоритмов, перебор L -классов, метод ветвей и границ.

Введение

Исходные данные многих прикладных задач дискретной оптимизации определяются неточно, с некоторой погрешностью. Поэтому исследование устойчивости как задач, так и алгоритмов их решения является актуальным направлением в указанной области [1-5].

В работах [4, 5] проведено исследование устойчивости ряда алгоритмов целочисленного программирования (ЦП) при достаточно малых изменениях релаксационных множеств задач. Было показано, что алгоритм перебора L -классов и дробный двойственный процесс с вполне регулярными отсечениями устойчивы на задачах ЦП с замкнутыми ограниченными релаксационными множествами. Установлено также, что метод ветвей и границ на задачах целочисленного и булева линейного программирования этим свойством не обладает.

В данной работе вводится понятие устойчивости алгоритма решения задачи дискретной оптимизации при малых колебаниях целевой функции. Изучаются некоторые алгоритмы целочисленного линейного программирования (ЦЛП). Доказано, что существуют как устойчивые, так и неустойчивые по целевой функции алгоритмы перебора L -классов. Установлена неустойчивость ряда алгоритмов ветвей и границ (в рамках схемы Лэнд и Дойг).

1. Основные определения и обозначения

Пусть Ψ - бесконечное семейство замкнутых ограниченных множеств из R^n , Ω - непустое множество из Ψ , Φ - бесконечный подкласс пространства $C(\Omega)$ непрерывных вещественнозначных функций, определенных на Ω , $f \in \Phi$. Рассматривается следующая задача ЦП:

$$f(x) \rightarrow \max, \quad x \in (\Omega \cap Z^n). \quad (1)$$

Пусть $\varepsilon > 0$ и $\| \cdot \|$ - норма в пространстве $C(\Omega)$, т.е. $\|f\| = \sup\{f(x) : x \in \Omega\}$. Будем говорить, что функция f_ε является ε -изменением функции f , если выполняется

неравенство $\left\| \frac{f}{\|f\|} - \frac{f_\varepsilon}{\|f_\varepsilon\|} \right\| \leq \varepsilon$.

Заметим, что для произвольной $f \in \Phi$ существует $\hat{\varepsilon} > 0$ такое, что при любом $\varepsilon < \hat{\varepsilon}$ выполняется включение $Z^*(f_\varepsilon) \subseteq Z^*(f)$, где $Z^*(f)$ - множество оптимальных решений задачи (1) с целевой функцией f . Такое включение используется в исследовании устойчивости задач дискретной оптимизации [3].

Обозначим через \mathcal{A} некоторый алгоритм, порождающий последовательность $S_{\mathcal{A}}(f, \Omega)$ точек из Ω при решении задачи (1) с целевой функцией f и релаксационным множеством Ω . Будем говорить, что алгоритм \mathcal{A} устойчив по целевой функции на задачах ЦП с целевыми функциями из Φ и релаксационными множествами из Ψ , если для любой функции $f \in \Phi$ и произвольного $\Omega \in \Psi$ существуют $\varepsilon_f > 0$ и полином $p(n)$, не зависящий от f и Ω , такие, что неравенство

$$|S_{\mathcal{A}}(f_\varepsilon, \Omega)| \leq p(n)|S_{\mathcal{A}}(f, \Omega)|$$

справедливо для любого ε -изменения f_ε при $\varepsilon < \varepsilon_f$.

Далее в этой статье мы будем рассматривать случай задачи ЦП, когда Ψ - семейство многогранников, а множество Φ состоит из линейных функций.

2. Метод перебора L -классов

Для анализа и решения задач ЦП был предложен метод перебора L -классов [6], который основан на L -разбиении пространства R^n . Применение этого метода дало хорошие результаты для ряда задач ЦП, например, для задачи о покрытии множества и задачи выполнимости.

Дадим определение L -разбиения. Точки $x, y \in R^n$ ($x \succ y$) называются L -эквивалентными, если не существует точки $z \in Z^n$, для которой $x \succeq z \succeq y$. Здесь \succ, \succeq - символы лексикографического сравнения. Эквивалентные точки образуют классы L -разбиения. Для любого множества $X \subseteq R^n$ фактор-множество X/L называется L -разбиением множества X , а элементы из X/L называются L -классами.

Рассмотрим идею метода перебора L -классов для задачи (1). Основной шаг метода заключается в переходе от одного L -класса релаксационного множества Ω к другому в порядке лексикографического убывания с учетом рекордного значения целевой функции $f(x)$. В процессе решения задачи порождается последовательность S точек $x^{(t)} \in \Omega$, обладающая свойствами:

- 1) $x^{(t)} \succ x^{(t+1)}$, $t = 1, 2, \dots$;
- 2) все точки $x^{(t)}$ принадлежат различным L -классам;
- 3) если множество $\Omega \cap Z^n$ состоит из более чем одной точки, то S содержит подпоследовательность целых точек $Q = \{z^{(t_k)}, k = 1, \dots, q\}$ такую, что $f(x^{(t)}) > f(z^{(t_k)})$ при $t > t_k$.

В рассматриваемом ниже варианте алгоритма перебора L -классов (далее этот алгоритм обозначается LC) процесс начинается с лексикографически максимальной точки $x^{(1)} \in \Omega$. Текущие точки $x^{(t)}$ строятся посредством нахождения лексикографического максимума вспомогательных подзадач линейного программирования (ЛП). Это может быть осуществлено, например, с помощью лексикографического двойственного симплекс-метода. Алгоритм завершает работу, если не удастся найти очередной L -класс. В случае, когда задача разрешима, лучшее из найденных целочисленных решений является оптимальным. Так как множество Ω ограничено, то алгоритм за конечное число шагов либо находит оптимум, либо устанавливает, что задача не имеет решения.

Алгоритм LC

Шаг 1. Определить начальное значение рекорда $rec = -\infty$.

Найти $x' = \text{lexmax } \Omega$. Возможны два случая.

1.1 Если $x' \in Z^n$, то вычислить новый рекорд $rec = f(x')$,
положить $p = n + 1$, $x'' = x'$ и перейти на шаг 3.

1.2 В случае $x' \notin Z^n$ перейти на шаг 2.

Шаг 2. Поиск следующего L -класса ("ход вниз").

Пусть $x'' = x'$. Найти $p = \min\{j : x''_j \neq \lfloor x''_j \rfloor, j = 1, \dots, n\}$.

Решить подзадачу:

найти $x' = \text{lexmax}\{x \in \Omega : f(x) \geq rec, x_1 = x''_1, \dots, x_{p-1} = x''_{p-1}, x_p \leq \lfloor x''_p \rfloor\}$.

Возможны следующие случаи.

2.1 Если эта подзадача не имеет решений и $p = 1$,

то перейти на шаг 4.

2.2 Если подзадача не имеет решений и $p > 1$, то на шаг 3.

2.3 Если подзадача имеет решение $x' \in Z^n$, то
обновить рекорд $rec = f(x')$, положить $p = n + 1$, $x'' = x'$,
перейти на шаг 3.

2.4 Если $x' \notin Z^n$, то перейти на шаг 2.

Шаг 3. Поиск следующего L -класса ("ход вверх").

Положить $\varphi = p - 1$. Решить подзадачу:

найти $x' = \text{lexmax}\{x \in \Omega : f(x) \geq rec, x_1 = x''_1, \dots, x_{\varphi-1} = x''_{\varphi-1}, x_\varphi \leq x''_\varphi - 1\}$.

Возможны следующие случаи.

3.1 Если подзадача не имеет решений и $\varphi = 1$, то на шаг 4.

3.2 Если подзадача не имеет решений и $\varphi > 1$, то
положить $p = \varphi$ и перейти на шаг 3.

3.3 Если получено решение $x' \in Z^n$, то
обновить рекорд $rec = f(x')$, положить $p = n + 1$, $x'' = x'$,
перейти на шаг 3.

3.4 Если $x' \notin Z^n$, то перейти на шаг 2.

Шаг 4. Процесс решения завершается. Лучшее найденное

целочисленное решение является оптимальным.

Если такого нет, то исходная задача не имеет допустимых решений.

Шаг 1 в алгоритме является предварительным и выполняется один раз. Основные итерации включают шаги 2 и 3.

С использованием свойств L -разбиения достаточно малых расширений релаксационного множества задачи ЦП [7] нами доказано следующее утверждение.

Лемма. Пусть $\Omega \in \Psi$ и $f \in \Phi$. Тогда существует $\varepsilon' > 0$ такое, что для любого ε -изменения $f_\varepsilon \in \Phi$ при $\varepsilon < \varepsilon'$ выполняется соотношение

$$|S_{LC}(f_\varepsilon, \Omega)| \leq 4(n+1)^2 |S_{LC}(f, \Omega)|.$$

Из леммы непосредственно следует устойчивость рассматриваемого алгоритма перебора L -классов.

Теорема. Алгоритм LC устойчив по целевой функции на задачах ЦП вида (1) с целевыми функциями из Φ и релаксационными множествами из Ψ .

Рассмотрим алгоритм $LC1$, который является модификацией алгоритма LC и приводит к более быстрому поиску оптимального решения. Этот алгоритм отличается от LC тем, что ограничение по рекорду $f(x) \geq rec$ заменено неравенством $f(x) \geq rec + \delta$, где δ определяется следующим образом. Пусть

$$\hat{\delta} = \inf\{|f(z') - f(z'')| : z', z'' \in (\Omega \cap Z^n) \text{ и } f(z') \neq f(z'')\}.$$

Так как Ω замкнуто и ограничено, то $\hat{\delta} > 0$. Выберем $0 < \delta \leq \hat{\delta}$. Если все коэффициенты целевой функции целочисленны, можно взять $\delta = 1$.

Покажем, что алгоритм $LC1$ не является устойчивым. Для этого построим семейство P задач ЦЛП, на котором число L -классов, просматриваемых алгоритмом $LC1$, растет произвольным образом при некоторых ε -изменениях $f_\varepsilon \in \Phi$ для любого $\varepsilon > 0$. Семейство P имеет вид:

$$\begin{aligned} f(x) &= x_2 \rightarrow \max, \\ \alpha_1 &\leq x_1 \leq \alpha_2, \beta_1 \leq x_2 \leq \beta_2, \\ x_1, x_2 &\in Z, \end{aligned}$$

где $\alpha_1, \alpha_2, \beta_1, \beta_2$ - целочисленные параметры.

Алгоритм $LC1$ для задач семейства P находит оптимальное решение $z^* = (\alpha_2, \beta_2)$ путем просмотра одного L -класса, т.е. $|S_{LC1}(f, \Omega)| = 1$. Определим следующее ε -изменение $f_\varepsilon \in \Phi$:

$$f_\varepsilon(x) = -\hat{\varepsilon} x_1 + x_2, \hat{\varepsilon} > 0.$$

При решении модифицированной задачи число пройденных алгоритмом $LC1$ L -классов при любом $\hat{\varepsilon} > 0$ равно $\alpha_2 - \alpha_1 + 1$. Таким образом, алгоритм $LC1$ не является устойчивым, так как рост длины последовательности $S_{LC1}(f, \Omega)$ зависит не от размерности пространства, а от параметров релаксационного множества задачи.

Для алгоритма LC на семействе P имеет место равенство $|S_{LC}(f_\varepsilon, \Omega)| = |S_{LC}(f, \Omega)|$ при рассмотренных изменениях целевой функции.

3. Метод ветвей и границ

Нами изучаются алгоритмы ветвей и границ решения задачи (1), основанные на схеме Лэнд и Дойг. В этой схеме ветвление производится по некоторой дробной координате оптимального решения текущей задачи ЛП, а верхними границами являются оптимальные значения целевых функций получаемых задач ЛП. Последовательность S в данном случае состоит из оптимальных решений указанных задач.

Установлено, что рассматриваемые алгоритмы ветвей и границ не являются устойчивыми по целевой функции на классе задач ЦЛП с ограниченными замкнутыми релаксационными множествами. Более подробно изучен алгоритм ветвей и границ \tilde{A} с ветвлением по первой дробной координате оптимального решения соответствующей задачи ЛП и симплекс-методом, находящим лексикографически минимальное решение задачи линейного программирования. Нами показано, что длина последовательности S , порождаемой алгоритмом \tilde{A} может увеличиваться произвольным образом при достаточно малых изменениях целевых функций задач. Для этого построено следующее семейство задач с параметрами p, q :

$$\begin{aligned} f(x) &= -x_3 \rightarrow \max, \\ 2(p-1)x_1 - p x_2 &= 0, \\ 2q x_1 - p x_3 &\geq 0, \end{aligned}$$

$$x_1 \leq \frac{p}{2}, x_3 \geq 0, \\ x_j \in Z, j = 1, 2, 3,$$

где $q \in Z_+$, p - простое.

Алгоритм \tilde{A} для задач данного семейства находит оптимальное решение $z^* = (0, 0, 0)$ за одну итерацию, т.е. $|S_{\tilde{A}}(f, \Omega)| = 1$. Рассмотрим следующее ε -изменение $f_\varepsilon \in \Phi$ целевой функции задачи семейства:

$$f_\varepsilon(x) = \hat{\varepsilon} x_1 + \hat{\varepsilon} x_2 - x_3, \hat{\varepsilon} > 0.$$

Модифицированная задача имеет то же самое оптимальное решение $z^* = (0, 0, 0)$, однако процесс решения начинается с оптимального решения соответствующей задачи ЛП, т.е. точки $\bar{x} = (\frac{p}{2}, p - 1, 0)$. При этом длина последовательности $S_{\tilde{A}}(f_\varepsilon, \Omega)$ превосходит значение параметра p для достаточно больших p . Таким образом, алгоритм \tilde{A} не является устойчивым по целевой функции для задач ЦЛП.

Проведенные исследования устойчивости по целевой функции алгоритмов дискретной оптимизации показывают, что число итераций многих алгоритмов может резко увеличиваться даже при малых колебаниях исходных данных.

Список литературы

- [1] Э.Н. Гордеев, В.К. Леонтьев *Общий подход к исследованию устойчивости решений в задачах дискретной оптимизации.* - ЖВМ и МФ, 1996, т. 36, № 1, с. 66-72.
- [2] В.А. Емеличев, Д.П. Подкопаев *О количественной мере устойчивости векторной задачи целочисленного программирования.* - ЖВМ и МФ, 1998, т. 38, № 11, с. 1801-1805.
- [3] И.В. Сергиенко, Л.Н. Козерацкая, Т.Т. Лебедева *Исследование устойчивости и параметрический анализ дискретных оптимизационных задач.* - Киев: Наукова думка, 1995, 170 с.
- [4] М.В. Девятерикова, А.А. Колоколов *Анализ устойчивости некоторых алгоритмов дискретной оптимизации.* - Автоматика и телемеханика, 2004, № 3, с. 48-54.
- [5] А.А. Kolokolov, M.V. Devyaterikova *Stability analysis of some discrete optimization algorithms.* - Proceedings of the Second International Workshop "Discrete Optimization Methods in Productions and Logistics", Omsk: Nasledie Dialog-Sibir Pbs., 2004, p. 180-184.
- [6] А.А. Колоколов *Применение регулярных разбиений в целочисленном программировании.* - Известия вузов. Математика, 1993, № 12, с. 11-30.
- [7] А.А. Колоколов, М.В. Девятерикова *Анализ устойчивости L-разбиения множеств в конечномерном пространстве.* - Дискретный анализ и исследование операций, Новосибирск, 2000, сер. 2, т. 7, № 2, с. 47-53.

ANALYSIS OF STABILITY OF SOME DISCRETE OPTIMIZATION ALGORITHMS UNDER SMALL ENOUGH VARIATIONS OF OBJECTIVE FUNCTION

M.V. Devyaterikova, A.A. Kolokolov

Omsk state technical university, Omsk

email: devy@omgtu.ru

Omsk branch of institute of mathematics of Siberian branch of RAS, Omsk

e-mail: kolo@iitam.omsk.net.ru

Abstract. We suggest a definition of stability of some discrete optimization algorithms under small enough variations of objective functions of the optimization problems. Stability of integer programming algorithms is investigated. It is shown that there exist as stable as unstable modifications of L -class enumeration algorithms for integer linear programming problem. Also it is proved that some branch-and-bound algorithms (Land and Doig scheme) are not stable.

Key words: discrete optimization, integer programming, algorithm stability, L -class enumeration, branch-and-bound method.

О МИНИМАКСНОЙ И МИНИСУММНОЙ ЗАДАЧАХ РАЗМЕЩЕНИЯ НА ПЛОСКОСТИ С ЗАПРЕЩЕННЫМИ ОБЛАСТЯМИ¹

Г.Г. Забудский

Омский филиал Института математики СО РАН, Омск
e-mail: zabudsky@iitam.omsk.net.ru

Аннотация. Рассматриваются задачи оптимального размещения объектов на плоскости вне прямоугольных запрещенных зон. Размещаемые объекты взаимосвязаны и имеют связи с объектами, расположенными на той же плоскости. Критериями являются минимизация максимального взвешенного расстояния или суммарной стоимости связей между объектами. Расстояния измеряются в метрике L_1 . Построены соответствующие модели целочисленного линейного программирования, кратко описаны алгоритмы решения задач.

Ключевые слова: задача размещения, целочисленное программирование, минимаксный, минисуммный, плоскость, запрещенная зона.

Введение

Задачи размещения взаимосвязанных объектов в некоторых областях с условием их расположения вне запрещенных зон часто встречаются на практике [1, 2, 3]. Запрещенными зонами могут быть элементы географического ландшафта (горы, реки, озера), объекты деятельности человека: здания, санитарные зоны и другие участки. Кроме запрещенных зон, на плоскости расположены объекты, имеющие связи (например, трубопроводные) с размещаемыми. Запрещенные зоны аппроксимированы прямоугольниками, и расстояния между объектами определены в метрике L_1 . Требуется разместить объекты вне указанных зон таким образом, чтобы минимизировать максимальное взвешенное расстояние или суммарную стоимость связей между объектами.

В п.1 приведены постановки задач. В п.2 построены модели целочисленного линейного программирования (ЦЛП) с булевыми переменными для указанных задач. В п.3 кратко описаны алгоритмы решения.

1. Постановки задач

Пусть $M = \{1, 2, \dots, m\}$ и $N = \{1, 2, \dots, n\}$ множество номеров фиксированных объектов на плоскости и множество номеров новых объектов, которые необходимо разместить на той же плоскости. Координаты фиксированного объекта P_i обозначим через (p_{1i}, p_{2i}) , $i \in M$, а координаты нового объекта $X_j - (x_j, y_j)$, $j \in N$. Удельные стоимости связей между i - м фиксированным и j - м новым объектами обозначим через w_{ij} , а через v_{jk} удельные стоимости связей между новыми объектами j и k , при этом считаем, что $v_{jk} = v_{kj}$, $k \neq j$. Кроме того, задано z прямоугольных запрещенных зон F_1, F_2, \dots, F_z , со сторонами параллельными осям координат (изотетичные прямоугольники). Через $Z = \{1, 2, \dots, z\}$ обозначим множество их номеров и через $F = \cup_{i=1}^z F_i$ объединение зон. Пусть $d(P_i, X_j)$ это расстояние между i - м фиксированным и j - м новым объектами, а $d(X_j, X_k)$ - это

¹Работа выполнена при поддержке гранта РГНФ (проект 04-02-00238а)

расстояние между j - м и k - м новыми объектами. Требуется разместить новые объекты X_1, X_2, \dots, X_n вне зон F_1, F_2, \dots, F_z так, чтобы максимальное взвешенное расстояние (удельная стоимость, умноженная на расстояние) либо суммарная стоимость между всеми объектами были минимальными. Первый критерий называется минимаксным, т.е.

$$\max \left\{ \max_{j,k \in N, j < k} v_{jk} \cdot d(X_j, X_k), \max_{i \in M, j \in N} w_{ij} \cdot d(P_i, X_j) \right\} \rightarrow \min_X, \quad (1)$$

а второй минисуммным, т.е.

$$\sum_{j,k \in N, j < k} v_{jk} \cdot d(X_j, X_k) + \sum_{i \in M} \sum_{j \in N} w_{ij} \cdot d(P_i, x_j) \rightarrow \min_X. \quad (2)$$

при ограничениях:

$$X_j \notin F, \quad j \in Z. \quad (3)$$

Задачу (1),(3) будем называть минимаксной, а (2),(3) - минисуммной. Если расстояния d измеряются в метрике L_1 , то задачи (1) и (2) можно переписать как задачи линейного программирования (ЛП).

Задачу (1) перепишем в эквивалентной формулировке с помощью введения дополнительного параметра x_0 :

$$x_0 \rightarrow \min \quad (4)$$

$$\left\{ \begin{array}{l} x_j + y_j + \frac{x_0}{w_{ij}} \geq -p_{1i} - p_{2i} \\ -x_j + y_j + \frac{x_0}{w_{ij}} \geq p_{1i} - p_{2i} \\ x_j - y_j + \frac{x_0}{w_{ij}} \geq p_{2i} - p_{1i} \\ -x_j - y_j + \frac{x_0}{w_{ij}} \geq -p_{1i} - p_{2i} \end{array} \right\} \quad i \in M, j \in N, \quad (5)$$

$$\left\{ \begin{array}{l} x_j + x_k + y_j + y_k + \frac{x_0}{v_{jk}} \geq 0 \\ -x_j - x_k + y_j + y_k + \frac{x_0}{v_{jk}} \geq 0 \\ x_j + x_k + y_j - y_k + \frac{x_0}{v_{jk}} \geq 0 \\ -x_j + x_k - y_j + y_k + \frac{x_0}{v_{jk}} \geq 0 \end{array} \right\} \quad j, k \in N, j < k, \quad (6)$$

Задача (2) записывается с помощью введения переменных $s = (s_{ij}), l = (l_{ij}), i \in M, j \in N, t = (t_{jk}), u = (u_{jk}), j, k \in N, j < k$, и сводится к задачам ЛП следующего вида:

$$\left. \begin{array}{l} \sum_{j=1}^{n-1} \sum_{k=j+1}^n v_{jk} \cdot t_{jk} + \sum_{i \in M} \sum_{j \in N} w_{ij} \cdot s_{ij} \rightarrow \min_{s,t} \\ -s_{ij} \leq x_j - p_{1i} \leq s_{ij}, \quad i \in M, j \in N \\ -t_{jk} \leq x_j - x_k \leq t_{jk}, \quad j, k \in N, j < k \end{array} \right\} \quad (7)$$

$$\left. \begin{array}{l} \sum_{j=1}^{n-1} \sum_{k=j+1}^n v_{jk} \cdot r_{jk} + \sum_{i \in M} \sum_{j \in N} w_{ij} \cdot l_{ij} \rightarrow \min_{l,r} \\ -l_{ij} \leq y_j - p_{2i} \leq l_{ij}, \quad i \in M, j \in N \\ -r_{jk} \leq y_j - y_k \leq r_{jk}, \quad j, k \in N, j < k \end{array} \right\} \quad (8)$$

С учетом специфики задач ЛП предложены более эффективные алгоритмы решения задач (1) и (2). В работе [4] выполняется переход от метрики L_1 к L_∞ . В этом случае задача (1) декомпозируется на две независимые подзадачи одинаковой структуры. Решение каждой из них находится с помощью алгоритма поиска специальных кратчайших цепей. В [5] также решается специальная задача ЛП. В работе [6] указанная задача сводится к поиску кратчайших путей в сети, длины дуг которой зависят от параметра.

Для минимизации функционала (2) необходимо найти поток минимальной стоимости в специальной сети [7]. В работе [8] задача сводится к решению серии задач поиска максимального потока. В работе [9] решаются задачи построения минимальных разрезов.

2. Построение моделей ЦЛП

Построим модели ЦЛП задач (1),(3) и (2),(3). Для этого запишем условия (3) в виде систем линейных неравенств с целочисленными переменными, так как область $\bar{F} = R^2 \setminus F$, не является выпуклой. Для этого множество \bar{F} представим в виде объединения прямоугольных областей - разрешенных зон R_1, R_2, \dots, R_g . Через $G = \{1, 2, \dots, g\}$ обозначим множество их номеров. Зона R_k ограничена прямоугольником $[(a_k, c_k); (b_k, d_k)]$, где (a_k, c_k) координаты левого нижнего угла, а (b_k, d_k) - правого верхнего. Для того чтобы осуществить указанное разбиение, необходимо построить контур (границу) области F .

Контур объединения прямоугольников находится с помощью метода плоского заметания, основанного на дереве отрезков [10]. Контур множества F состоит из набора непесекающихся циклов, образованных чередующимися вертикальными и горизонтальными ребрами. Любое ребро ориентировано так, что фигура расположена слева от него; т.е., цикл ориентирован по часовой стрелке, если он ограничивает "дыру", и против часовой стрелки, если он является внешней границей компоненты связности области F .

Алгоритм построения разрешенных зон. Ограничим область F прямоугольником со сторонами: $A = \min_{k \in Z} a_k$, $C = \min_{k \in Z} c_k$, $B = \max_{k \in Z} b_k$, $D = \max_{k \in Z} d_k$. Тогда можно выделить четыре неограниченные разрешенных зоны: $x_i \leq A$, $x_i \geq B$, $y_i \leq C$ и $y_i \geq D$.

Пусть контур F построен. Будем определять разрешенные зоны внутри прямоугольника $[(A, C), (B, D)]$ "двигаясь" вдоль оси Y . Упорядочим в порядке возрастания координаты по оси X вертикальных ребер: $A = x_1 < x_2 < \dots < x_t = B$. Строим список точек пересечения прямой $x = x_2$ с горизонтальными ребрами. Упорядочиваем его по возрастанию координаты Y : $(x_2, y_1), \dots, (x_2, y_s)$, где $y_1 = C$, $y_s = D$. Проверяем, принадлежит или нет точка (x_2, y_1) границе запрещенной зоны. Далее, перебирая по порядку все точки из списка, начиная с (x_2, y_2) , проверяем, принадлежит ли точка правой границе запрещенной зоны или правой границе разрешенной зоны. Если точка - это левый конец горизонтального ребра либо является его внутренней точкой, то она принадлежит правой границе разрешенной зоны. Если точка - правый конец горизонтального ребра, то она принадлежит правой границе запрещенной зоны. Признаком выхода из любой зоны служит точка, не являющаяся началом какого-либо горизонтального ребра, либо точка, являющаяся одновременно концом одного горизонтального отрезка и началом другого, либо точка (x_2, y_s) . Перебрав все точки из списка $(x_2, y_1), \dots, (x_2, y_s)$ и сформировав разрешенные зоны в полосе $[x_1, x_2]$, переходим к координате x_3 и повторяем процесс.

Алгоритм построения по шагам. Пусть k - счетчик числа разрешенных зон, i - номер очередной координаты по оси X , j - номер очередной координаты по оси Y . Введем

логическую переменную $flag$, причем $flag = true$, если мы "двигаемся" по правой границе некоторой разрешенной зоны, и $flag = false$, если мы "двигаемся" по правой границе запрещенной зоны.

Шаг1. Упорядочим список вертикальных ребер по возрастанию координаты X , а список горизонтальных ребер - по возрастанию координаты Y . Полагаем $i := 2, j := 1, k := 4$, (т.к. уже имеется четыре разрешенные зоны) и переходим на шаг 2.

Шаг2. Находим список точек пересечения прямой $x = x_i$ с горизонтальными ребрами и прямыми $y = C, y = D$. Упорядочиваем его по возрастанию координаты Y и переходим на шаг 3.

Шаг3. Если точка (x_i, C) не является началом какого-либо горизонтального ребра, но принадлежит ему, то полагаем $flag := false, j := 2$. Переходим на шаг 4.

Если точка (x_i, C) является началом какого-либо горизонтального ребра, либо не принадлежит никакому ребру, то $flag := true, k := k + 1, a_k := x_{i-1}, b_k := x_i, c_k := C, j := 2$. Переходим на шаг 4.

Шаг4.

4.1. Если ($y_j = D$ и $flag = false$), тогда если $x_i = B$ то СТОП. Иначе ($x_i \neq B$), полагаем $i := i + 1, j := 1$ и переходим на шаг 2.

4.2. Если ($y_j = D$ и $flag = true$) тогда $d_k := y_j$. Вносим зону $[(a_k, c_k), (b_k, d_k)]$ в список разрешенных зон. Если $x_i = B$, то СТОП, иначе ($x_i \neq B$) $i := i + 1, j := 1$. На шаг 2.

4.3. Если $y_j \neq D$ и (x_i, y_j) является началом и не является концом горизонтального ребра, то полагаем $j := j + 1$. Переходим на шаг 4.

4.4. Если $y_j \neq D$ и (x_i, y_j) является концом либо принадлежит горизонтальному ребру, то

4.4.1. Если $flag = true$ то полагаем $d_k := y_j, flag := false, j := j + 1$. На шаг 4.

4.4.2. Если $flag = false$, то полагаем $k := k + 1, a_k := x_{i-1}, b_k := x_i, c_k := y_j, flag := true, j := j + 1$. Переходим на шаг 4.

Для уменьшения количества разрешенных зон необходимо проверить возможность объединения некоторых из них в прямоугольники. Так как зоны определялись при "движении" вдоль оси Y , то, очевидно, что объединить, например, зоны R_k и R_j можно только в случаях: $b_k = a_j$ и $d_k = c_j$ или $b_j = a_k$ и $d_j = c_k$.

Запишем модели ЦЛП для задач (1),(3) и (2),(3). Не ограничивая общности, можно считать, что все фиксированные объекты и запрещенные зоны находятся в первой четверти. Иначе этого можно добиться параллельным сдвигом. Пусть $k_1 = \max_{i \in M, k \in G} \{p_{1i}, b_k\}$, $k_2 = \max_{i \in M, k \in G} \{p_{2i}, d_k\}$. Нетрудно показать, что оптимальное решение задач находится в прямоугольнике с координатами $[(0, 0), (k_1, k_2)]$. Условие принадлежности объекта i только одной из зон записывается с помощью булевой переменной $h_{ik}, i \in N, k \in G$, где $h_{ik} = 1$, если i -й объект находится k -й разрешенной зоне, иначе $h_{ik} = 0$. Тогда условия принадлежности каждого объекта только одной из разрешенных зон имеют вид:

$$\left. \begin{aligned} x_i - a_k + a_k(1 - h_{ik}) &\geq 0 \\ y_i - c_k + c_k(1 - h_{ik}) &\geq 0 \\ -x_i + k_1 - h_{ik}(k_1 - b_k) &\geq 0 \\ -y_i + k_2 - h_{ik}(k_2 - d_k) &\geq 0 \\ \sum_{k=1}^G h_{ki} &= 1 \\ h_{ik} &\in \{0, 1\} \end{aligned} \right\} i \in N, k \in G, \quad (9)$$

В результате условия (4)-(6),(9) образуют модель ЦЛП задачи (1),(3), а условия (7)-(9) модель ЦЛП задачи (2),(3).

3. Алгоритмы решения задач

В работе [1] описан алгоритм решения минимаксной задачи для случая $n = 1, g = 1$. Он легко распространяется на случай произвольного числа зон, так как в его основе лежит процедура нахождения оптимального размещения на ребрах контура F .

Для общего случая можно применить достаточно разработанный аппарат решения задач целочисленного программирования. Нижние оценки целевой функции в задаче (7)-(9) можно получить с использованием сетевых задач [9].

Список литературы

- [1] Г.Г.Забудский *Алгоритм решения минимаксной задачи размещения объекта на плоскости с запрещенными зонами.*-Автоматика и телемеханика, 2004,№4,с.93-100.
- [2] Г.Г.Забудский, И.В.Нежинский *Решение задач размещения в евклидовом пространстве.* – Вестник ОмГУ, 1999,Вып.2, с.17–19.
- [3] Г.Г.Забудский, М.В.Орлова *Решение минисуммной задачи размещения на плоскости с запрещенными областями.*-Матер.конф. "Дискретный анализ и исследование операций", Новосибирск, 2004,с.157.
- [4] Dearing P.M., Francis R.L. *A network flow solution to a multifacility minimax location problem involving rectilinear distances.* Transportation Science, 1974,Vol.8,№4, p.126-141.
- [5] Morris J.G. *A Linear Programming Approach to the Solution of Constrained Multi-Facility Minimax Location Problems where Distances are Rectangular.* Oper. Res. Quar., 1973, Vol. 24, p. 419-435.
- [6] Ichimori T. *A shortest path approach to a multifacility minimax location problem with rectilinear distances.* Journal of the Operations Research Society of Japan,1974, Vol.8, p. 126-141.
- [7] Cabot A.V., Francis R.L. *A network flow solution to rectilinear distances facility location problem.* AIIE Trans.,1970, Vol.2, N2, p. 132-141.
- [8] В.А.Трубин *Эффективный алгоритм для задачи Вебера с прямоугольной метрикой.*- Кибернетика, 1978, N6, с.67-70.
- [9] Picard J.C., Ratliff H.D. *A cut approach to the rectilinear distance facility location problem.* Operations Research, 1978, Vol.2,N 3, p. 422-433.
- [10] Препарата Ф., Шеймос М. *Вычислительная геометрия.* М.: Мир, 1989,478 с.

ABOUT MINIMAX AND MINISUM LOCATION PROBLEMS ON A PLANE WITH FORBIDDEN AREAS

G.G. Zabudsky

*Omsk Branch of Sobolev Institute of Mathematics, Omsk
e-mail: zabudsky@iitam.omsk.net.ru*

Abstract. Problems of locating new facilities with respect to existing facilities on a plane are considered. There are rectangular forbidden areas on the plane. The forbidden areas are isothetic rectangles, i.e., rectangles whose sides are parallel to coordinate axes. New facilities must be placed outside of the forbidden areas. Distances between all facilities are measured by the rectilinear metric. The objective is to minimize the maximum or minimize the sum of the weighted distances between existing and new facilities and between pairs of new facilities.

We present mixed-integer programming models for the problems. and give briefly survey of some algorithms for solving the problems.

Key words: location problem, integer programming, minimax, minisum, plane, forbidden area.

О ПРЕДСТАВЛЕНИИ ПЕРЕСТАНОВКАМИ ДОПУСТИМЫХ РЕШЕНИЙ ОДНОМЕРНОЙ ЗАДАЧИ УПАКОВКИ В КОНТЕЙНЕРЫ ¹

В.В. Залюбовский

*Институт математики им. С.Л. Соболева, Новосибирск
e-mail: slava@math.nsc.ru*

Аннотация. Предлагается метод представления допустимых решений одномерной задачи упаковки в контейнеры в виде перестановок специального вида. Вводится понятие *NF-активных* упаковок и показывается, что поиск оптимального решения можно вести только в этом классе. Доказано, что каждой перестановке соответствует ровно одна *NF-активная* упаковка, а при дополнительных условиях на структуру перестановки удается добиться также однозначности обратного соответствия. Кроме того, описан класс максимальных упаковок, использование которого позволяет сократить просматриваемое подмножество допустимых решений в процессе поиска оптимума.

Ключевые слова: задача упаковки, пространство решений, метод ветвей и границ,

Введение

Одномерная задача упаковки в контейнеры (ЗУ) может быть сформулирована следующим образом: для заданного множества L из n предметов с размерами (весеами) a_1, \dots, a_n и идентичных контейнеров вместимости C необходимо определить, какое количество контейнеров требуется для упаковки всех предметов. Это одна из классических задач комбинаторной оптимизации. Она является NP-трудной в сильном смысле [1]. Большинство публикаций, посвященных ЗУ, касаются приближенных алгоритмов и их поведения в среднем или наихудшем случае. Эти алгоритмы обычно основаны на очень простых правилах и зачастую обеспечивают удовлетворительные результаты [2]. Результаты, касающиеся асимптотически точного подхода к ЗУ, приведены в [3].

В отличие от приближенных, число известных точных алгоритмов для ЗУ невелико. Первый из них был предложен в [4] и представлял собой простой метод ветвей и границ (МВГ) с поиском в глубину и стратегией ветвления, выбирающей предметы в порядке невозрастания их весов. Для оценки значений целевой функции на частичных решениях используется простейшая нижняя оценка, получающаяся в предположении о делимости предметов. Наиболее известным на сегодняшний день является алгоритм МТР [5], в котором были введены улучшенные нижние оценки, а также процедура редукции доминируемых решений. Алгоритм BISON [6] представляет собой гибрид метода ветвей и границ и алгоритма локального поиска, использующий целый ряд процедур вычисления нижних оценок, редукции и эвристики. В работе [7] описан точный алгоритм, в основе которого лежит комбинация методов ветвей и границ и генерации столбцов.

В настоящей работе предлагается способ представления множества допустимых решений в виде перестановок специального вида, который имеет определенные преимущества в сравнении с известными ранее и может быть успешно использован в методе ветвей и границ для решения ЗУ.

¹Работа выполнена при поддержке РФФИ (проект 05-01-00395)

1. Представление допустимых решений в виде перестановок

Реализация метода ветвей и границ зависит от представления (кодировки) решения. Выбор кодировки во многом определяет качество алгоритма. В процессе комбинаторного поиска множество кодов задает пространство решений, и исходная задача нахождения оптимального решения сводится к поиску наилучшего элемента (кода) в этом пространстве. Для того, чтобы этот поиск был эффективным, пространство решений должно удовлетворять следующим минимальным требованиям [8].

1. Пространство решений конечно.
2. Каждое решение допустимо.
3. Решение задачи может быть получено по его коду за полиномиальное время.
4. Наилучший элемент в пространстве решений соответствует оптимальному решению исходной задачи.

Пространство решений, удовлетворяющее перечисленным требованиям, называется *P-допустимым*.

Заметим, что традиционное представление решения ЗУ в терминах булевых переменных x_{ij} , где $x_{ij} = 1$, если предмет j помещается в контейнер i , и $x_{ij} = 0$ в противном случае, не является *P-допустимым*, поскольку произвольный набор $x_{ij} \in \{0, 1\}$ не гарантирует допустимости решения.

Далее мы опишем схему кодирования решений ЗУ, основанную на перестановках, и докажем, что определяемое ей пространство решений является *P-допустимым*.

Поскольку предлагаемая схема в качестве декодирующей процедуры использует алгоритм Next-fit (NF), приведем его описание.

Алгоритм Next-fit. [2] Первый предмет помещается в первый контейнер. Каждый последующий предмет помещается в тот же контейнер, что и предыдущий, до тех пор, пока в нем достаточно свободного места. В противном случае предмет помещается в новый контейнер.

Очевидно, алгоритм NF имеет линейную сложность. Нетрудно показать, что для любого множества L выполнено неравенство $NF(L) \leq 2 \cdot OPT(L) - 1$, где $NF(L)$ и $OPT(L)$ — число контейнеров, используемых для упаковки множества L соответственно алгоритмом *NF* и оптимальным алгоритмом.

От перестановки к упаковке.

Для заданного множества предметов L и перестановки $\pi = (\pi_1, \pi_2, \dots, \pi_n)$ целых чисел $1, 2, \dots, n$, обозначим $\mathcal{P}_{NF}(L, \pi)$ результат применения алгоритма *NF* к последовательности $L_\pi = (a_{\pi_1}, a_{\pi_2}, \dots, a_{\pi_n})$. Перестановку π при этом будем называть *кодом* упаковки $\mathcal{P}_{NF}(L, \pi)$.

Пример 1. Рассмотрим множество L из 5 предметов: $a_1 = 5, a_2 = 4, a_3 = 3, a_4 = 2, a_5 = 1$. Пусть вместимость контейнера C равна 10 и $\pi = (1, 5, 2, 4, 3)$. Тогда $\mathcal{P}_{NF}(L, \pi) = (B_1, B_2)$, где $B_1 = [a_1, a_5, a_2]$ и $B_2 = [a_4, a_3]$.

Заметим, что произвольной перестановке π соответствует некоторая допустимая упаковка $\mathcal{P}_{NF}(L, \pi)$. Поскольку число перестановок конечно, и алгоритм *NF* имеет линейную

сложность, можно заключить, что множество перестановок определяет пространство решений ЗУ, удовлетворяющее свойствам 1–3 из определения P -допустимого пространства.

Для проверки свойства 4 необходимо описать процедуру получения кода для заданной упаковки.

От упаковки к перестановке.

Пусть $\mathcal{P} = (B_1, B_2, \dots, B_m)$ — допустимая упаковка и $L_i = (b_1^i, b_2^i, \dots, b_{k_i}^i)$ — произвольным образом упорядоченные множества предметов, помещенных в контейнер B_i . Определим последовательность $L_{\mathcal{P}}$ как конкатенацию L_i : $L_{\mathcal{P}} = (b_1^1, \dots, b_{k_1}^1, b_1^2, \dots, b_{k_2}^2, \dots, b_1^m, \dots, b_{k_m}^m)$. Тогда для построения перестановочного кода по заданной упаковке мы просто перечисляем индексы предметов в порядке, определяемом $L_{\mathcal{P}}$. Перестановку, полученную в результате этого процесса, обозначим $\pi(\mathcal{P})$.

Пример 2. Рассмотрим множество предметов L из примера 1 и упаковку $\mathcal{P} = (B_1, B_2, B_3)$, где $B_1 = [a_1, a_5]$, $B_2 = [a_2, a_4]$ и $B_3 = [a_3]$. Тогда $L_{\mathcal{P}} = (a_1, a_5, a_2, a_4, a_3)$ и $\pi(\mathcal{P}) = (1, 5, 2, 4, 3)$.

Теперь мы можем рассмотреть упаковку $\mathcal{P}_{NF}(L, \pi(\mathcal{P}))$ и сравнить ее с исходной упаковкой \mathcal{P} .

Прежде всего заметим, что в общем случае $\mathcal{P}_{NF}(L, \pi(\mathcal{P}))$ может не совпадать с \mathcal{P} . Действительно, для примера 2 имеем $\mathcal{P}_{NF}(L, \pi(\mathcal{P})) = ([1, 5, 2], [4, 3])$, в то время как $\mathcal{P} = ([1, 5], [2, 4], [3])$.

Определение 1. Допустимую упаковку \mathcal{P} множества предметов L будем называть NF -активной если $\mathcal{P}_{NF}(L, \pi(\mathcal{P}))$ совпадает с \mathcal{P} .

Пусть $\mathcal{P} = (B_1, B_2, \dots, B_m)$ — допустимая упаковка множества предметов L . Обозначим число используемых этой упаковкой контейнеров $|\mathcal{P}|$. Для $a_i \in L$ положим $Ind(\mathcal{P}, a_i) = j$, если в упаковке \mathcal{P} предмет a_i помещен в контейнер B_j .

Лемма 1. Для любой допустимой упаковки \mathcal{P} множества предметов L справедливо неравенство

$$|\mathcal{P}_{NF}(L, \pi(\mathcal{P}))| \leq |\mathcal{P}| .$$

Лемма 2. Для любой допустимой упаковки \mathcal{P} множества предметов L существует NF -активная упаковка \mathcal{P}' такая что

$$|\mathcal{P}'| \leq |\mathcal{P}| .$$

Для доказательства леммы 1 достаточно заметить, что для произвольного предмета $a_i \in L$ справедливо следующее неравенство:

$$Ind(\mathcal{P}_{NF}(L, \pi(\mathcal{P})), a_i) \leq Ind(\mathcal{P}, a_i) .$$

Лемма 2 непосредственно следует из леммы 1 и показывает, что при поиске оптимального решения ЗУ можно ограничиться рассмотрением лишь активных упаковок.

Установленные свойства приводят к следующему заключению.

Теорема 1. Множество всех перестановок определяет P -допустимое пространство решений ЗУ. Оно состоит из $n!$ перестановок, каждая из которых может быть трансформирована в допустимую упаковку за время $O(n)$, и по меньшей мере одна из

них соответствует оптимальному решению ЗУ.

2. Регулярные перестановки

Нетрудно заметить, что соответствие между множеством NF -активных упаковок и множеством перестановок не является взаимно-однозначным, поскольку одной упаковке может соответствовать несколько перестановок.

Пример 3. Рассмотрим множество предметов L из 5 элементов: $a_1 = 5, a_2 = 4, a_3 = 3, a_4 = 2, a_5 = 1$. Пусть вместимость контейнера C равна 10. Рассмотрим следующие перестановки: $\pi_1 = (1, 5, 2, 4, 3)$, $\pi_2 = (2, 1, 5, 3, 4)$ и $\pi_3 = (4, 3, 1, 5, 2)$. Очевидно, $\mathcal{P}_{NF}(L, \pi_1)$, $\mathcal{P}_{NF}(L, \pi_2)$ и $\mathcal{P}_{NF}(L, \pi_3)$ представляют собой одну и ту же упаковку.

Для устранения избыточности кодировки мы введем понятие регулярной перестановки. Для простоты изложения далее будем предполагать, что все предметы имеют различный вес. (В общем случае можно считать, что из двух предметов одинакового веса предмет с наименьшим индексом является более тяжелым).

Определение 2. Для заданного множества предметов $L = \{a_1, \dots, a_n\}$ и вместимости контейнера C перестановка $\pi = (\pi_1, \pi_2, \dots, \pi_n)$ называется *регулярной*, если существуют целые m , $1 \leq m \leq n$ и $0 = k_0 < k_1 < \dots < k_{m-1} < k_m = n$, для которых справедливы следующие условия:

$$\sum_{j=k_{p-1}+1}^{k_p} a_{\pi_j} \leq C \quad \text{для всех } 1 \leq p \leq m; \quad (1)$$

$$\sum_{j=k_{p-1}+1}^{k_p+1} a_{\pi_j} > C \quad \text{для всех } 1 \leq p \leq m-1; \quad (2)$$

$$a_{\pi_{k_{p-1}+1}} \geq a_{\pi_{k_p+1}} \quad \text{для всех } 1 \leq p \leq m-1; \quad (3)$$

$$a_{\pi_{k_{p-1}+i}} \geq a_{\pi_{k_p+j}} \quad \text{для всех } 1 < i < j \leq k_p - k_{p-1}; 1 \leq p \leq m. \quad (4)$$

Значения k_p определяют разбиение заданного множества предметов L на m подмножеств. Неравенства (1) отражают ограничение на вместимость контейнера. Ограничения (2) требуют, чтобы упаковка была активной. Соотношения (3) гарантируют упорядочение контейнеров по неубыванию размера первого (наибольшего) предмета. И, наконец, согласно условию (4) предметы внутри каждого контейнера должны быть упорядочены по убыванию размера.

Легко убедиться в том, что если для некоторой допустимой упаковки существует представление в виде регулярной перестановки, то оно единственно.

Кроме того, хотя нельзя гарантировать существование кодировки в виде регулярной перестановки для произвольной допустимой (и даже активной) упаковки, справедливо следующее утверждение.

Лемма 3. Для любой допустимой упаковки \mathcal{P} множества предметов L существует регулярная упаковка \mathcal{P}' такая что

$$|\mathcal{P}'| \leq |\mathcal{P}|.$$

Для доказательства леммы опишем процедуру построения по заданной упаковке \mathcal{P} регулярной упаковки. Для этого вначале упорядочим предметы в каждом контейнере по

невозрастанию их весов, а сами контейнеры — по невозрастанию первых (максимальных) предметов. Это даст нам некоторую перестановку π . Построим по ней упаковку $\mathcal{P}_{NF}(L, \pi)$. Если полученная упаковка оказалась не регулярной, то повторяем процедуру. Очевидно, после конечного числа шагов мы получим регулярную упаковку, использующую не более $|\mathcal{P}|$ контейнеров.

Теорема 2. *Множество всех регулярных перестановок определяет P -допустимое безызбыточное пространство решений ЗУ.*

3. Доминируемые решения и максимальные упаковки

Наряду с устранением избыточности представления допустимых решений, действенным способом ускорения поиска оптимального решения является исключение из рассмотрения решений, заведомо не являющихся оптимальными. В методе ветвей и границ для этого успешно используются процедуры нахождения нижних оценок, позволяющие отсекал области, не содержащие оптимум. Кроме того, этим же целям служат различные критерии доминированности [6].

Заметим, что алгоритм Next-Fit, используемый в качестве декодирующей процедуры, сам по себе исключает из рассмотрения некоторые заведомо доминируемые упаковки. Так, он гарантирует, что каждый контейнер (кроме, быть может, последнего) будет заполнен по меньшей мере наполовину. Но представление решений ЗУ в виде перестановок позволяет сформулировать более сильное правило редукции множества допустимых решений.

Пусть имеется некоторая NF -активная упаковка $\mathcal{P} = (B_1, \dots, B_m)$. Будем говорить, что упаковка \mathcal{P} является *максимальной*, если свободное пространство в любом контейнере B_i , $i = 1, \dots, m - 1$, не превышает размера минимального предмета, расположенного в контейнерах B_{i+1}, \dots, B_m . Другими словами, мы разрешаем начинать упаковку нового контейнера только в том случае, когда ни один из оставшихся предметов в текущий контейнер не помещается. Перестановку, соответствующую максимальной упаковке, будем также называть максимальной.

Теорема 3. *Множество всех максимальных перестановок определяет P -допустимое пространство решений ЗУ.*

В силу теорем 2 и 3, при реализации метода ветвей и границ для решения ЗУ можно ограничиться рассмотрением максимальных регулярных упаковок. Кроме того, концепция максимальных упаковок дает возможность осуществлять процесс ветвления на множестве допустимых решений на уровне контейнеров, а не отдельных предметов.

Заключение

Предложенный способ представления допустимых решений задачи упаковки в контейнеры в виде перестановок может быть использован в алгоритмах ветвей и границ и локального поиска. Описанные классы перестановок, гарантирующие безызбыточность и редукцию рассматриваемого допустимого множества, позволяют ускорить процесс поиска оптимального решения.

Особый интерес представляет вопрос о распространении подобного подхода к представлению множества решений более общих задач. В частности, двумерных задач

упаковки в полосу и прямоугольники, а также d -мерной задачи векторной упаковки.

Список литературы

- [1] M.R. Garey, D.S. Johnson. *Computers and Intractability: a guide to the theory of NP-completeness*. Freeman: San Francisco, 1979.
- [2] E.G. Coffmann, M.R. Garey, D.S. Johnson. *Approximation algorithms for bin packing: a survey*. In: *Approximation Algorithms for NP-hard Problems*. PWS Publishing, Boston, 1997, p. 46–93.
- [3] Э.Х. Гимади, В.В. Залюбовский. *Задача упаковки в контейнеры: асимптотически точный подход*. — Известия ВУЗов. Математика, 1997, N12, с. 25–33.
- [4] S. Eilon, N. Christofides. *The loading problem*. — *Management Science*, 1971, V. 17, p. 259–267.
- [5] S. Martello, P. Toth. *Knapsack Problems*. Chichester: Wiley, 1990.
- [6] A. Scholl A., R. Klein R., C. Jurgens. *BISON: A fast hybrid procedure for exactly solving the one-dimensional bin packing problem*. — *Computers and Operations Research*, 1997, V. 24, p. 627–645.
- [7] J.M. Valerio de Carvalho. *Exact solution of bin-packing problems using column generation and branch-and-bound*. — *Annals of Operations Research*, 1999, V. 86, p. 629–659.
- [8] H. Murata, K. Fujiyoshi, S. Nakatake, Y. Kajitani. *VLSI module placement based on rectangle-packing by the sequence-pair*. — *IEEE Trans. Computer-Aided Design*, 1996, V. 15, p. 1518–1524.

A PERMUTATION-BASED ENCODING SCHEME FOR THE BIN-PACKING PROBLEM

V.V. Zalyubovsky

Sobolev Institute of Mathematics, Novosibirsk

e-mail: slava@math.nsc.ru

Abstract. We present a new encoding scheme for the well-known one-dimensional bin packing problem. It uses a permutation of items for representation of feasible solution and linear-time Next-fit algorithm as decoding procedure. The scheme defines P-admissible solution space and seems promising for a branch-and-bound and local search algorithms.

Key words: bin packing, solution space, branch-and-bound

ЗАДАЧА ОПТИМАЛЬНОГО РАЗМЕЩЕНИЯ ЦЕНТРОВ ТЕЛЕКОММУНИКАЦИЙ В РЕГИОНЕ

Л.А. Заозерская¹, Е. Китриноу², А.А. Колоколов¹

¹Омский филиал Института математики СО РАН, Омск
e-mail: zaoyer@iitam.omsk.net.ru, kolo@iitam.omsk.net.ru

²Университет Эгейского моря, Чиос, Греция
e-mail: E.Kitrinou@soc.aegean.gr

Аннотация. Рассматривается задача оптимального размещения центров телекоммуникаций. Представлена ее математическая модель в виде двухкритериальной задачи поиска доминирующего множества в графе. Анализируются свойства модели и методы ее решения. Приводятся результаты вычислительного эксперимента для части островов Эгейского моря.

Ключевые слова: многокритериальная оптимизация, дискретная оптимизация, целочисленное программирование, задача о доминирующем множестве, перебор L -классов, центр телекоммуникаций

Введение

Важную роль в развитии экономики отдаленных районов и закреплении в них населения играет использование телекоммуникационных технологий. В связи с этим актуальной является задача оптимального размещения центров телекоммуникаций (ТК). Ранее для указанной задачи нами предложена однокритериальная модель, представляющая собой некоторое обобщение задачи поиска оптимального доминирующего множества в графе, и на ее основе проведены расчеты для одного из архипелагов Эгейского моря [6].

В данной работе указанные исследования продолжаются в направлении применения многокритериальной оптимизации. Предлагается и анализируется двухкритериальная задача оптимального размещения центров ТК, в которой минимизируется общая стоимость размещения центров и максимизируется уровень удовлетворения потребностей региона в услугах этих центров. Строится соответствующая модель целочисленного линейного программирования (ЦЛП), обсуждаются методы ее решения и приводятся результаты экспериментальных исследований.

1. Постановка задачи

Пусть P_1, \dots, P_n – множество пунктов потребления услуг центров ТК, которые являются одновременно и пунктами возможного размещения этих центров, а c_j – это стоимость размещения центра в P_j , $j = \overline{1, n}$. Известны расстояния $d_{ij} = d(P_i, P_j)$ между пунктами P_i, P_j , $i, j = \overline{1, n}$, $i \neq j$ (например, в евклидовой метрике). Пусть задана величина d , определяющая максимально возможное расстояние между пунктом обслуживания и центром ТК. Размещение центров ТК является допустимым, если для каждого пункта имеется возможность обслуживания по крайней мере одним из центров. Необходимо найти допустимое размещение центров ТК, при котором минимизируется общая стоимость создания центров и максимизируется их суммарная эффективность.

При определении эффективности размещения центров ТК рассматриваются m социально-экономических и технологических факторов. Обозначим через S_k^j , $k = \overline{1, m}$, значения этих факторов для пункта P_j , $j = \overline{1, n}$. К ним относятся, например, численность населения, величина валового продукта, число пользователей в окружающей области, оснащённость различными видами связи, число студентов, школьников и т.д. [6]. Пусть w_k^j , $k = \overline{1, m}$, $j = \overline{1, n}$ – весовые коэффициенты, выражающие важность k -го фактора для j -го пункта. Тогда показатели эффективности размещения центров ТК в пунктах P_j , $j = \overline{1, n}$, вычисляются следующим образом

$$u_j = \sum_{k=1}^m w_k^j S_k^j.$$

Значения w_k^j определяет лицо, принимающее решение, или группа экспертов.

На основе расстояний d_{ij} и числа d построим граф $G = (V, E)$ с множеством вершин $V = \{P_1, \dots, P_n\}$ и множеством ребер $E \subseteq \{(P_i, P_j) \mid i, j = \overline{1, n}; i \neq j\}$. Ребро (P_i, P_j) содержится в E , если $d_{ij} \leq d$. Пусть A – матрица смежности вершин графа G . Положим $a_{ii} = 1$, $i = \overline{1, n}$. Тогда исходную задачу можно рассматривать как задачу поиска доминирующего множества в графе с двумя критериями.

Введем булевы переменные

$$x_j = \begin{cases} 1, & \text{если центр ТК размещается в пункте } P_j, \\ 0, & \text{иначе,} \end{cases} \quad j = \overline{1, n}.$$

Для второго критерия заменим максимизацию на минимизацию и построим математическую модель в виде задачи ЦЛП

$$F_1(x) = \sum_{j=1}^n c_j x_j \rightarrow \min \quad (1)$$

$$F_2(x) = - \sum_{j=1}^n u_j x_j \rightarrow \min \quad (2)$$

при условиях

$$\sum_{j=1}^n a_{ij} x_j \geq 1, \quad i = \overline{1, n}, \quad (3)$$

$$x_j \in \{0, 1\}, \quad j = \overline{1, n}. \quad (4)$$

Далее будем предполагать, что $c_j > 0$, $u_j > 0$, $j = \overline{1, n}$.

2. Анализ модели

Рассмотрим многокритериальную задачу дискретной оптимизации

$$F(x) = (F_1(x), \dots, F_m(x)) \rightarrow \min$$

$$x \in X,$$

где X – некоторое конечное множество.

Обозначим через \tilde{X} множество Парето-оптимальных решений (ПМ), а через X^0 - полное множество альтернатив (ПМА) (см., например, [4]). ПМА определяется как подмножество $X^0 \subseteq \tilde{X}$ минимальной мощности, такое, что $F(X^0) = F(\tilde{X})$, где $F(X') = \{F(x) \mid x \in X'\}$.

Для многих задач дискретной оптимизации имеет место следующее свойство [4]: для любого X существуют такие значения параметров целевой вектор-функции $F(x)$, при которых выполняется равенство

$$X^0 = \tilde{X} = X. \quad (5)$$

Задача (1)-(4) удовлетворяет этому свойству. Например, легко видеть, что для целевых функций с коэффициентами $c_j = u_j = 2^{j-1}$, $j = \overline{1, n}$, любые допустимые решения являются Парето-оптимальными. Максимальная мощность ПМА для случая произвольных $c_j > 0$, $j = \overline{1, n}$, будет равна $2^n - 1$, а для случая $c_j = 1$, $j = \overline{1, n}$, она равна n .

В [1, 4] исследовались вопросы разрешимости многокритериальных дискретных задач с помощью алгоритма линейной свертки критериев (АЛС). Задача (1)-(4) относится к числу задач, неразрешимых с помощью АЛС. Примером задачи, для которой АЛС нельзя получить все элементы ПМ, является задача с целевыми функциями (1), (2) при $n \geq 3$, $c_j = 2^{j-1}$, $u_j = 3^{j-1}$, $j = \overline{1, n}$, с ограничениями (4) и

$$\sum_{j=1}^n x_j \geq 1. \quad (6)$$

Для нее любая ненулевая булева точка x является Парето-оптимальным решением. Вместе с тем точка $x' = (1, 0, \dots, 0, 1)$ не может быть получена как оптимальное решение задачи минимизации функции $F^\lambda = \sum_{j=1}^n (\lambda c_j - (1 - \lambda)u_j)x_j$ при ограничениях (4), (6) для любого $\lambda \in [0, 1]$.

3. Методы решения задачи

Ввиду того, что получение всех элементов ПМ или ПМА для (1)-(4) может оказаться достаточно сложным, в качестве оптимальных решений часто используются лексикографические минимумы задачи [2, 3].

Для вектор-функции $F(X) = (F_1(x), \dots, F_m(x))$ Парето-оптимальное решение x^* называется лексикографическим минимумом (ЛМ) задачи, если $F(x^*) = \text{lexmin}\{F(x) \mid x \in \tilde{X}\}$. Множество всех ЛМ задачи при различных порядках их компонент вектор-функции образуют лексикографическое множество.

Для задачи (1)-(4) вектор-функции $F(x) = (F_2, F_1)$ соответствует единственный ЛМ, а именно, решение $(1, 1, \dots, 1)$. С практической точки зрения более интересным является ЛМ, отвечающий вектор-функции $F(x) = (F_1, F_2)$. Рассмотрим способы его получения.

1) Алгоритм линейной свертки критериев (АЛС).

В этом случае от решения многокритериальной задачи переходят к однокритериальной задаче с целевой функцией, являющейся линейной сверткой критериев.

Проблема нахождения ЛМ любой многокритериальной задачи с числом критериев не меньше 2 и конечным множеством допустимых решений разрешима с помощью АЛС [2, 3]. Следуя [3], коэффициенты линейной свертки для задачи (1)-(4) можно определить таким образом:

$$\lambda_1 = \frac{\alpha + 1}{\alpha + 2}, \quad \lambda_2 = \frac{1}{\alpha + 2},$$

где $\alpha = \sum_j^n u_j - [\tilde{u}_0] + 1$. Здесь \tilde{u}_0 – оптимальное значение целевой функции задачи линейного программирования (ЛП):

$$\sum_{j=1}^n u_j x_j \rightarrow \min$$

при условиях (3) и $0 \leq x_j \leq 1, j = \overline{1, n}$.

2) Метод последовательной оптимизации.

При таком подходе от решения многокритериальной задачи переходят к решению последовательности однокритериальных задач. Для поиска ЛМ задачи (1)-(4) методом последовательной оптимизации опишем модификацию гибридного алгоритма, предложенного для задачи о покрытии множества [5]. Этот алгоритм базируется на алгоритме перебора L -классов и является конечным. Его основная идея состоит в просмотривании элементов L -разбиения той части релаксационного множества задачи, для которой значение целевой функции не превосходит $\rho - 1$ (ρ – текущий рекорд целевой функции). Очередной элемент L -разбиения находится путем решения последовательности задач ЛП.

Решение задачи (1)-(4) сводится к двум этапам. На первом этапе решается задача о покрытии (1), (3), (4) алгоритмом из [5]. Обозначим через x^* ее оптимальное решение и положим $c_0^* = F_1(x^*)$.

На втором этапе требуется решить задачу (2)-(4) с дополнительным ограничением

$$\sum_{j=1}^n c_j x_j = c_0^*. \quad (7)$$

Отметим, что задача (2), (4), (7) является одномерной задачей о рюкзаке, которая достаточно эффективно решается алгоритмом перебора L -классов, так как в этом случае решения задач ЛП получаются аналитически. Поэтому текущую допустимую точку задачи (2)-(4), (7) будем искать следующим образом.

Начиная с точки x^* , решаем алгоритмом перебора L -классов задачу (2), (4), (7). Обозначим через x' целочисленную точку, найденную алгоритмом. Если x' удовлетворяет ограничениям (3), то x' – допустимое решение задачи (2)-(4), (7), иначе осуществляется переход к очередному элементу L -разбиения релаксационного многогранника задачи (2), (4), (7).

Другим известным подходом к определению множества решений многокритериальной задачи является метод уступок. От двухкритериальной задачи (1)-(4) можно перейти к однокритериальной задаче \mathcal{P}_1 :

$$F_1(x) = \sum_{j=1}^n c_j x_j \rightarrow \min$$

при ограничениях (3), (4) и $\sum_{j=1}^n u_j x_j \geq u_0$

или к задаче \mathcal{P}_2

$$F_2(x) = \sum_{j=1}^n u_j x_j \rightarrow \max,$$

при ограничениях (3), (4) и $\sum_{j=1}^n c_j x_j \leq c_0$.

Варьируя значения параметра u_0 (соответственно c_0), можно получать множество допустимых решений, в том числе и Парето-оптимальные решения исходной задачи. Здесь u_0 – требуемый уровень эффективности размещения, а c_0 – объем средств, выделенных для создания центров ТК.

Заметим, что, если $c_j = 1, j = \overline{1, n}$, то оптимальное решение задачи \mathcal{P}_2 является Парето-оптимальным для (1)-(4). При произвольных c_j для определения Парето-оптимальных решений потребуется дополнительно решать задачу вида \mathcal{P}_1 с параметром u_0 , равным оптимальному значению целевой функции задачи \mathcal{P}_2 .

4. Результаты вычислительного эксперимента

Нами решалась задача размещения центров ТК для архипелага из 24 островов Эгейского моря, включающего острова Родос, Лесмос, Кос и др. Для расчетов использовались данные Греческой организации статистики (ESYE) и Греческой организации телекоммуникаций (OTE). Значение d изменялось в пределах от 50 до 150 км ($\min d_{ij} = 20$, $\max d_{ij} = 526$). В вычислительном эксперименте при формировании критерия эффективности размещения использовались данные об общем числе потребителей услуг центров ТК, числе студентов, школьников. Для значений c_j рассматривались два варианта: 1) все $c_j = 1$, что соответствует размещению типовых центров ТК; 2) c_j вычислялись как значения некоторой вогнутой функции от числа потребителей, $j = \overline{1, n}$. При решении задач был использован пакет программ, разработанный в лаборатории дискретной оптимизации ОФ ИМ СО РАН.

Проведенные исследования показали, что предложенная модель применима для поиска оптимального размещения центров телекоммуникаций.

Список литературы

- [1] В.А. Емеличев, М.К. Кравцов *О неразрешимости векторных задач дискретной оптимизации на системах подмножеств в классе алгоритмов линейной свертки критериев* – ДАН, Математика, 1994, Т. 334, N1, с. 9-11.
- [2] В.А. Емеличев, М.К. Кравцов, О.А. Янушкевич *Лексикографические оптимумы многокритериальной задачи дискретной оптимизации.* – Математические заметки, 1995, Т. 58, Вып. 3, с. 365-371.
- [3] В.В. Подиновский, В.М. Гаврилов *Оптимизация по последовательно применяемым критериям.* Москва: Советское радио, 1975, 192 с.
- [4] В.А. Перепелица *Многокритериальные задачи теории графов. Алгоритмический подход.* Киев: УМК ВЦ ЗГУ, 1989, 67 с.
- [5] A.V. Eremeev, A.A. Kolokolov, L.A. Zaozerskaya *A hybrid algorithm for set covering.* Proc. of International Workshop on Discrete Optimization Methods Design. Minsk, 2000, pp. 123-129.

- [6] E. Kitrinou, A.A. Kolokolov, L.A. Zaozerskaya *The location choice for telecenters in remote areas. The case of the Aegean islands*. Proc. of Second International Workshop on Discrete Optimization Methods in Production and Logistics. Omsk, 2004, pp. 61-65.

THE PROBLEM OF THE OPTIMAL LOCATION CHOICE FOR TELECENTERS IN REGION

L.A. Zaozerskaya¹, E. Kitrinou², A.A Kolokolov¹

¹ *Omsk Branch of Sobolev Institute of Mathematics SB RAS, Omsk*
e-mail: zaozer@iitam.omsk.net.ru, kolo@iitam.omsk.net.ru

² *University of the Aegean, Chios, Greece*
e-mail: E.Kitrinou@soc.aegean.gr

Abstract. We consider the problem of telecenters location in remote areas. A bicriterial optimization problem of finding a dominating set in graph is suggested. We investigate the properties of this model and method for solving it. The results of computer experiment for some Islands of the Aegean region are presented.

Key words: multiextremal optimization, discrete optimization, integer programming, dominating set problem, *L*-class enumeration, telecenter

ГИБРИДНЫЙ АЛГОРИТМ ВЕТВЕЙ И ГРАНИЦ С ЛОКАЛЬНЫМ ПОИСКОМ ДЛЯ СМЕШАНО-ЦЕЛОЧИСЛЕННОЙ ЗАДАЧИ РАЗМЕЩЕНИЯ С ОГРАНИЧЕНИЯМИ НА МОЩНОСТИ ПРОИЗВОДСТВА¹

Д.С. Иваненко

*Институт математики им. С. Л. Соболева СО РАН, Новосибирск
e-mail: ivanen@math.nsc.ru*

Аннотация. Рассматривается задача размещения с ограничениями на мощности производства в смешано-целочисленной постановке. Предлагается гибридный эвристический алгоритм решения задачи, сочетающий идеи методов локального поиска и ветвей и границ. Алгоритм осуществляет локальный поиск в пространстве целочисленных переменных. Окрестность текущего решения просматривается при помощи метода ветвей и границ. Приводятся результаты численных экспериментов на трудных в вычислительном отношении примерах.

Ключевые слова: Задача размещения, локальный поиск, метод ветвей и границ, гибридные алгоритмы.

Введение

Задача размещения с ограничениями на мощности производства является обобщением простейшей задачи размещения [1]. Предполагаются заданными два множества: множество потребителей и множество потенциальных пунктов размещения производства (предприятий). Для каждого предприятия задано максимально возможное количество производимой продукции. Требуется удовлетворить запросы всех потребителей с минимальными суммарными затратами, которые складываются из затрат на организацию производства и транспортно-эксплуатационных затрат.

В зависимости от особенностей обслуживания потребителей различают несколько постановок задачи. В данной работе рассматривается задача, в которой потребители могут одновременно обслуживаться несколькими предприятиями. При этом количество продукции, необходимое потребителю зависит от того, какими именно предприятиями он обслуживается. В такой постановке переменные задачи естественным образом разбиваются на две группы: непрерывные и 0-1 переменные. При фиксированных значениях 0-1 переменных получаем задачу транспортного типа. Модели такого рода часто возникают при решении различных задач стандартизации [1].

Рассматриваемая задача является *NP*-трудной в сильном смысле, что является препятствием для построения малотрудоёмких точных алгоритмов. В последние годы наблюдается рост интереса к построению гибридных алгоритмов, сочетающих в себе преимущества точных и приближенных методов. В данной работе предлагается приближенный гибридный алгоритм. Он реализует идею локальных ветвлений, сочетая в себе алгоритм локального поиска и метод ветвей и границ [2]. Алгоритм осуществляет локальный поиск в пространстве 0-1 переменных. Для просмотра окрестности используется стандартный метод ветвей и границ с оценкой линейного программирования.

¹Работа выполнена при поддержке РФФИ (проект 03-01-00455)

1. Постановка задачи

Заданы множество потребителей $J = \{1, \dots, n\}$ и множество предприятий $I = \{1, \dots, m\}$. Для каждого предприятия $i \in I$ задана его область обслуживания $J(i) \subseteq J$, т.е. подмножество потребителей, которые могут получать его продукцию. Множества $J(i)$ могут пересекаться и в объединении дают все множество J . Каждое предприятие $i \in I$ характеризуется максимально возможным объемом производства $V_i \geq 0$ и величиной затрат на открытие предприятия $f_i \geq 0$, которая не зависит от количества производимой продукции.

Считаются также заданными объемы $p_{ij} \geq 0$, $i \in I$, $j \in J$ продукции i -го предприятия, необходимые j -му потребителю и эксплуатационные затраты $c_{ij} \geq 0$, $i \in I$, $j \in J$. Если $j \notin J(i)$ для некоторого $i \in I$, то $p_{ij} = c_{ij} = +\infty$.

Требуется обслужить всех потребителей с минимальными суммарными затратами. Суммарные затраты на обслуживание складываются из затрат на открытие предприятий f_i , $i \in I$ и эксплуатационных затрат c_{ij} , $i \in I$, $j \in J$.

Переменные задачи (x, y) имеют следующий смысл:

$x_i = 1$, если принимается решение об открытии i -го предприятия и 0 — в противном случае;

$0 \leq y_{ij} \leq 1$ — доля запросов j -го потребителя, обслуживаемая i -м предприятием.

С использованием введенных обозначений задача может быть записана следующим образом: найти

$$\min_{(x,y)} \left\{ \sum_{i \in I} f_i x_i + \sum_{i \in I} \sum_{j \in J} c_{ij} y_{ij} \right\} \quad (1)$$

при ограничениях

$$\sum_{i \in I} y_{ij} = 1, \quad j \in J, \quad (2)$$

$$\sum_{j \in J} p_{ij} y_{ij} \leq V_i x_i, \quad i \in I, \quad (3)$$

$$0 \leq y_{ij} \leq x_i, \quad i \in I, j \in J, \quad (4)$$

$$x_i \in \{0, 1\}, \quad i \in I, j \in J. \quad (5)$$

Целевая функция (1) выражает суммарные затраты на обслуживание потребителей. Равенства (2) гарантируют, что все потребители будут обслужены. Ограничения (3) запрещают предприятиям обслуживать потребителей, если их мощности исчерпаны. Неравенства (4) не позволяют использовать предприятия, которые не были открыты.

2. Схема алгоритма

Предлагаемый гибридный алгоритм основан на идее локальных ветвлений [3]. Он представляет собой итеративный алгоритм, сочетающий в себе метод локального поиска и алгоритм ветвей и границ. Алгоритм начинает работу с некоторого допустимого решения задачи. Такое решение легко получить, используя оптимальное решение (\tilde{x}, \tilde{y}) линейной релаксации исходной задачи. Для этого достаточно произвести разумное округление дробных компонент \tilde{x}_i решения (\tilde{x}, \tilde{y}) . Если в линейной релаксации нет допустимых решений, то исходная задача также несовместна.

На каждой итерации алгоритма рассматривается сужение исходной задачи на некоторую окрестность текущего решения. Полученная в результате такого сужения вспомогательная задача решается точно при помощи метода ветвей и границ. Оптимальное решение вспомогательной задачи сравнивается с рекордным и, если новое решение имеет лучшее значение целевой функции, оно заменяет собой рекордное решение. После этого все просмотренные решения исключаются из допустимого множества задачи, и алгоритм переходит к следующей итерации. Если же на некоторой итерации не было найдено решения, улучшающего рекорд, алгоритм переходит к проверке решений, которые не были просмотрены ранее.

Обозначим задачу (1) – (5) через Π . Область ее допустимых решений обозначим за P . Рассмотрим решение $(\bar{x}, \bar{y}) \in P$. Определим окрестность $N(\bar{x}, k)$ решения (\bar{x}, \bar{y}) следующим образом:

$$N(\bar{x}, k) = \{(x, y) \mid \Delta(\bar{x}, x) \leq k\},$$

где $\Delta(\bar{x}, x)$ – расстояние Хэмминга между \bar{x} и x . Пусть $S(\bar{x}) = \{i \in I \mid \bar{x}_i = 1\}$. Решение (x, y) принадлежит окрестности $N(\bar{x}, k)$ решения (\bar{x}, \bar{y}) если, и только если, оно удовлетворяет следующему линейному ограничению:

$$\sum_{i \in S(\bar{x})} (1 - x_i) + \sum_{i \in I \setminus S(\bar{x})} x_i \leq k, \quad (6)$$

и не принадлежит ей, если оно удовлетворяет ограничению:

$$\sum_{i \in S(\bar{x})} (1 - x_i) + \sum_{i \in I \setminus S(\bar{x})} x_i \geq k + 1. \quad (7)$$

Таким образом, добавляя к задаче Π ограничение (6), получим вспомогательную задачу $\Pi \downarrow \bar{x}$, которая является сужением задачи Π на множество $P \cap N(\bar{x}, k)$. Аналогично, добавление к задаче Π ограничения (7) приведет к задаче $\Pi \uparrow \bar{x}$, допустимым множеством которой будет множество $P \setminus N(\bar{x}, k)$.

Предположим, что имеется "черный ящик", на вход которого подаются данные задачи, а на выходе мы получаем её оптимальное решение или информацию о неразрешимости задачи. В роли такого "черного ящика" может выступать процедура, реализующая метод ветвей и границ для задачи смешано-целочисленного линейного программирования произвольного вида. Для получения нижних границ при этом могут использоваться оптимальные решения линейных релаксаций возникающих при ветвлении подзадач. Схема гибридного алгоритма, использующего такой "черный ящик" представлена на рисунке 1.

На первом шаге рассматривается вспомогательная задача $\Pi \downarrow x^1$, в процессе решения которой при помощи "черного ящика" (на рисунке изображен треугольником) будет найдено оптимальное решение $(x^2, y^2) \in P \cap N(\bar{x}, k)$. После этого данная ветвь отсекается и алгоритм переходит к задаче $\Pi \uparrow x^1$, запрещая тем самым уже просмотренные решения. На втором шаге алгоритма рассматривается новая подзадача $(\Pi \uparrow x^1) \downarrow x^2$ минимизации функции (1) на множестве $(P \setminus N(x^1, k)) \cap N(x^2, k)$. Если при этом улучшающего решения не обнаружено, то решение (x^2, y^2) является локальным оптимумом. Далее эта ветвь отсекается и алгоритм переходит к задаче $(\Pi \uparrow x^1) \uparrow x^2$, решением которой будет точка $(x^3, y^3) \in P \setminus (N(x^1, k) \cup N(x^2, k))$. Наилучшее из решений (x^2, y^2) и (x^3, y^3) является оптимальным решением задачи Π .

Легко видеть, что описанный выше алгоритм является точным. Его трудоемкость, как и трудоемкость метода ветвей и границ, в худшем случае не удастся ограничить полиномом от длины записи задачи. Тем не менее, описанный метод решения может быть легко

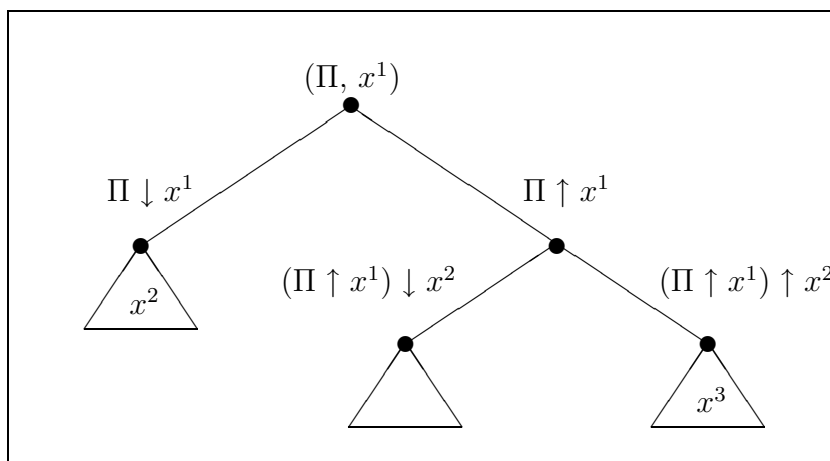


Рис. 1: Схема алгоритма локальных ветвлений

модифицирован для получения приближенного решения задачи за относительно небольшое время в тех случаях, когда метод ветвей и границ требует значительных временных затрат.

Рассмотрим модифицированную версию алгоритма, в которой задано упорядоченное семейство окрестностей

$$\aleph = \{N(\cdot, k_1), N(\cdot, k_2), \dots, N(\cdot, k_l)\}$$

для $k_1 < k_2 < \dots < k_l$. Алгоритм начинает работу с окрестностью $N(\cdot, k_1)$ и находит локальный оптимум. Далее выбирается окрестность $N(\cdot, k_2)$ и алгоритм переходит к следующей итерации. Если найденное на предыдущем шаге решение является локальным оптимумом относительно новой окрестности, то выбирается следующая по порядку окрестность в семействе \aleph и так далее. Алгоритм останавливается, если найден локальный оптимум для окрестности $N(\cdot, k_l)$. Если же улучшающее решение было найдено в некоторой окрестности $N(\cdot, k_s)$, $1 < s \leq l$, то алгоритм возвращается к поиску по окрестности $N(\cdot, k_1)$, используя новое рекордное решение в качестве начальной точки.

3. Результаты экспериментов

Численные эксперименты проводились на двух классах входных данных. Первый класс содержит тестовые примеры из библиотеки тестовых задач "Дискретные задачи размещения" (<http://math.nsc.ru/AP/benchmarks/index.html>). Второй класс состоял из примеров, для решения которых методу ветвей и границ требовалось существенное время. Примеры из второго класса имеют одинаковые мощности предприятий $V_i = 10$ и равные стоимости $f_i = 1000$. Область обслуживания каждого предприятия состоит ровно из 10-и элементов, выбираемых случайным образом из множества потребителей. Элементы матрицы c_{ij} равномерно распределены на отрезке $[1, 100]$, а величины p_{ij} выбирались равновероятно из множества $\{1, 2, 3, 4, 5\}$. Расчеты проводились на PC Intel Celeron D 325 с тактовой частотой 2,8 МГц. Набор окрестностей: $\aleph = \{N(\cdot, 1), N(\cdot, 2), N(\cdot, 3)\}$. В качестве "черного ящика" использован свободно распространяемый пакет GLPK [4]. Его версия для Microsoft Windows доступна по адресу <http://gnuwin32.sourceforge.net/packages/glpk.htm>

Первый класс тестовых примеров в вычислительном плане является легким для метода ветвей и границ. Средняя величина относительного разрыва двойственности составляет

менее 1%. Методу ветвей и границ требуется, в среднем, 4,5 минуты, чтобы отыскать оптимальное решение задачи. Средние временные затраты гибридного алгоритма составили ~ 8 минут. В 5-и случаях из 20-и было найдено оптимальное решение задачи. Средняя относительная погрешность алгоритма на этом классе составила менее 0,5%.

Результаты расчетов на примерах из второго класса приведены в Таблице 1. Таблица содержит информацию об относительном разрыве двойственности (δ), времени работы алгоритмов ветвей и границ (t_{ex}) и гибридного алгоритма с локальным поиском (t_{heu}), а также относительную погрешность полученных локальных оптимумов (γ). Из Таблицы 1 видно, что гибридный алгоритм получает решения с малой относительной погрешностью. Время его работы существенно меньше, чем у метода ветвей и границ.

№	δ (%)	t_{ex} (чч:мм:сс)	t_{heu} (чч:мм:сс)	γ (%)
1	3,7009995	4:15:17	0:37:42	0,4208992
2	3,8154847	4:01:06	0:18:25	0,6462926
3	3,7088593	2:04:18	0:08:54	2,3140223
4	3,6968552	2:59:00	0:37:43	3,244732
5	2,9588782	1:28:56	0:21:07	1,7048029
6	3,2957268	0:50:18	0:10:05	1,098596
7	3,9215238	3:21:15	0:38:23	1,1517562
8	3,9674716	4:52:25	0:24:24	3,2382784
9	3,8997123	7:28:47	0:20:52	4,0311489
10	3,5855149	1:46:21	0:18:56	2,7903347

Таблица 1: Результаты расчетов на втором классе тестовых примеров

Заключение

В работе предложен гибридный алгоритм решения задачи размещения с ограничениями на мощности производства. Алгоритм сочетает в себе методы локального поиска и ветвей и границ. Результаты экспериментов показывают, что предложенный метод решения разумно применять в тех случаях, когда стандартный метод ветвей и границ требует неприемлемо большого времени. Предложенный подход является достаточно гибким и может быть легко адаптирован, например, в схему локального поиска с чередующимися окрестностями [2].

Список литературы

- [1] Береснев В. Л., Гимади Э. Х., Дементьев В. Т. Экстремальные задачи стандартизации. Новосибирск: Наука, 1978.
- [2] Кочетов Ю., Младенович Н., Хансен П. Локальный поиск с чередующимися окрестностями - Дискретный анализ и исследование операций, Серия 2, 10(1), 2003, 11-43.
- [3] Fischetti M., Lodi A. Local branching. - Math. Program., Ser. B, 98, p. 23–47.

[4] The GLPK home page. (<http://www.gnu.org/software/glpk/glpk.html>)

A HYBRID BRANCH-AND-BOUND HEURISTIC WITH LOCAL SEARCH FOR THE MULTI-SOURCE CAPACITATED FACILITY LOCATION PROBLEM

Dmitry S. Ivanenko

Sobolev Institute of Mathematics, Novosibirsk, Russia
e-mail: ivanen@math.nsc.ru

Abstract. We propose a hybrid heuristic algorithm for the multi-source Capacitated Facility Location Problem. The heuristic mixes ideas of branch-and-bound and local search. It performs a local search through the 0-1 variables space, exploring neighbors with the branch-and-bound. Numerical results on hard benchmarks are discussed.

Key words: capacitated facility location, local search, branch-and-bound, hybrid heuristics

ШИРИНА РЕШЕТОЧНЫХ ГРАФОВ СПЕЦИАЛЬНОЙ СТРУКТУРЫ

С.Д. Иванова

Омский государственный университет, Омск
e-mail: IvanovaSD@yandex.ru

Аннотация. В задаче о ширине графа требуется найти нумерацию вершин неориентированного графа, которая минимизирует максимальную разность номеров смежных вершин. Эта задача является NP -трудной, в том числе и для решеточных графов. В данной статье найдена ширина прямоугольных решеток с двумя прямоугольными угловыми выемками и предложен способ построения оптимальной нумерации за полиномиальное время.

Ключевые слова: оптимизация, ширина графа, нумерация, решеточный граф.

Введение

Пусть $G = (V(G), E(G))$ – неориентированный граф, $N = |V(G)|$. *Нумерацией* графа G называется биекция $\varphi : V(G) \rightarrow \{1, 2, \dots, N\}$. Множество всех нумераций графа G обозначим через $\Phi(G)$. *Длина ребра uv при нумерации φ* равна $|\varphi(u) - \varphi(v)|$. Наибольшая длина ребра называется *шириной нумерации φ* и обозначается $B(G, \varphi)$.

Задача о ширине графа состоит в отыскании нумерации φ^* , для которой значение $B(G, \varphi^*)$ минимально. Величина $B(G) = \min\{B(G, \varphi) : \varphi \in \Phi(G)\}$ называется *шириной графа G* . Нумерация, ширина которой равна ширине графа, называется *оптимальной*.

Задача о ширине графа является NP -трудной [1], в том числе и для решеточных графов [2]. Граф G называется *решеточным*, если $V(G) \subset \mathbb{Z}^2$ и пара вершин смежна тогда и только тогда, когда евклидово расстояние между ними равно 1.

Известно, что задача о ширине графа полиномиально разрешима для прямоугольных решеток [3]. Нами изучается подкласс решеточных графов, которые могут быть представлены в виде прямоугольных решеток с прямоугольными угловыми выемками. В работе [4] рассматривался случай, когда решетка имеет одну угловую выемку, а также один частный случай решеток с двумя угловыми выемками. В данной статье исследуется общий случай прямоугольных решеток с двумя прямоугольными угловыми выемками. Найдена ширина этих графов и указан способ построения оптимальной нумерации за полиномиальное время.

1. Определения и предварительные сведения

В работе будут использованы следующие обозначения: $[z_1, z_2] = \{z_1, z_1 + 1, \dots, z_2\}$, где $z_1, z_2 \in \mathbb{Z}$, $z_1 \leq z_2$; $G_{m,n}$ – решеточный граф с множеством вершин $V(G_{m,n}) = [1, m] \times [1, n]$, $\mathcal{G}_{m,n} = \{G : V(G) = [1 + \alpha_1, m + \alpha_2] \times [1 + \alpha_1, n + \alpha_2], \alpha_1, \alpha_2 \in \mathbb{Z}, m, n \geq 1\}$.

Прямоугольная решетка – это решеточный граф из класса $\mathcal{G}_{m,n}$ для некоторых $m, n \geq 1$. Имеет место следующая

Теорема 1. [3] Пусть $m \geq n$. Если $m \geq 2$, то $B(G_{m,n}) = n$.

Пусть G – решеточный граф и G' – минимальная по включению прямоугольная решетка, содержащая G . Не ограничивая общности, будем считать, что $G' = G_{m,n}$ для неко-

торых m, n . Множество $\{(i, j) \in V(G) : i + j = k + 1\}$ называется k -й диагональю графа G , $k \in [1, m + n - 1]$.

Мы используем обозначение φ_d для нижней диагональной нумерации, которая строится следующим образом: вершины нумеруются по диагоналям, каждая диагональ пробегается сверху вниз. Таким образом, $\varphi_d(v) < \varphi_d(w)$, если и только если $v_1 + v_2 \leq w_1 + w_2$, причем в случае равенства $v_1 < w_1$, где $v = (v_1, v_2)$, $w = (w_1, w_2)$. Аналогично определяется верхняя диагональная нумерация: $\varphi_u(v) < \varphi_u(w)$, если и только если $v_1 + v_2 \leq w_1 + w_2$, причем в случае равенства $v_1 > w_1$. Пример нижней диагональной нумерации приведен на рис. 1.

Легко видеть, что $B(G_{m,n}, \varphi_d) = B(G_{m,n}) = n$ при $n \leq m$ и $B(G_{m,n}, \varphi_u) = B(G_{m,n}) = m$ при $n \geq m$.

Теорема 1 позволяет определить ширину прямоугольной решетки аналитически. В [4] аналогичный результат был получен для решетки с одной прямоугольной угловой выемкой. В работе [4] рассматривался решеточный граф L с множеством вершин $V(L) = V(G_{m,n}) \setminus V_1$, где $V_1 = [1, a_1] \times [n - b_1 + 1, n]$, $m, n \geq 3$, $a_1 \in [1, m - 2]$, $b_1 \in [1, n - 2]$. Такой граф изображен на рис. 1.

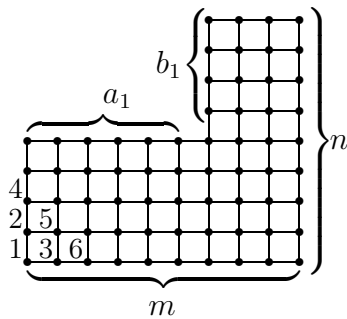


Рис. 1

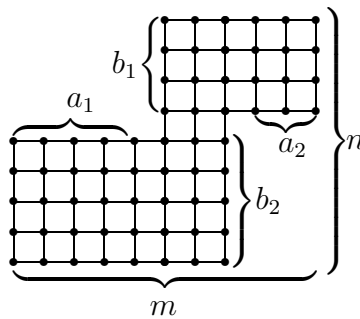


Рис. 2

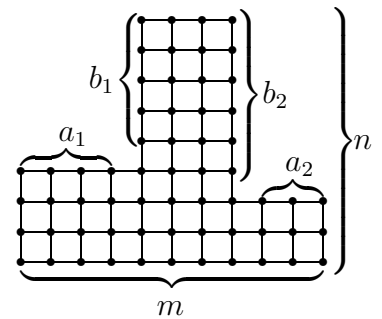


Рис. 3

Введем обозначения: $m_1 = m - a_1$, $n_1 = n - b_1$. Нетрудно показать, что $B(L) = B(L, \varphi_d) = n$ при $m_1 \geq n$, и $B(L) = B(L, \varphi_u) = m$ при $n_1 \geq m$, поэтому далее будем считать, что $m_1 < n$, $n_1 < m$. Приведенная ниже теорема является следствием теоремы 5, доказанной в работе [4].

Теорема 2. Если $n_1 \neq m_1$, то $B(L) = \max(n_1, m_1)$, иначе $B(L) = n_1 + 1$.

В [4] было показано, что $B(L, \varphi_d) = B(L)$ при $n_1 \geq m_1$ и $B(L, \varphi_u) = B(L)$ при $n_1 \leq m_1$.

В следующих разделах мы определим ширину решеток с двумя прямоугольными угловыми выемками и укажем способы построения оптимальных нумераций.

2. Прямоугольные решетки с двумя выемками в противоположных углах

В данном параграфе рассматривается решеточный граф L с множеством вершин $V(L) = V(G_{m,n}) \setminus (V_1 \cup V_2)$, где $V_1 = [1, a_1] \times [n - b_1 + 1, n]$, $V_2 = [m - a_2 + 1, m] \times [1, b_2]$, $m, n \geq 3$, $a_i \in [1, m - 2]$, $b_i \in [1, n - 2]$, $i = 1, 2$. Этот граф изображен на рис. 2.

Введем обозначения: $p = m - a_1 - a_2$, $q = n - b_1 - b_2$, $m_i = m - a_i$, $n_i = n - b_i$, $i = 1, 2$. Нетрудно показать, что $B(L) = B(L, \varphi_d) = n$ при $p \geq n$, и $B(L) = B(L, \varphi_u) = m$ при $q \geq m$, поэтому далее положим $p < n$, $q < m$. Чтобы исключить вырожденные случаи, будем считать, что $\max(p, q) \geq 1$. Без ограничения общности положим $m_2 \geq n_1 \geq \min(m_1, n_2)$ и

$$\text{если } p, q \leq 1, \text{ то } m_1 \geq n_2. \quad (1)$$

Положим $\delta(L) = \max\{B(H) : H - \text{максимальная по включению прямоугольная подрешетка графа } L\}$. Очевидно, $B(L) \geq \delta(L)$. В графе L существует не больше четырех максимальных по включению прямоугольных подрешеток размерностей $m_2 \times n_1$, $m_1 \times n_2$, $p \times n$ (если $p \geq 1$) и $m \times q$ (если $q \geq 1$). Справедлива следующая

Теорема 3. Если выполнено одно из условий:

- (I) $p = n_1$,
- (II) $n_1 = m_1 < \min(m_2, n_2)$, $n_1(n_1 - 1) - pq - 1 > (2n_1 - p - q - 2)n_1$,
то $B(L) = n_1 + 1$, иначе $B(L) = \delta(L)$.

Подобный результат был доказан в [4] для частного случая $p, q \geq 1$. Доказательство в общем случае аналогично.

Способ построения оптимальной нумерации зависит от значений параметров. Опишем процесс нумерации в случае, когда $p \neq n_1$, $n_1 = m_1 < \min(m_2, n_2)$ и выполнено неравенство

$$n_1(n_1 - 1) - pq - 1 \leq (2n_1 - p - q - 2)n_1, \quad (2)$$

во всех остальных случаях одна из диагональных нумераций является оптимальной.

Без ограничения общности можно считать $p \leq q$. Обозначим через H_1 подграф графа L , порожденный вершинами первых $m_2 + 1$ диагоналей L , а через H_2 – подграф, порожденный вершинами последних $n_2 + 1$ диагоналей. Из (1) и (2) следует, что $H_1 \cap H_2 = \emptyset$.

Введем обозначения: $h = a_2 + b_2 - 3$ – число диагоналей в подграфе $L \setminus (H_1 \cup H_2)$; $d(k)$ – мощность k -й диагонали подграфа $L \setminus (H_1 \cup H_2)$ (т.е. $(k + m_2 + 1)$ -й диагонали графа L), $v_{k,t}$ – t -я сверху вершина k -й диагонали, $k \in [1, h]$, $t \in [1, d(k)]$.

Определим нумерацию φ^* следующим образом. Присвоим номера $1, 2, \dots, |V(H_1)|$ вершинам подграфа H_1 , используя нижнюю диагональную нумерацию, и номера $N - |V(H_2)| + 1, N - |V(H_2)| + 2, \dots, N$ вершинам подграфа H_2 , используя верхнюю диагональную нумерацию. Вершины подграфа $L \setminus (H_1 \cup H_2)$ нумеруются в следующем порядке: $v_{1,1}, v_{1,2}, \dots, v_{1,d(1)}, v_{2,1}, \dots, v_{h,d(h)}$, на шаге (k, t) вершина $v_{k,t}$ получает номер $\varphi^*(v_{k,t}) = \max\{u : u \in U(k, t), u \leq \mu(k, t) + n_1\}$. Здесь $\mu(k, t)$ – минимальный из номеров смежных с $v_{k,t}$ вершин, уже получивших номера; $U(k, t)$ – неиспользованные на предыдущих шагах номера из множества $[\eta(k) + 1, \eta(k) + d(k)]$, где $\eta(k) = \sum_{i=1}^{k-1} d(i) + |V(H_1)|$ – количество вершин на предыдущих диагоналях графа L .

Данный алгоритм строит нумерацию φ^* , ширина которой равна $\delta(L) = n_1$.

3. Прямоугольные решетки с двумя выемками в смежных углах

В данном параграфе рассматривается решеточный граф L с множеством вершин $V(L) = V(G_{m,n}) \setminus (V_1 \cup V_2)$, где $V_1 = [1, a_1] \times [n - b_1 + 1, n]$, $V_2 = [m - a_2 + 1, m] \times [n - b_2 + 1, n]$, $m \geq 4$, $n \geq 3$, $a_i \in [1, m - 3]$, $b_i \in [1, n - 2]$, $i = 1, 2$. Чтобы исключить вырожденные случаи, будем считать, что $a_1 + a_2 \leq m - 2$. Такой граф изображен на рис. 3.

Введем обозначения: $p = m - a_1 - a_2$, $m_i = m - a_i$, $n_i = n - b_i$, $i = 1, 2$. Нетрудно показать, что $B(L) = B(L, \varphi_u) = m$ при $n_2 \geq m$, и $B(L) = B(L, \varphi_d) = n$ при $p \geq n$, поэтому далее будем считать, что $n_2 < m$, $p < n$. Без ограничения общности положим $b_1 \leq b_2$.

Теорема 4. Пусть $\beta = \min\{\beta_1, \beta_2, \beta_3, \beta_4\}$, где $\beta_1 = \max(n_1 + n_2, p)$, $\beta_2 = \max(n_2, p + a_1)$, $\beta_3 = \max(n_1, p + \min(a_2, n_2))$, $\beta_4 = n$. Если выполнено одно из условий: (I) $n_1 + n_2 = p$, (II) $n_2 = p + a_1$, (III) $n_1 = p + \min(a_2, n_2) < \beta_2$, то $B(L) = \beta + 1$, иначе $B(L) = \beta$.

Укажем способ построения оптимальной нумерации φ^* в зависимости от значения β .

1. Пусть $\beta = \beta_1$. Заметим, что в этом случае $n_1 + n_2 \leq p + n_2$, откуда следует $n_1 \leq p$. Определим нумерацию φ^* следующим образом. Сначала трансформируем исходный граф. Удалим из графа L все ребра uv , где $u = (a_1 + n_1, j)$, $v = (a_1 + n_1 + 1, j)$ (в силу сделанного выше замечания $a_1 + n_1 \leq a_1 + p < m$). Получим граф, состоящий из двух компонент связности. Правую компоненту симметрично отразим относительно ее левого края и сдвинем на вектор $(-n_1 - 1, n_1)$. Полученный граф обозначим через L' . Схема преобразований показана на рис. 4.

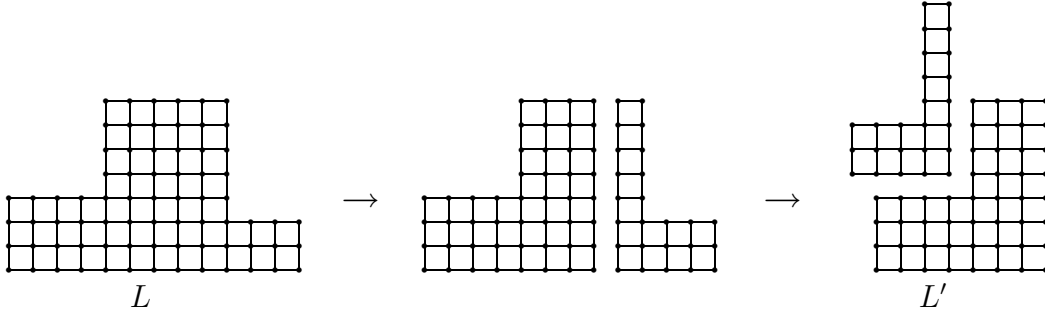


Рис. 4: Преобразование графа L в случае $\beta = \beta_1$

Граф L' состоит из двух непересекающихся решеточных графов. Пронумеруем его вершины согласно нижней диагональной нумерации, если $n_1 + n_2 > p$, и согласно верхней диагональной нумерации, если $n_1 + n_2 \leq p$. Построенной нумерации графа L' соответствует нумерация φ^* исходного графа.

2. $\beta = \beta_2 < \beta_1$. Граф L симметрично отразим относительно его левого края и пронумеруем вершины полученного графа согласно нижней диагональной нумерации, если $n_2 > a_1 + p$, и согласно верхней диагональной нумерации в противном случае. Через φ^* обозначим нумерацию графа L , соответствующую построенной нумерации графа L' .

3. $\beta = \beta_3 < \beta_i, i < 3$. Возможны два случая:

3.1 $a_2 \leq n_2$. Тогда положим $\varphi^* \equiv \varphi_d$ при $n_1 > a_2 + p$, и $\varphi^* \equiv \varphi_u$ в противном случае.

3.2 $a_2 > n_2$. Если $p + n_2 \geq n_1$, определим нумерацию φ^* следующим образом. Удалим из графа L все ребра uv , где $u = (a_1 + p, j)$, $v = (a_1 + p + 1, j)$. Правую компоненту полученного графа симметрично отразим относительно ее левого края и повернем на 90 градусов по часовой стрелке вокруг вершины $w = (a_1 + p + 1, n_2)$. Затем сдвинем ее на вектор $(-p - 1, p)$, если $p + n_2 > n_1$, и на вектор $(-p - 2, p + 1)$, если $p + n_2 = n_1$. Схема преобразований показана на рис. 5.

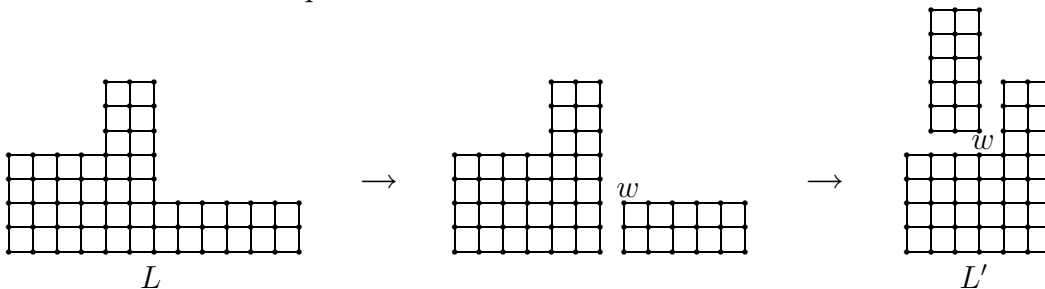


Рис. 5: Преобразование графа L в случае $\beta = \beta_3 < \beta_i, i < 3, a_2 > n_2, p + n_2 \geq n_1$

Пронумеруем вершины полученного графа L' согласно верхней диагональной нумерации. Через φ^* обозначим нумерацию графа L , соответствующую построенной нумерации графа L' .

Пусть теперь $p+n_2 < n_1$. Удалим из графа L все ребра uv , где $u = (i, n_1)$, $v = (i, n_1+1)$. Верхнюю компоненту полученного графа симметрично отразим относительно ее правого края, затем повернем на 90 градусов по часовой стрелке вокруг вершины $w = (a_1+p, n_1+1)$ и сдвинем на вектор $(1, -2)$. Схема преобразований показана на рис. 6.

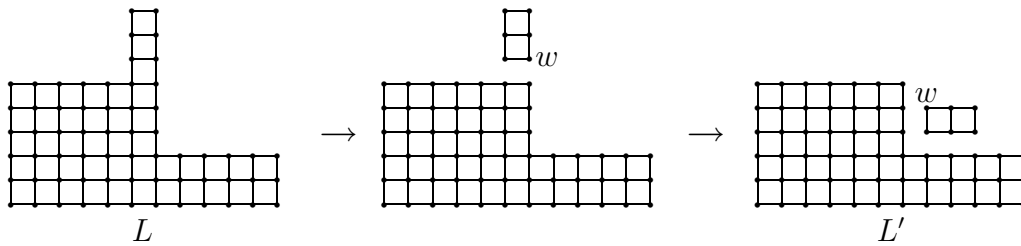


Рис. 6: Преобразование графа L в случае $\beta = \beta_3 < \beta_i$, $i < 3$, $a_2 > n_2$, $p + n_2 < n_1$

Пронумеруем вершины полученного графа L' согласно нижней диагональной нумерации. Через φ^* обозначим нумерацию графа L , соответствующую нумерации графа L' .

4. $\beta = \beta_4 < \beta_i$, $i < 4$. В этом случае положим $\varphi^* \equiv \varphi_d$.

Можно показать, что нумерация φ^* , построенная в каждом из четырех случаев, является оптимальной.

Заключение

Определена ширина прямоугольных решеток с двумя прямоугольными угловыми выемками. Для каждого случая расположения выемок указан способ построения оптимальной нумерации, который может быть представлен в виде полиномиального алгоритма.

Список литературы

- [1] Ch.H. Papadimitriou *The NP-completeness of the bandwidth minimization problem*. Computing, 1976, v. 16, p. 263-270.
- [2] J. Díaz, M.D. Penrose, J. Petit, M.J. Serna *Approximating layout problems on random geometric graphs*. Journal of Algorithms, 2001, v. 39, N1, p. 78-116.
- [3] J. Chvátalová *Optimal labelling of a product of two paths*. Discrete Math., 1975, v. 11, p. 249-253.
- [4] S.D. Ivanova *A polynomial time solvable case of bandwidth minimization problem on lattice graphs*. Proc. of the 2nd Intern. Workshop "Discrete Optimization Methods in Production and Logistics", Omsk-Irkutsk, July, 2004, Omsk: Nasledie. Dialog-Sibir Pbs., 2004, p. 169-175.

THE BANDWIDTH OF LATTICE GRAPHS WITH SPECIAL STRUCTURE

S.D. Ivanova

Omsk State University, Omsk
e-mail: IvanovaSD@yandex.ru

Abstract. Bandwidth minimization problem asks for a permutation of vertices of an undirected graph that minimizes the maximum difference between the endpoints of its edges. Computing the bandwidth is NP-hard even when restricted to lattice graphs. In this paper, we compute bandwidth of the rectangular lattices with two rectangular corner holes and construct an optimal numbering in polynomial time.

Key words: optimization, bandwidth of graph, numbering, lattice graph

ОЦЕНКИ ПОГРЕШНОСТИ ПРИБЛИЖЕННОГО АЛГОРИТМА ДЛЯ ЗАДАЧИ О РАСКРАСКЕ ГРАФА

В.П. Ильев

Омский государственный университет, Омск
e-mail: iljev@iitam.omsk.net.ru

Аннотация. В статье исследуется простой приближенный алгоритм решения задачи о минимальной раскраске графа, которая рассматривается как частный случай задачи о минимальном зависимом множестве наследственной системы. Получены гарантированные оценки погрешности алгоритма для задачи о минимальной раскраске графа в терминах степеней вершин графа и его дополнения.

Ключевые слова: раскраска графа, наследственная система, градиентный алгоритм, оценка погрешности.

Введение

Раскраской неориентированного n -вершинного графа $G = (V, E)$ называется разбиение $P = (V_1, \dots, V_k)$ множества V на попарно непересекающиеся подмножества V_1, \dots, V_k , где каждое V_i – независимое множество вершин графа. Число $k = |P|$ называется *мощностью* или *числом* раскраски P . Задача о минимальной раскраске состоит в отыскании раскраски минимальной мощности (равной хроматическому числу $\chi(G)$ графа G).

Мы будем рассматривать задачу о минимальной раскраске графа как частный случай задачи о минимальном зависимом множестве наследственной системы. В работе [4] получена гарантированная оценка погрешности градиентного алгоритма для задачи о минимальном зависимом множестве. Как следствие этого результата с использованием известных оценок числа независимости получены гарантированные оценки погрешности градиентного алгоритма для задачи о минимальной раскраске графа в терминах степеней его вершин, а также степеней вершин его дополнения.

1. Задача о минимальном зависимом множестве наследственной системы

Пусть U – конечное множество. Определим *наследственную систему* \mathcal{H} на множестве U как разбиение семейства 2^U всех подмножеств U на два непересекающихся непустых семейства \mathcal{A} и $\mathcal{D} = 2^U \setminus \mathcal{A}$, удовлетворяющих следующим *аксиомам наследственности*:

$$1) (A \in \mathcal{A}, A' \subseteq A) \Rightarrow A' \in \mathcal{A}, \quad 2) (D \in \mathcal{D}, D \subseteq D') \Rightarrow D' \in \mathcal{D}.$$

Множества семейства \mathcal{A} называются *независимыми*, а множества из \mathcal{D} – *зависимыми*. Поскольку каждое из семейств \mathcal{A}, \mathcal{D} однозначно определяет наследственную систему \mathcal{H} , мы будем записывать $\mathcal{H} = (U, \mathcal{A})$ или $\mathcal{H} = (U, \mathcal{D})$ в зависимости от того, какая сторона наследственной системы в данный момент нас интересует. Заметим, что пару (U, \mathcal{A}) называют также *системой независимости*.

Хорошо известна *задача о максимальном независимом множестве*, в которой требуется найти независимое множество максимального веса заданной системы независимости $\mathcal{H} = (U, \mathcal{A})$. В настоящей работе нас будет интересовать следующая задача.

Задача о минимальном зависимом множестве.

Дана наследственная система \mathcal{H} на множестве U и неотрицательная аддитивная функция $f : U \rightarrow R_+$. Требуется найти такое $D_O \in \mathcal{D}$, что $f(D_O) = \min\{f(D) : D \in \mathcal{D}\}$, где \mathcal{D} – семейство всех зависимых множеств наследственной системы \mathcal{H} .

Задача о минимальном зависимом множестве NP -трудна, так как ее частными случаями являются многие NP -трудные задачи комбинаторной оптимизации, в том числе задача о покрытии множества, задача о минимальном k -связном остовном подграфе и другие.

Рассмотрим следующий приближенный алгоритм решения задачи о минимальном зависимом множестве, который является дискретным аналогом алгоритма наискорейшего спуска.

Градиентный алгоритм GR.

Шаг 0. Упорядочить множество $U = \{u_1, \dots, u_n\}$ по невозрастанию значений функции f ; $D \leftarrow U$; перейти на шаг 1.

Шаг i ($i = 1, 2, \dots, n$). Если $D \setminus \{u_i\} \in \mathcal{D}$, то $D \leftarrow D \setminus \{u_i\}$; если $i < n$, то перейти на шаг $i + 1$, иначе $D_{GR} \leftarrow D$.

Конец.

Пусть $\mathcal{H} = (U, \mathcal{D})$ – произвольная наследственная система, $W \subseteq U$. *Циклом множества W* называется любое минимальное по включению зависимое множество, содержащее W . Определим *кривизну* наследственной системы $\mathcal{H} = (U, \mathcal{D})$ как

$$c_{\mathcal{D}} = c_{\mathcal{D}}(\mathcal{H}) = \max_{\substack{W \subseteq U, \\ W \notin \mathcal{D}}} \frac{g_U(W) - |W|}{g_L(W) - |W|},$$

где $g_L(W) = \min\{|C| : C \text{ – цикл множества } W\}$ – *нижний обхват множества W* , $g_U(W) = \max\{|C| : C \text{ – цикл множества } W\}$ – *верхний обхват множества W* .

В работе [4] доказана следующая

Теорема 1. Пусть $\mathcal{H} = (U, \mathcal{D})$ – произвольная наследственная система. Тогда для любой аддитивной целевой функции задачи о минимальном зависимом множестве имеет место достижимая оценка

$$\frac{f(D_{GR})}{f(D_O)} \leq c_{\mathcal{D}}, \quad (1)$$

где $c_{\mathcal{D}}$ – кривизна системы \mathcal{H} , $D_O \in \mathcal{D}$ – оптимальное решение задачи о минимальном зависимом множестве, а $D_{GR} \in \mathcal{D}$ – решение, найденное алгоритмом GR.

Для невзвешенного варианта задачи оценка (1) может быть уточнена.

Следствие. [4] Пусть $\mathcal{H} = (U, \mathcal{D})$ – произвольная наследственная система. Тогда

$$\frac{|D_{GR}|}{|D_O|} \leq \frac{g_U(\emptyset)}{g_L(\emptyset)}, \quad (2)$$

где $D_O \in \mathcal{D}$ – оптимальное решение невзвешенной задачи о минимальном зависимом множестве, а $D_{GR} \in \mathcal{D}$ – решение, найденное алгоритмом GR.

2. Оценки погрешности алгоритма для задачи о раскраске графа

Задача о минимальной раскраске графа может быть сведена к задаче о минимальном зависимом множестве наследственной системы следующим образом.

Пусть $P = (V_1, \dots, V_k)$, $P' = (V'_1, \dots, V'_l)$ – две раскраски графа $G = (V, E)$, $k \leq l$. Будем говорить, что P – подраскраска раскраски P' , если P получена из P' объединением некоторых множеств V'_i . Например, $P = (V_1, V_2)$ – подраскраска $P' = (V_1, V'_2, V'_3)$, где $V_2 = V'_2 \cup V'_3$.

Для заданного n -вершинного графа $G = (V, E)$ рассмотрим наследственную систему $\mathcal{H} = (U, \mathcal{D})$, где $U = 2^V$, а \mathcal{D} есть семейство всех раскрасок графа G . Теперь задача о минимальной раскраске графа становится задачей о минимальном зависимом множестве наследственной системы $\mathcal{H} = (U, \mathcal{D})$.

Задача о минимальной раскраске графа является NP -трудной [1]. Для нее известен $O(n(\log \log n)^2 / (\log n)^3)$ -приближенный полиномиальный алгоритм [3].

Для приближенного решения задачи о минимальной раскраске графа применим следующий простой алгоритм, который можно рассматривать как вариант градиентного алгоритма GR для невзвешенной задачи о минимальном зависимом множестве.

Алгоритм COL.

Шаг 0. Начать с тривиальной раскраски $P = (V_1, \dots, V_n)$, где $V_i = \{v_i\}$, $i = 1, \dots, n$, предварительное упорядочение элементов разбиения не требуется; перейти на шаг 1.

Шаг i ($i = 1, \dots, n$). Текущая раскраска P содержит множество V_i . Если существует такой номер $j \in \{i+1, \dots, n\}$, что множество $V_i \cup V_j$ независимо в графе G , то $V_j \leftarrow V_i \cup V_j$ и удалить V_i из P . Если $i < n$, то перейти на шаг $i+1$, иначе $P_{COL} \leftarrow P$.

Конец.

Как следствие оценки (2) может быть получена верхняя оценка мощности раскраски, найденной алгоритмом COL .

Лемма 1. Для любого n -вершинного графа G алгоритм COL находит раскраску P_{COL} , удовлетворяющую условию

$$|P_{COL}| \leq \left\lfloor \frac{n + \chi}{2} \right\rfloor,$$

где $\chi = \chi(G)$ – хроматическое число графа G .

Доказательство. Как и ранее, рассмотрим наследственную систему $\mathcal{H} = (U, \mathcal{D})$, где \mathcal{D} – семейство всех раскрасок графа $G = (V, E)$.

Рассмотрим раскраски P_O , P_1 , такие, что $|P_O| = g_L(\emptyset) = \chi$, $|P_1| = g_U(\emptyset)$. Если P_1 содержит множества $V_i = \{v_i\}$ и $V_j = \{v_j\}$, то вершины v_i и v_j смежны в графе G . Поэтому число тривиальных множеств $V_i = \{v_i\}$ в P_1 не превосходит χ , так как в противном случае $|P_O| > \chi$. Следовательно, $g_U(\emptyset) = |P_1| \leq \chi + \lfloor (n - \chi)/2 \rfloor = \lfloor (n + \chi)/2 \rfloor$. Отсюда,

$$\frac{g_U(\emptyset)}{g_L(\emptyset)} \leq \frac{\lfloor \frac{n+\chi}{2} \rfloor}{\chi}.$$

Таким образом, в силу (2) получаем

$$\frac{|P_{COL}|}{|P_O|} \leq \frac{\lfloor \frac{n+\chi}{2} \rfloor}{\chi}, \quad (3)$$

откуда

$$|P_{COL}| \leq |P_O| \frac{\lfloor \frac{n+\chi}{2} \rfloor}{\chi} = \left\lfloor \frac{n+\chi}{2} \right\rfloor.$$

Следствие. В любом n -вершинном графе G , отличном от пустого, алгоритм COL находит такую раскраску P_{COL} , что

$$\frac{|P_{COL}|}{|P_O|} \leq \frac{\lfloor \frac{n+2}{2} \rfloor}{2}, \quad (4)$$

где P_O – минимальная раскраска графа G .

Действительно, т.к. $\chi(G) \geq 2$, то $\lfloor (n+\chi)/2 \rfloor / \chi \leq \lfloor (n+2)/2 \rfloor / 2$, дальнейшее следует из (3).

Используя лемму 1 и нижние оценки числа независимости графа, можно получить оценки погрешности алгоритма COL в терминах степеней вершин графа G и его дополнения.

Теорема 2. Пусть $G = (V, E)$ – n -вершинный граф, $d'(v)$ – степень вершины $v \in V$ в дополнительном графе G' , $d(G')$ – средняя степень вершин дополнительного графа G' . Тогда

$$\frac{|P_{COL}|}{|P_O|} \leq \frac{d(G') + 2}{2}, \quad (5)$$

$$\frac{|P_{COL}|}{|P_O|} \leq \frac{1}{2} \left(\frac{n}{\sum_{v \in V} \frac{1}{d'(v)+1}} + 1 \right), \quad (6)$$

где P_{COL} – раскраска, полученная алгоритмом COL , а P_O – минимальная раскраска графа G .

Доказательство. Докажем оценку (5). Очевидно, $\chi(G) \geq \alpha(G')$, где $\alpha(G')$ – число независимости графа G' . По теореме Турана [5], $\alpha(G') \geq n/(d(G') + 1)$. Тогда по лемме 1

$$\begin{aligned} \frac{|P_{COL}|}{|P_O|} &\leq \frac{\lfloor \frac{n+\chi}{2} \rfloor}{\chi} \leq \frac{n+\chi}{2\chi} = \frac{1}{2} \left(\frac{n}{\chi} + 1 \right) \leq \frac{1}{2} \left(\frac{n}{\alpha(G')} + 1 \right) \\ &\leq \frac{1}{2} \left(\frac{n(d(G') + 1)}{n} + 1 \right) = \frac{d(G') + 2}{2}. \end{aligned}$$

Докажем оценку (6). В работе [6] (см. также [2]) доказано следующее обобщение оценки Турана:

$$\alpha(G') \geq \sum_{v \in V} \frac{1}{d'(v) + 1}.$$

Далее, повторяя цепочку неравенств доказательства неравенства (5), получаем

$$\frac{|P_{COL}|}{|P_O|} \leq \frac{1}{2} \left(\frac{n}{\alpha(G')} + 1 \right) \leq \frac{1}{2} \left(\frac{n}{\sum_{v \in V} \frac{1}{d'(v)+1}} + 1 \right).$$

Учитывая равенства $d(v) = n - 1 - d'(v)$, $d(G) = n - 1 - d(G')$, где $d(v)$ и $d'(v)$ – степени вершины v в графах G и G' , а $d(G)$ и $d(G')$ – средние степени вершин графов G и G' соответственно, получим аналогичные оценки в терминах степеней вершин графа G .

Следствие 1. Пусть $G = (V, E)$ – n -вершинный граф, $d(v)$ – степень вершины $v \in V$ в графе G , $d(G)$ – средняя степень вершин графа G . Тогда

$$\frac{|P_{COL}|}{|P_O|} \leq \frac{n + 1 - d(G)}{2}, \quad (7)$$

$$\frac{|P_{COL}|}{|P_O|} \leq \frac{1}{2} \left(\frac{n}{\sum_{v \in V} \frac{1}{n-d(v)}} + 1 \right). \quad (8)$$

Следствие 2. Для любого n -вершинного графа G с m ребрами

$$\frac{|P_{COL}|}{|P_O|} \leq \frac{n(n + 1) - 2m}{2n}. \quad (9)$$

Доказательство. Заметим, что $nd(G) = \sum_{v \in V} d(v) = 2m$, откуда в силу (7)

$$\frac{|P_{COL}|}{|P_O|} \leq \frac{n + 1 - d(G)}{2} = \frac{n(n + 1) - nd(G)}{2n} = \frac{n(n + 1) - 2m}{2n}.$$

Замечание. Оценка (4) достигается на графе, полученном из полного двудольного графа $K_{\lfloor (n+1)/2 \rfloor, \lceil (n-1)/2 \rceil}$ удалением паросочетания мощности $\lfloor (n-1)/2 \rfloor$. Оценки (5) – (9) достигаются на полном n -вершинном графе.

Список литературы

- [1] М. Гэри, Д. Джонсон *Вычислительные машины и труднорешаемые задачи* М.: Мир, 1982, 416 с.
- [2] J.R. Griggs *Lower bounds on the independence number in terms of the degrees* - J. Combin. Theory, Ser. B, 1983, v. 34, p. 22-39.
- [3] M.M. Halldórsson *A still better performance guarantee for approximate graph coloring* - Inform. Process. Letters, 1993, v. 45, p. 19-23.
- [4] V. П'ев *Hereditary systems and greedy-type algorithms*. - Discrete Appl. Math., 2003, v. 132, p. 137-148.
- [5] P. Turán *On an extremal problem in graph theory* (in Hungarian) - Mat. Fiz. Lapok, 1941, v. 48, p. 436-452.
- [6] V.K. Wei *A lower bound on the stability number of a simple graph* - Technical Memorandum No. 81-11217-9, Bell Laboratories, 1981.

PERFORMANCE GUARANTEES OF AN APPROXIMATION ALGORITHM FOR GRAPH COLORING

V.P. Il'ev

Omsk State University, Omsk
e-mail: iljev@itam.omsk.net.ru

Abstract. In the paper, we study a simple approximation algorithm for the minimum graph coloring problem. We consider this problem as a special case of the minimization problem on a hereditary system. Performance guarantees of the algorithm for minimum graph coloring in terms of degrees of vertices are obtained.

Key words: graph coloring, hereditary system, greedy algorithm, performance guarantee.

МАТЕМАТИЧЕСКАЯ МОДЕЛЬ ОЦЕНКИ СОСТАВА СИСТЕМЫ ТЕХНИЧЕСКИХ СРЕДСТВ¹

В.С. Ишутенко, Ю.А. Кочетов

*Секция прикладных проблем при Президиуме СО РАН, Новосибирск
Институт математики им. С.Л. Соболева СО РАН, Новосибирск
e-mail: grigor@ad-sbras.nsc.ru, jkochet@math.nsc.ru*

Аннотация. В работе приводится математическая постановка задачи выбора оптимального состава системы в случае, когда сценарий выполнения работ области применения заранее не известен. В качестве критерия оптимизации выступают суммарные затраты на пополнение системы новыми техническими средствами и затраты на выполнение работ. Обсуждаются пути развития модели для случая двух лиц, принимающих решения.

Ключевые слова: система технических средств, линейное программирование, двухуровневое программирование.

Введение

Задачи выбора оптимального состава систем технических средств давно привлекают к себе внимание исследователей [1]. Интерес к ним объясняется не только широкими приложениями, но и необходимостью разрабатывать специализированные методы решения, учитывающие специфику данного класса задач и, как правило, их большую размерность.

В настоящей работе рассматривается математическая модель, в которой состав системы уже известен. Требуется оценить необходимость пополнения системы для того, чтобы выполнить определенный круг работ. Сценарий их выполнения предполагает, что часть работ выполняется на первом этапе, часть на втором и т.д. [2]. К сожалению, задание сценария в значительной степени предопределяет оптимальный состав системы. Какой именно сценарий будет реализован на практике, заранее неизвестно. Новый сценарий, вообще говоря, приводит к новому составу системы, а перебор сценариев не дает ответа на вопрос об оптимальном составе системы, способным выполнить все работы при любом сценарии. В данной работе предлагается новая математическая модель, учитывающая одновременно несколько сценариев. Она не выводит за рамки задач линейного программирования, что позволяет использовать стандартное программное обеспечение [6]. Обсуждаются пути дальнейшего развития этой модели, в том числе и в рамках задач двухуровневого линейного программирования.

1. Основные понятия

Под системой технических средств принято понимать [1] совокупность технических средств (ТС), объединенных общностью функционального назначения и общей сферой применения. В качестве ТС чаще всего рассматривается массовая продукция различных отраслей машиностроения: машины, механизмы, приборы и т.п. Сферу применения исследуемой системы ТС характеризует совокупность видов работ, выполняемых с их ис-

¹Работа выполнена при поддержке РФФИ (проект 03-01-00455)

пользованием. Множество видов работ, выполнение которых должно быть обеспечено исследуемой системой, называют областью применения. Будем предполагать, что процесс выполнения работ области применения представляется как многоэтапный, когда часть работ выполняется на первом этапе, часть на втором и т.д. При этом конкретные ТС, используемые на данном этапе и не вышедшие из строя, могут быть использованы на последующих этапах. Виды работ на каждом этапе количественно будем характеризовать средним числом одновременно выполняемых единичных работ. Состав системы называют допустимым, если система способна обеспечить в требуемых объемах выполнение всех работ области применения. Критерий выполнения каждой единичной работы предполагается известным.

Под суммарными затратами на пополнение и функционирование системы будем понимать сумму затрат на покупку и транспортировку необходимых ТС, а также затрат на выполнение работ области применения, включая затраты на восстановление ТС, вышедших из строя в ходе выполнения работ. Задача выбора оптимального состава системы ТС состоит в следующем. Требуется найти допустимый состав системы и назначить исполнителей каждой работы области применения так, чтобы суммарные затраты были минимальными.

Предположим теперь, что имеется несколько сценариев выполнения работ. Состав системы будем называть допустимым, если он обеспечивает выполнение всех работ в требуемых объемах в каждом отдельно взятом сценарии. Теперь содержательная постановка задачи формулируется следующим образом. Требуется найти допустимый состав системы и назначить исполнителей в каждом сценарии так, чтобы суммарные затраты на пополнение и функционирование системы были минимальными.

2. Математическая модель

Обозначим через I множество образцов технических средств, которые входят или могут входить в состав рассматриваемой системы ТС. Через $v_i^0, i \in I$ обозначим количество ТС i -го образца в составе системы, а через $V_i, i \in I$ предельные возможности пополнения системы изделиями i -го образца. Если $V_i = 0$ для некоторого $i \in I$, то возможность пополнить систему ТС данного образца отсутствует. Затраты на пополнение системы изделиями i -го ТС обозначим через $c_i, i \in I$.

Область применения технических средств зададим множеством работ J . Будем считать, что известно множество сценариев K выполнения работ и для каждого сценария $k \in K$ задано разбиение множества J на непересекающиеся множества $J = \bigcup (J_{lk}, l \in L_k)$, определяющие поэтапный порядок выполнения работ в данном сценарии. Упорядоченное множество L_k задает последовательность этапов k -го сценария. Через φ_{jlk} обозначим число работ j -го вида, подлежащее выполнению на l -м этапе k -го сценария.

Примем следующие предположения относительно правил использования ТС.

1. Каждое ТС на каждом этапе любого сценария может быть использовано только для выполнения одной работы.

2. Единичная работа каждого вида выполняется нарядом ТС одного образца. В соответствии с этим для $i \in I, j \in J$ считаем известным величину $p_{ij} > 0$ равную количеству изделий i -го образца, требующихся для выполнения работы j -го вида и величину $c_{ij} \geq 0$ равную необходимым при этом затратам.

3. При выполнении каждой работы часть изделий выходит из строя. Вышедшие из строя изделия могут быть частично восстановлены во время последующих этапов. Для

$i \in I, j \in J$, считаем известной величину $p_{ij}^1 \in [0, p_{ij}]$, равную количеству ТС i -го образца, вышедших из строя при выполнении j -й работы. Для $i \in I, l, l' \in L_k, l' > l$, предполагаем также известной величину $\gamma_{ill'} \in [0, 1]$ — долю ТС i -го образца, вышедших из строя на l -м этапе и восстановленных в течение $l + 1, \dots, l'$ этапов. Затраты на ремонт изделий i -го образца обозначим через $c_i^1, i \in I$.

Введем следующие переменные:

$v_i \geq 0, i \in I$ — число изделий i -го образца, которыми необходимо пополнить состав системы.

$x_{ijk} \geq 0, i \in I, j \in J_{lk}, l \in L_k, k \in K$ — доля работ j -го вида, выполняемая нарядами ТС i -го образца в k -м сценарии.

С использованием введенных обозначений математическая модель оценки состава системы ТС может быть записана следующим образом. Найти

$$\min_{v_i, x_{ijk}} \sum_{i \in I} (c_i v_i + \max_{k \in K} \sum_{l \in L_k} \sum_{j \in J_{lk}} \varphi_{jlk} x_{ijk} (c_{ij} + c_i^1 p_{ij}^1))$$

при ограничениях

$$\begin{aligned} \sum_{i \in I} x_{ijk} &= 1, \quad j \in J_{lk}, l \in L_k, k \in K; \\ 0 &\leq v_i \leq V_i, \quad i \in I; \\ \sum_{j \in J_{lk}} \varphi_{jlk} p_{ij}^1 x_{ijk} &\leq v_i^0 + v_i - \sum_{l'=1}^{l-1} \sum_{j \in J_{l'k}} \varphi_{j l' k} p_{ij}^1 x_{ij k} (1 - \gamma_{ill'}), \quad i \in I, l \in L_k, k \in K; \\ x_{ijk} &\geq 0, \quad i \in I, j \in J_{lk}, l \in L_k, k \in K. \end{aligned}$$

Целевая функция задает суммарные затраты на пополнение системы новыми ТС, затраты на выполнение работ и ремонт вышедших из строя ТС при реализации одного из рассматриваемых сценариев. Первое ограничение требует выполнения всех работ, какой бы из сценариев не был реализован. Второе ограничение устанавливает верхнюю границу на возможные объемы пополнения системы. Третье ограничение задает связь между требуемым количеством ТС на каждом этапе выполнения работ и имеющимся количеством ТС к этому этапу с учетом потерь и скорости восстановления изделий.

Сформулированная задача в отличие от [2] не содержит целочисленных переменных и может быть решена стандартными средствами линейного программирования [6].

3. Пути развития модели

Рассмотрим несколько вариантов уточнения модели с целью более детального описания ее компонентов.

3.1. Учет ремонтпригодности. В целевой функции последнее слагаемое задает стоимость ремонта вышедших из строя изделий. Возможно, что не все изделия подлежат восстановлению и часть из них следует заменить новыми. Это обстоятельство можно учесть, если ввести в рассмотрение величину $p_{ij}^2 \in [0, p_{ij}^1]$, равную количеству ТС i -го образца, вышедших из строя при выполнении j -й работы и не подлежащих восстановлению. При новых предположениях о ремонтпригодности ТС, последнее слагаемое в целевой функции принимает вид

$$c_i^1 (p_{ij}^1 - p_{ij}^2) + c_i p_{ij}^2.$$

Ограничения задачи не меняются, но задание матрицы $\gamma_{il'}$ должно быть согласованным с величинами p_{ij}^2 , т.е. $p_{ij}^1(1 - \gamma_{il'}) \geq p_{ij}^2$, для всех $i \in I, j \in J, l, l' \in L_k, l > l'$.

3.2. Возможности пополнения системы. Ограничения задачи устанавливают верхнюю границу на объемы пополнения системы. Эта граница задается для каждого образца ТС отдельно. Альтернативным вариантом могло бы стать суммарное ограничение на дополнительно привлекаемые ТС:

$$\sum_{i \in I} \lambda_i v_i \leq V,$$

где коэффициенты λ_i задают важность (вес, приоритет, цену и т.п.) изделий i -го образца, а величина V определяет суммарные возможности по привлечению ТС в систему. Заметим, что в данном случае речь не идет о контроле за номенклатурой образцов. Попытки управлять качественным составом системы приводят к частично-целочисленным моделям [1] и выводят за рамки линейного программирования.

3.3. Модель двух лиц, принимающих решения. Следуя [3], рассмотрим ситуацию, когда для покрытия части расходов приходится привлекать средства инвестора. Обозначим через $B > 0$ величину кредита и будем предполагать, что инвестор хочет получить на эти средства часть изделий по специальным ценам $\beta_i > 0, i \in I$. Важность каждого образца для инвестора обозначим через $\alpha_i > 0, i \in I$. Если состав системы $v_i^0 + v_i, i \in I$ уже известен, то инвестор стремится максимизировать свою выгоду, выбирая значения переменных $\omega_i \geq 0, i \in I$ — число изделий i -го образца, направляемых инвестору:

$$\max_{\omega_i} \left\{ \sum_{i \in I} \alpha_i \omega_i \mid \sum_{i \in I} \beta_i \omega_i \leq B, \omega_i \leq v_i^0 + v_i, i \in I \right\}.$$

Зная задачу инвестора, требуется выбрать допустимый состав системы, имеющий минимальные суммарные затраты, т.е. найти

$$\min_{v_i, x_{ijk}} \sum_{i \in I} (c_i v_i + \max_{k \in K} \sum_{l \in L_k} \sum_{j \in J_{lk}} \varphi_{jlk} x_{ijk} (c_{ij} + c_i^1 (p_{ij}^1 - p_{ij}^2) + c_i p_{ij}^2))$$

при ограничениях

$$\begin{aligned} \sum_{i \in I} x_{ijk} &= 1, \quad j \in J_{lk}, l \in L_k, k \in K; \\ 0 &\leq v_i \leq V_i, \quad i \in I; \\ \sum_{i \in I} \lambda_i v_i &\leq V; \\ \sum_{j \in J_{lk}} \varphi_{jlk} p_{ij} x_{ijk} &\leq v_i^0 + v_i - \omega_i^* - \sum_{l'=1}^{l-1} \sum_{j \in J_{l'k}} \varphi_{jl'k} p_{ij}^1 x_{ijl'k} (1 - \gamma_{il'l}), \quad i \in I, l \in L_k, k \in K; \\ x_{ijk} &\geq 0, \quad i \in I, j \in J_{lk}, l \in L_k, k \in K, \end{aligned}$$

где ω_i^* — оптимальное решение задачи инвестора. Сформулированная задача относится к классу задач двухуровневого линейного программирования [5]. Для ее решения предложен точный алгоритм полиномиальной трудоемкости [4].

Список литературы

- [1] В.Л. Береснев *Математические модели развития систем технических средств* - Дискрет. анализ и исслед. операций. Сер. 2. 2001. т. 7, N1. с. 78–96.
- [2] Ю.А. Кочетов, М.Г. Пащенко *Лагранжевы релаксации в задаче выбора оптимального состава системы технических средств* - Управляемые системы. Сб. науч. тр. Вып. 31. Новосибирск: Институт математики СО АН СССР, 1993. с. 26–39.
- [3] Ю.А. Кочетов, А.В. Плясунов *Задача выбора ряда изделий с частичным внешним финансированием* - Дискрет. анализ и исслед. операций. Сер. 2. 2002. т. 9, N2. с. 78–96.
- [4] Ю.А. Кочетов, А.В. Плясунов *Полиномиально разрешимый класс задач двухуровневого линейного программирования* - Дискрет. анализ и исслед. операций. Сер. 2. 1997. т. 4, N2. с. 23–33.
- [5] О. Ben-Ayed *Bilevel linear programming* - Computers and Operations Research, 1993, т. 20. с. 485–501.
- [6] J.T. Linderoth. T.K. Ralphs *Noncommercial Software for Mixed-Integer Linear Programming*, http://www.optimization-online.org/DB_HTML/2004/12/1028.html

MATHEMATICAL MODEL TO EVALUATE THE COMPOSITION OF A TECHNICAL TOOL SYSTEM

V. Ishutenko, Yu. Kochetov

*Application Section at the Presidium of the Siberian Branch RAS, Novosibirsk
Sobolev Institute of Mathematics, Novosibirsk
e-mail: grigor@ad-sbras.nsc.ru, jkochet@math.nsc.ru*

Abstract. In this paper we introduce mathematical model to evaluate the composition of a technical tool system for the case when scenario of the tasks fulfilment is not defined in advance. As a criterion optimization we use the total cost for additional tools and the cost to carry out the tasks. Further directions for improvement are discussed for the model with two decision makers.

Key words: technical tool system, linear programming, bilevel programming.

АЛГОРИТМЫ ЛЕКСИКОГРАФИЧЕСКОГО ПЕРЕБОРА ДЛЯ РЕШЕНИЯ ЗАДАЧИ ВЫПОЛНИМОСТИ И НЕКОТОРЫХ ЕЕ ОБОБЩЕНИЙ

А.А. Колоколов, А.В. Адельшин, Д.И. Ягофарова

Омский филиал Института математики им. С.Л. Соболева СО РАН, Омск
e-mail: kolo@iitam.omsk.net.ru, adelshin@iitam.omsk.net.ru, yagofarova@iitam.omsk.net.ru

Аннотация. Рассматриваются задачи выполнимости и максимальной выполнимости логической формулы. Предлагается новый вариант алгоритма перебора L -классов для задачи выполнимости, а также построенные на его основе алгоритмы лексикографического перебора булевых векторов для невзвешенной задачи максимальной выполнимости. Обсуждаются результаты вычислительного эксперимента.

Ключевые слова: задача выполнимости, задача максимальной выполнимости, целочисленное программирование, L -разбиение, перебор L -классов.

Введение

Значительное число исследований в области дискретной оптимизации посвящено задаче выполнимости логической формулы и ее обобщениям [1-6]. Широкое практическое применение этих задач в экономике, управлении, проектировании и других областях является стимулом для разработки алгоритмов их решения. Обзор алгоритмов для задачи выполнимости представлен в [5]. В работах [1-3, 6] проводились исследования задач выполнимости и максимальной выполнимости на основе моделей целочисленного линейного программирования (ЦЛП) и L -разбиения.

Ранее для задачи выполнимости был разработан алгоритм [3], основанный на методе перебора L -классов. В данной статье описывается новый вариант алгоритма такого типа, краткое сообщение о нем имеется в [2]. Кроме того, предложены алгоритмы лексикографического перебора булевых векторов для невзвешенной задачи максимальной выполнимости, основанные на сведениях к последовательности задач выполнимости. Для указанных алгоритмов, проведены вычислительные эксперименты с использованием электронной библиотеки [8].

1. Постановки задач

Приведем постановку задачи выполнимости (SAT). Пусть x_1, \dots, x_n — переменные, принимающие значение *истина* или *ложь*. Под литералом понимается либо переменная x_j , либо ее отрицание \bar{x}_j . Рассмотрим логическую формулу F , представляющую собой конъюнкцию формул (скобок) $C_i, i = 1, \dots, t$, каждая из которых является дизъюнкцией литералов. Требуется определить, выполнима ли формула F , т.е. существует ли такой набор значений переменных, при котором F принимает значение *истина*.

Известно, что задача SAT является NP-полной, а в случае, когда каждая скобка содержит не более двух литералов, задача полиномиально разрешима [4].

Приведем модель ЦЛП для задачи SAT. Для этого введем булевы переменные y_1, \dots, y_n такие, что y_j соответствует переменной x_j , а $(1 - y_j)$ — ее отрицанию. Условие выполни-

мости формулы F эквивалентно существованию решения системы:

$$\sum_{j \in C_i^+} y_j - \sum_{j \in C_i^-} y_j \geq 1 - |C_i^-|, \quad i = 1, \dots, m, \quad (1)$$

$$0 \leq y_j \leq 1, \quad j = 1, \dots, n, \quad (2)$$

$$y_j \in Z, \quad j = 1, \dots, n, \quad (3)$$

где C_i^- и C_i^+ — множества индексов переменных, входящих в скобку C_i с отрицанием и без него соответственно. В качестве целевой функции можно рассматривать, например, $f(y) = y_1 \rightarrow \max$ или $f(y) = \sum_{j=1}^n y_j \rightarrow \max$, где $y = (y_1, \dots, y_n)$. В данной работе используется лексикографическая постановка задачи ЦЛП (см. п.2).

Пусть скобки C_1, \dots, C_m имеют неотрицательные веса c_1, \dots, c_m . Задача *максимальной выполнимости* (MAX-SAT) состоит в отыскании набора значений переменных, при котором суммарный вес выполненных скобок будет наибольшим. Сформулируем соответствующую модель ЦЛП:

$$\sum_{i=1}^m c_i z_i \rightarrow \max, \quad (4)$$

$$\sum_{j \in C_i^-} y_j - \sum_{j \in C_i^+} y_j + z_i \leq |C_i^-|, \quad i = 1, \dots, m, \quad (5)$$

$$0 \leq y_j \leq 1, \quad y_j \in Z, \quad j = 1, \dots, n, \quad (6)$$

$$0 \leq z_i \leq 1, \quad z_i \in Z, \quad i = 1, \dots, m. \quad (7)$$

Здесь z_i — вспомогательная булева переменная, соответствующая скобке C_i , $i = 1, \dots, m$. Если в допустимом решении задачи (4)-(7) для некоторого i имеет место $z_i = 1$, то формула C_i принимает значение *истина*. Оптимальное значение целевой функции равно максимальному суммарному весу выполненных скобок. В данной работе рассматривается невзвешенный вариант задачи MAX-SAT, в котором все c_i равны 1. Эта задача является NP-трудной, даже если каждая скобка в формуле содержит не более двух литералов.

2. Метод перебора L -классов

Введем необходимые определения и обозначения. Точки $x, y \in R^n$ ($x \succ y$) называются *L -эквивалентными*, если не существует отделяющей их точки $z \in Z^n$, т.е. такой, что выполняется $x \succeq z \succeq y$. Здесь \succ, \succeq — знаки лексикографического сравнения. Такое отношение порождает разбиение любого множества на непересекающиеся классы эквивалентности, которые называются *L -классами*. Соответствующее фактор-множество называется *L -разбиением* множества X . Указанное разбиение обладает рядом полезных свойств [7], применяемых при разработке и исследовании алгоритмов целочисленного программирования, в частности, алгоритмов перебора L -классов.

Нами используется также лексикографическое сравнение множеств. Пусть $X, Y \subset R^n$. Будем считать, что $X \succ Y$, если для любых $x \in X$ и $y \in Y$ выполняется $x \succ y$.

Для описания алгоритма решения задачи выполнимости рассмотрим следующую лексикографическую постановку задачи ЦЛП:

$$\text{найти } y^* = \text{lexmax}(M \cap Z^n), \quad (8)$$

где M — многогранник, определяемый системой ограничений (1)–(2). Множество

$$M_* = \{y' \in M : y' \succ y \text{ для всех } y \in (M \cap Z^n)\}$$

называется *дробным накрытием* задачи (8), а M_*/L — ее L -*накрытием*.

Приведем идею метода перебора L -классов для задачи (8). Пусть L -накрытие задачи состоит из классов V_1, V_2, \dots, V_p , причем $V_1 \succ V_2 \succ \dots \succ V_p$.

Основной шаг метода заключается в переходе от одного класса из M_*/L к другому в порядке лексикографического убывания. В процессе перебора алгоритм порождает последовательность точек $y^{(t)} \in M_*$, обладающую свойствами:

- $y^{(t)} \succ y^{(t+1)}$, $t = 1, 2, \dots, s - 1$; $s \leq p$;
- все точки принадлежат различным L -классам.

В общем случае для задачи ЦЛП переход от одного L -класса к другому выполняется путем решения некоторого числа задач линейного программирования. Для решения задачи выполнимости поиск представителей $y^{(t)}$ указанных классов можно осуществлять, не решая задач линейного программирования, что заметно повышает эффективность алгоритма. Это основано на следующем свойстве [3]: в каждом L -классе многогранника M содержится *полуцелочисленная точка*, т.е. точка со значениями координат из множества $\{0, \frac{1}{2}, 1\}$. Поиск представителей осуществляется с помощью специальной комбинаторной процедуры, в процессе выполнения которой строится лексикографически убывающая последовательность недопустимых целочисленных точек.

3. Алгоритм перебора L -классов для задачи выполнимости

Приведем описание предлагаемого нами алгоритма перебора L -классов для задачи SAT, который за конечное число шагов в случае выполнимости формулы находит лексикографически максимальное решение задачи (8), либо указывает на невыполнимость формулы.

В алгоритме используется *процедура упрощения формулы*, которая включает следующие основные преобразования:

- если найдена скобка x_j , то исключаем из формулы литерал \bar{x}_j и скобки, содержащие литерал x_j ; переменной y_j “автоматически” присваиваем значение 1;
- если найдена скобка \bar{x}_j , то исключаем из формулы литерал x_j и скобки, содержащие литерал \bar{x}_j ; переменной y_j “автоматически” присваиваем значение 0;
- в ходе упрощения проверяется наличие единичных скобок x_j и \bar{x}_j для некоторого j ; существование такой пары скобок означает, что получено *противоречие*.

Упрощение продолжается до тех пор пока в текущей формуле имеются единичные скобки.

Предполагается, что перед началом работы алгоритма исходная формула не содержит единичных скобок. Алгоритм завершает работу в двух случаях: если не удастся найти очередной L -класс или если в формуле не осталось скобок. В последнем случае алгоритм переходит на заключительный шаг, где оставшимся переменным присваиваются значения, равные 1. Следует отметить, что указанным переменным можно присвоить любые булевы значения. Это дает множество различных допустимых решений задачи (8).

Алгоритм LCE

Шаг 0 (поиск представителя первого L -класса). Находим полуцелочисленную точку $y^{(1)} \in V_1$. Переходим к шагу 1 первой итерации.

Итерация t ($t \geq 1$).

Шаг 1. Начинаем с точки $y^{(t)} = (a_1^{(t)}, \dots, a_{s-1}^{(t)}, \frac{1}{2}, y_{s+1}^{(t)}, \dots, y_n^{(t)})$, где $a_i^{(t)} \in \{0, 1\}$, $i = 1, \dots, s-1$, — зафиксированы; $y_j^{(t)} \in \{0, \frac{1}{2}, 1\}$, $j = s+1, \dots, n$.

Если в формуле не осталось скобок, то переходим на шаг 4.

Иначе выполняем следующее.

Первую дробную координату $y_s^{(t)}$ округляем до 0. Выполняем *процедуру упрощения формулы*. В ходе этого преобразования некоторые координаты $y_j^{(t)}$ ($j = s+1, \dots, n$) “автоматически” примут значения 0 или 1. Обозначим полученную точку $\tilde{y}^{(t)}$. После упрощения формулы возникает один из случаев:

- В формуле не осталось скобок. Рассматривая точку $\tilde{y}^{(t)}$, переходим на шаг 4.
- Скобки остались, противоречие не обнаружено. Переходим на шаг 2.
- Скобки остались, обнаружено противоречие. Возвращаемся к точке $y^{(t)}$. Переходим на шаг 3.

Шаг 2 (переход к следующему L -классу). Находим полуцелочисленную точку $y^{(t+1)} \in V_{t+1}$. Переходим к шагу 1 итерации ($t+1$).

Шаг 3. Среди координат вектора $y^{(t)}$ ищем максимальную по номеру координату $a_p^{(t)} = 1$, которой это значение не было присвоено “автоматически” ($p < s$). Возможны случаи:

- Такая координата *существует*, тогда просматриваем вектор $y^{(t)}$. Если после назначения $a_p^{(t)} = 1$ какой-то переменной было присвоено значение 0 или 1, то меняем его на $\frac{1}{2}$. Значение координаты с номером p уменьшаем до $\frac{1}{2}$. В результате получаем точку $y^{(t+1)} \in V_{t+1}$ и переходим на шаг 1 итерации ($t+1$).
- Такой координаты *не существует*. В этом случае алгоритм завершает работу: *формула невыполнима*.

Шаг 4 (заключительный). Алгоритм завершает работу: *формула выполнима*. Оптимальное решение y^* строим из текущего вектора $\tilde{y}^{(t)}$, заменяя дробные значения координат на 1.

4. Алгоритмы для задачи максимальной выполнимости

Для решения невзвешенной задачи MAX-SAT нами предложены точные алгоритмы, основанные на сведении к последовательности задач SAT. Решение задач выполнимости осуществляется алгоритмом LCE. Отметим, что вместо него можно применять и другие точные алгоритмы.

Приведем краткое описание алгоритмов решения задачи MAX-SAT. Каждой скобке C_i поставим в соответствие булеву переменную \tilde{z}_i , $i = 1, \dots, m$. Тогда логической формуле F будет отвечать множество булевых m -векторов $\tilde{z}^k = (\tilde{z}_1^k, \dots, \tilde{z}_m^k)$, где $k = 1, \dots, 2^m$. Разобьем множество всех векторов \tilde{z}^k на подмножества следующим образом: \tilde{z}^{k_1} и \tilde{z}^{k_2} принадлежат одному подмножеству, если число их нулевых компонент одинаково. При этом

будем говорить, что все векторы с числом нулевых компонент, равным τ , образуют τ -слой ($\tau = 0, \dots, m$).

Идея алгоритмов состоит в частичном переборе τ -слоев по некоторым правилам, например, с учетом текущего рекордного значения целевой функции ρ или в порядке возрастания τ . На каждом τ -слое осуществляется направленный перебор векторов \tilde{z}^k в соответствии с лексикографическим порядком. В процессе решения для \tilde{z}^k формулируется следующая задача: найти такой набор значений логических переменных, при котором скобка C_i выполнена тогда и только тогда, когда $\tilde{z}_i^k = 1$, $i = 1, \dots, m$. Если решение задачи найдено, то текущее рекордное значение ρ обновляется. После рассмотрения τ -слоя ищем новый слой, содержащий векторы с числом нулей, меньшим τ . Если такой слой существует, то переходим к перебору векторов на нем. В противном случае алгоритм завершает работу, текущее значение ρ является оптимальным значением целевой функции.

Предложенную схему можно использовать при построении алгоритмов решения других обобщений задачи выполнимости, например, задачи минимальной выполнимости [1].

Алгоритмы реализованы в среде DELPHI 6.0, проведен вычислительный эксперимент на сериях задач из класса Uniform-Random-3-SAT из библиотеки SATLIB [8] и на других тестовых примерах. Результаты расчетов показывают перспективность применения метода перебора L -классов для решения задачи выполнимости и ее обобщений.

Список литературы

- [1] А.В. Адельшин *Исследование задач максимальной и минимальной выполнимости с использованием L -разбиения*. - Автоматика и телемеханика, 2004, N3, с. 35-42.
- [2] Ю.Н. Калльрат, А.А. Колоколов, Д.И. Ягофарова *Алгоритм перебора L -классов для задачи выполнимости*. - Материалы Всероссийской конференции "Проблемы оптимизации и экономические приложения", Омск, 2003, с. 94.
- [3] А.А. Колоколов, А.В. Адельшин, Ю.Н. Чередова *Применение L -разбиения к исследованию некоторых задач выполнимости*. - Труды XII Байкальской межд. конференции, Иркутск, 2001, т. 1, с. 166-172.
- [4] S.A. Cook *The complexity of theorem-proving procedure*. - Proc. 3rd Annual ACM Symposium on the Theory of Computing, 1971, p. 151-159.
- [5] J. Gu, P. Purdom, J. Franco, B. Wah *Algorithms for the Satisfiability (SAT) Problem: A Survey*. - DIMACS Series in Discrete Mathematics and Theoretical Computer Science, 1996, 131 p.
- [6] A. Kolokolov, J. Kallrath, D. Yagofarova *Analysis and Solving the Satisfiability Problem using L -partition*. - Operations Research 2003. Annual International Conference of the German Operations Research Society, University of Heidelberg, 2003, p. 128.
- [7] А.А. Колоколов *Регулярные разбиения и отсеечения в целочисленном программировании*. - Сиб. журнал исследования операций, 1994, N2, с. 18-39.
- [8] www.cs.ubc.ca/~hoos/SATLIB/benchm.html

LEXICOGRAPHICAL ENUMERATION ALGORITHMS FOR THE SAT AND SOME OF ITS GENERALIZATIONS

A.A. Kolokolov, A.V. Adelshin, D.I. Yagofarova

Omsk Branch of Sobolev Institute of Mathematics SB RAS, Omsk

e-mail: kolo@iitam.omsk.net.ru, adelshin@iitam.omsk.net.ru, yagofarova@iitam.omsk.net.ru

Abstract. A lot of investigations in discrete optimization are devoted to the satisfiability (SAT) and the maximum satisfiability (MAX-SAT) problems. We study these problems using the integer programming models and L -partition approach. We suggest a new version of L -class enumeration algorithm (LCE) for the SAT and algorithms of lexicographical enumeration for the unweighted MAX-SAT which constructed on the basis of the LCE. Results of computational experiment are discussed.

Key words: satisfiability problem, maximum satisfiability problem, integer programming, L -partition, L -class enumeration.

ЗАДАЧИ ДИСКРЕТНОЙ ОПТИМИЗАЦИИ И ПРОГРАММНЫЙ КОМПЛЕКС ДЛЯ ЭСКИЗНОГО ПРОЕКТИРОВАНИЯ ОДЕЖДЫ

А.А. Колоколов

*Омский филиал Института математики им. С.Л. Соболева СО РАН, Омск
e-mail: kolo@iitam.omsk.net.ru*

З.Е. Нагорная

Омский государственный институт сервиса, Омск

О.Н. Гуселетова

*Омский государственный университет, Омск
e-mail: guseletova@promex.ru*

А.В. Ярош

*Сибирская государственная автомобильно-дорожная академия, Омск
e-mail: a.v.yarosh@rambler.ru*

Аннотация. Развивается предложенный нами подход к автоматизации эскизного проектирования одежды, основанный на использовании задач дискретной оптимизации с логическими и ресурсными ограничениями. Описываются математические модели и разработанный на базе указанного подхода программный комплекс для эскизного проектирования верхней женской плечевой одежды. Обсуждаются результаты вычислительного эксперимента для различных силуэтных форм демисезонных пальто.

Ключевые слова: дискретная оптимизация, автоматизация, проектирование одежды, логические ограничения, технический эскиз.

Введение

Одним из важнейших этапов проектирования одежды является процесс создания эскизов новых моделей. От эскизного проектирования во многом зависит конкурентоспособность готовой продукции на рынке швейных изделий. В настоящее время для решения данной задачи существуют специализированные дизайнерские системы [1 – 3]. При их использовании приходится перебирать и сравнивать большое количество сочетаний деталей и элементов одежды, поэтому в ряде случаев достаточно интересные и перспективные модели могут оказаться не рассмотренными, а выбранные варианты не всегда будут наилучшими. В связи с этим весьма актуальным является более широкое использование математического аппарата в указанном направлении.

В [4 – 8] и других работах нами развивается подход к автоматизации эскизного проектирования одежды, основанный на использовании задач дискретной оптимизации с логическими и ресурсными ограничениями. Логические ограничения естественным образом возникают и учитываются при использовании различных положений и рекомен-

даций, вытекающих из теории гармонизации костюма [9].

В данной работе предлагаются новая постановка задачи эскизного проектирования одежды и математические модели, обобщающие рассмотренные ранее постановки и задачи в направлении учета групп составляющих (деталей, усложняющих элементов) и характеристик одежды (силуэтных форм, объемов, свойств материалов и т.д.). Описывается созданный на этой основе программный комплекс и приводятся результаты вычислительного эксперимента для женских демисезонных пальто.

1. Постановка задачи

Рассмотрим процесс проектирования технических эскизов одежды, которые формируются из множества составляющих и характеристик. Для формулировки задачи введем следующие обозначения:

g - число групп составляющих и характеристик;

α - номер группы составляющих и характеристик, $\alpha \in G$, $G = \{1, \dots, g\}$;

n_α - число элементов в группе α , $J_\alpha = \{1, \dots, n_\alpha\}$;

v_j^α - составляющая или характеристика с номером j из группы α ;

x_j^α - логическая переменная, которая принимает значение *истина*, если v_j^α входит в изделие, и - значение *ложь* в противном случае;

s_{lj}^α - вес составляющей или характеристики v_j^α по l -му показателю, характеризующему степень целесообразности ее включения в изделие, $l \in L$, $L = \{1, \dots, w\}$, $j \in J_\alpha$;

p_l - нижняя граница для суммарной значимости составляющих по l -му показателю, включенных в изделие;

C_i - логическая формула, соответствующая i -му логическому ограничению, $i \in I$, которая представляет собой дизъюнкцию переменных x_j^α и/или их отрицаний $\overline{x_j^\alpha}$; эти формулы связывают переменные, относящиеся к различным группам составляющих и характеристик; отметим, что формулы с номерами $I' \subseteq I$ должны быть обязательно выполнены;

d_i - вес формулы C_i , характеризующий степень необходимости ее выполнения, $i \in I \setminus I'$;

$\overline{m_\alpha}$ - число логических ограничений для группы α , $I_\alpha = \{1, \dots, m_\alpha\}$;

$\overline{C_{\alpha r}}$ - логическая формула, которая отвечает r -му логическому ограничению и связывает переменные группы α , $r \in I_\alpha$, причем формулы с номерами из $I'_\alpha \subseteq I_\alpha$ должны быть обязательно выполнены;

d_r^α - вес формулы $\overline{C_{\alpha r}}$, характеризующий степень необходимости ее выполнения, $r \in I_\alpha \setminus I'_\alpha$;

a_{kj}^α - объем k -го ресурса, требуемого для изготовления j -ой составляющей изделия группы α , $k \in K$; $j \in J_\alpha$;

b_k - имеющийся объем k -го ресурса, $k \in K$.

Требуется найти значения логических переменных, при которых выполняются формулы C_i , $i \in I'$ и $\overline{C_{\alpha r}}$, $r \in I'_\alpha$, $\alpha \in G$, ограничения по ресурсам и по суммарной значимости включенных в изделие составляющих, а общий вес выполненных формул C_i , $i \in I \setminus I'$ и $\overline{C_{\alpha r}}$, $r \in I_\alpha \setminus I'_\alpha$, $\alpha \in G$, будет максимальным.

Чтобы построить модель ЦЛП, необходимо от логических переменных перейти к булевым, а логические ограничения заменить соответствующими им линейными неравенствами. Введем множества индексов логических переменных $C_{\alpha i}^-$ и $C_{\alpha i}^+$, входящих в C_i из

группы α , с отрицанием и без него, соответственно, для всех $i \in I$. Аналогично введем множества индексов переменных $\widetilde{C_{\alpha r}^-}$ и $\widetilde{C_{\alpha r}^+}$ для формул $\widetilde{C_{\alpha r}}$, $r \in I_\alpha$, $\alpha \in G$. Заменяем x_j^α на булеву переменную y_j^α , ее отрицание $\overline{x_j^\alpha}$ на $1 - y_j^\alpha$, а символ дизъюнкции \vee на знак $+$. Логической формуле C_i поставим в соответствие следующее линейное неравенство ("жесткое" ограничение):

$$\sum_{\alpha \in G} \left(\sum_{j \in C_{\alpha i}^-} y_j^\alpha - \sum_{j \in C_{\alpha i}^+} y_j^\alpha \right) \leq |C_i^-| - 1.$$

Если формула C_i при определенных условиях может не выполняться, то для нее вводится вспомогательная булева переменная z_i и строится неравенство ("мягкое" ограничение):

$$\sum_{\alpha \in G} \left(\sum_{j \in C_{\alpha i}^-} y_j^\alpha - \sum_{j \in C_{\alpha i}^+} y_j^\alpha \right) + z_i \leq |C_i^-|.$$

Аналогично для формул $\widetilde{C_{\alpha r}}$, $r \in I_\alpha \setminus I'_\alpha$ вводятся вспомогательные переменные z_r^α .

Модель ЦЛП для данной задачи имеет вид:

$$y_0 = \sum_{i \in I \setminus I'} d_i z_i + \sum_{\alpha \in G} \sum_{r \in I_\alpha \setminus I'_\alpha} d_r^\alpha z_r^\alpha \rightarrow \max \quad (1)$$

при условиях:

$$\sum_{\alpha \in G} \left(\sum_{j \in C_{\alpha i}^-} y_j^\alpha - \sum_{j \in C_{\alpha i}^+} y_j^\alpha \right) \leq |C_i^-| - 1, \quad i \in I', \quad (2)$$

$$\sum_{\alpha \in G} \left(\sum_{j \in C_{\alpha i}^-} y_j^\alpha - \sum_{j \in C_{\alpha i}^+} y_j^\alpha \right) + z_i \leq |C_i^-|, \quad i \in I \setminus I'; \quad (3)$$

$$\sum_{j \in C_{\alpha r}^-} y_j^\alpha - \sum_{j \in C_{\alpha r}^+} y_j^\alpha \leq |\widetilde{C_{\alpha r}^-}| - 1, \quad r \in I'_\alpha; \quad \alpha \in G, \quad (4)$$

$$\sum_{j \in C_{\alpha r}^-} y_j^\alpha - \sum_{j \in C_{\alpha r}^+} y_j^\alpha + z_r^\alpha \leq |\widetilde{C_{\alpha r}^-}|, \quad r \in I_\alpha \setminus I'_\alpha; \quad \alpha \in G, \quad (5)$$

$$\sum_{\alpha \in G} \sum_{j \in J_\alpha} s_{lj}^\alpha y_j^\alpha \geq p_l, \quad l \in L, \quad (6)$$

$$\sum_{\alpha \in G} \sum_{j \in J_\alpha} a_{kj}^\alpha y_j^\alpha \leq b_k, \quad k \in K, \quad (7)$$

$$z_i, z_r^\alpha, y_j^\alpha \in \{0, 1\}, \quad i \in I \setminus I'; \quad r \in I_\alpha \setminus I'_\alpha; \quad j \in J_\alpha; \quad \alpha \in G. \quad (8)$$

В целевой функции (1) максимизируется сумма весов выполненных логических формул. "Жесткие" логические ограничения, которые обязательно должны быть выполнены, задаются условиями (2), (4). Системы неравенств (3), (5) - это "мягкие" логические ограничения, необходимость выполнения которых зависит от степени их важности. Ограничения (2),(3) связывают составляющие и характеристики, относящиеся к различным

группам, а (4),(5) - составляющие или характеристики внутри одной группы. Требования проектировщика по включению в изделие составляющих задаются условиями (6). Неравенства (7) представляют собой ограничения для используемых ресурсов, например, по расходу материалов и т.д., (8) - условие булевости переменных.

Если в оптимальном решении этой задачи для некоторого i имеет место $z_i = 1$, то соответствующая формула C_i принимает значение истина. Аналогичное утверждение справедливо для переменных z_r^α и формул $\widetilde{C_{ar}}$.

2. Программный комплекс

С целью апробации описанного подхода и построенных математических моделей нами создан программный комплекс для автоматизации проектирования серий и единичных эскизов одежды с учетом логических, ресурсных и других ограничений.

Комплекс представляет собой достаточно сложную систему, состоящую из следующих основных модулей: процедуры ввода и корректировки исходных данных для построения технических эскизов; математических моделей и алгоритмов поиска оптимальных решений; базы данных характеристик, составляющих изделий, их изображений, готовых моделей технических эскизов, логических и других ограничений; модулей визуализации решений и промежуточных вариантов, подготовки отчета.

В программном комплексе предусмотрено три режима построения технических эскизов: автоматизированный, полуавтоматизированный и интерактивный.

На этапе визуализации с помощью полученного решения осуществляется обращение к базе данных изображений составляющих моделей и строится один или несколько технических эскизов, которые при необходимости можно занести в базу данных, как готовые модели одежды. Полученные эскизы анализируются и в случае их соответствия требованиям технического задания на проектирование осуществляется переход на этапы оформления отчета и вывода на печать.

Следует отметить, что с помощью созданного программного обеспечения можно анализировать готовые эскизы по значениям ряда показателей с целью оценки их качества, а также проводить сравнение нескольких эскизов для выбора наилучшего из них.

3. Вычислительный эксперимент

С целью апробации предложенных математических моделей и разработанного программного комплекса нами были выполнены экспериментальные исследования. Для этого использовалась задача проектирования женского демисезонного пальто, в которой учитывались 36 составляющих (9 из них были зафиксированы) и 37 логических ограничений.

Проведенные расчеты показали, что с помощью программного комплекса можно достаточно быстро получать множество разнообразных вариантов единичных технических эскизов, а также серий моделей одежды. Комплекс может быть использован для анализа и совершенствования разработанных эскизов, а также в учебном процессе при подготовке специалистов швейной промышленности и сферы сервиса.

Список литературы

- [1] Кривобородова Е.Ю., Коблякова Е.Б., Перегняк Л.Р. *Современные методы формирования графических изображений при автоматизации проектирования одежды*. Швейная промышленность. - 2000. - №5, с. 46-47.
- [2] www.assol.mipt.ru
- [3] www.koppermann.com
- [4] Колоколов А.А., Нагорная З.Е., Гуселетова О.Н., Ярош А.В., Богутова Т.М. *Автоматизация эскизного проектирования одежды с использованием некоторых задач дискретной оптимизации*. Российская конф. Дискретный анализ и исследование операций. Материалы конф. - Новосибирск, ИМ им. С.Л. Соболева СО РАН, 2004, с. 223.
- [5] Колоколов А.А., Нагорная З.Е., Гуселетова О.Н., Ярош А.В. *Применение задач дискретной оптимизации с логическими ограничениями для автоматизации эскизного проектирования одежды*. Междунар. науч.-технич. конф. "Динамика систем, механизмов и машин", 2004г. ОмГТУ. Омск: Изд-во ОмГТУ, 2004, с. 281-284.
- [6] Колоколов А.А., Ярош А.В. *Проектирование одежды с использованием некоторых моделей дискретной оптимизации*. Омский научный вестник. - 2002. - Вып. 20, с. 91 - 94.
- [7] Колоколов А.А., Ярош А.В. *Автоматизация эскизного проектирования одежды с использованием моделей дискретной оптимизации*. Препринт. - Омск: Изд-во ОмГТУ, 2004, 24 с.
- [8] Ярош А.В., Гуселетова О.Н., Богутова Т.М. *Разработка алгоритмов и программ для автоматизации эскизного проектирования одежды*. Молодежь, наука, творчество - 2004. Сборник статей II межвуз. науч.-практич. конф. студентов и аспирантов, 2004г. Ч. 2. - ОГИС, 2004, с. 152 - 153.
- [9] Козлова Т.В. *Основы теории проектирования костюма*. Учеб. Для вузов. - М.: Лег-промбытиздат, 1988, 352 с.

DISCRETE OPTIMIZATION PROBLEMS AND SOFTWARE FOR OUTLINE DESIGNING THE CLOTHES

A.A. Kolokolov

*Omsk Branch of Sobolev Institute of Mathematics, Omsk
e-mail: kolo@iitam.omsk.net.ru*

Z.E. Nagornaya

Omsk State Institute of Service , Omsk

O.N. Guseletova

*Omsk State University, Omsk
e-mail: guseletova@promex.ru*

A.V. Yarosh

*Siberian State Automobile and Highway Academy, Omsk
e-mail: a.v.yarosh@rambler.ru*

Abstract. We develop the approach to CAD of the clothes, based on the application of some discrete optimization problems with the logical and other restrictions. The corresponding mathematical model and the software for outline designing the clothes are described. Computing experiment for several silhouette forms of women coats was carried out.

Keywords: discrete optimization, integer programming, CAD, outline designing, logical restrictions, clothes.

ПРИБЛИЖЕННЫЙ АЛГОРИТМ ДЛЯ ЗАДАЧИ ТЕОРИИ РАСПИСАНИЯ С ОГРАНИЧЕНИЕМ НА ЧИСЛО ВОЗОБНОВЛЯЕМЫХ РЕСУРСОВ, ПОТРЕБЛЯЕМЫХ ОДНОЙ РАБОТОЙ ¹

А.В.Кононов

*Институт математики им. С. Л. Соболева СО РАН, Новосибирск
e-mail: alvenko@math.nsc.ru*

Аннотация. Рассматривается задача теории расписаний с возобновляемыми ресурсами. Каждая работа имеет директивные сроки на ее начало и завершение. Требуется максимизировать взвешенное число выполненных работ. Предлагается приближенный полиномиальный алгоритм основанный на технике локальных отношений.

Ключевые слова: Приближенный алгоритм, теория расписаний, возобновляемые ресурсы.

Введение

Рассмотрим следующую задачу P теории расписаний. Даны множество работ $\mathcal{J} = \{J_1, J_2, \dots, J_n\}$ и множество возобновляемых ресурсов (машин) $Res = \{1, 2, \dots, L\}$. Для каждой работы J_i заданы интервал времени $[r_i, d_i]$, время p_i , требуемое на ее выполнение, и ее вес w_i . Без ограничения общности будем считать, что $p_i \leq d_i - r_i$. Пусть $S_i(\sigma)$ и $C_i(\sigma)$ обозначают время начала и время завершения выполнения работы J_i в некотором расписании σ . В допустимом расписании каждая работа может быть выполнена только внутри заданного интервала, то есть $r_i \leq S_i(\sigma)$ и $C_i(\sigma) \leq d_i$. Кроме того, для каждой работы J_i задано множество $Res(i) \subseteq Res$ требуемых ресурсов. В период своего выполнения работа потребляет весь запас требуемого ресурса. Следовательно, никакие две работы использующие одинаковый ресурс не могут обслуживаться одновременно. Последнее замечание позволяет рассматривать множество ресурсов, как множество специализированных параллельных машин. Пусть $U_i(\sigma) = 1$, если работа J_i выполнена в расписании σ и $U_i(\sigma) = 0$, в противном случае. Требуется найти допустимое расписание, максимизирующее взвешенное число выполненных работ $\sum_{i=1}^n w_i U_i$. В дальнейшем, мы предположим, что все параметры задачи – целые, неотрицательные числа.

Сформулированная задача является NP -трудной в сильном смысле даже для случая с одним ресурсом, поскольку к ней сводится задача нахождения допустимого расписания множества работ на одной машине с директивными сроками на начало и завершение обслуживания.

Ситуация, в которых работа одновременно выполняется на нескольких специализированных машинах, по-видимому, впервые была рассмотрена Кравчиком и Кубале [6]. Они рассмотрели задачу $P|fix_j|C_{max}$, в которой все работы должны быть выполнены в интервале $[0, \infty]$, и требуется минимизировать длину расписания. Авторы показали, что она является NP -трудной даже в случае, если каждая работа требует для обслуживания ровно две машины, то есть $|L_i| = 2$ для всех $i = 1, \dots, n$. Обзор результатов по задачам теории расписаний с работами, требующими для выполнения больше одного процессора одновременно, представлен в [5].

¹Работа выполнена при поддержке РФФИ (проект 03-01-00455)

1. Алгоритм

В [2] Бар-Йегуда и Евен разработали новую эффективную технику для построения приближенных алгоритмов. Ниже мы дадим краткое описание этого метода для задач на максимум.

Пусть $\omega \in R^n$ – вектор стоимости и F – множество ограничений на вектора $x \in R^n$. Вектор $x \in R^n$ является допустимым решением индивидуальной задачи (F, ω) , если он удовлетворяет всем ограничениям из F . Стоимость решения равна ωx . Допустимое решение оптимально, если его значение максимально среди всех допустимых решений. Допустимое решение x называется ρ -приближенным, если $\omega x \geq \rho \omega x^*$, где x^* – оптимальное решение.

Теорема 1. [2] Пусть $\omega, \omega_1, \omega_2$ будут вектора стоимостей такие, что $\omega = \omega_1 + \omega_2$. Тогда, если вектор x является ρ -приближенным решением одновременно для индивидуальных задач (F, ω_1) и (F, ω_2) , то он является ρ -приближенным решением и для индивидуальной задачи (F, ω) .

Основываясь на этой теореме в [1] была предложена общая схема построения алгоритмов для задач на максимум. Мы покажем, что эта схема может быть реализована для задачи P .

Пусть $M(i)$ – множество интервалов в которых может быть расписана работа J_i , то есть $M(i) = \{[a, a + p_i) | r_i \leq a, a + p_i \leq d_i\}$. И пусть $\mathcal{M} = \bigcup_{i=1}^n M(i)$. Сопоставим каждому интервалу $I \in M(i)$ стоимость $\omega(I) = w_i$ и множество требуемых ресурсов $Res(I) = Res(i)$. Обозначим через $A(I)$ множество интервалов конкурирующих с интервалом $I \in M(i)$, $A(I) = \{I' \in M(i)\} \cup \{I' | I \cap I' \neq \emptyset \text{ и } Res(I) \cap Res(I') \neq \emptyset\}$. Таким образом, множество $A(I)$ содержит все интервалы, сопоставленные той же работе, что и интервал I , включая и его самого, и все интервалы, которые пересекаются с I и требуют один из ресурсов из $Res(I)$. Теперь решение задачи P эквивалентно выбору максимального по стоимости подмножества интервалов $\tilde{\mathcal{M}} \subseteq \mathcal{M}$, удовлетворяющего условию $|\tilde{\mathcal{M}} \cap A(i)| = 1$ для всех $I \in \tilde{\mathcal{M}}$. Перед описанием общей схемы работы алгоритма введем еще одно обозначение. Пусть $e(I)$ будет правым концом интервала I .

Алгоритм $Approx(\mathcal{M}, \omega)$.

1. If $\mathcal{M} = \emptyset$, then **STOP Return** $\tilde{\mathcal{M}} := \emptyset$.
2. If существует \mathcal{I} с $\omega(\mathcal{I}) \leq 0$,
3. **then call** $Approx(\mathcal{M} \setminus \mathcal{I}, \omega)$,
4. **else**
5. выберем $\tilde{I} \in \mathcal{M}$ такой что $e(\tilde{I}) = \min_{I \in \mathcal{M}} e(I)$.
6. Положим $\omega_1(I) = \begin{cases} \omega(\tilde{I}), & I \in A(\tilde{I}); \\ 0, & \text{иначе.} \end{cases}$
7. $\omega_2(I) = \omega(I) - \omega_1(I)$.
8. $\mathcal{M}' := Approx(\mathcal{M}, \omega_2)$.
9. **If** $\mathcal{M}' \cup \tilde{I}$ допустимое решение:
10. **Return** $\tilde{\mathcal{M}} := \mathcal{M}' \cup \tilde{I}$

11. **else Return** $\tilde{\mathcal{M}} := \mathcal{M}'$.

Алгоритм $Approx(\mathcal{M}, \omega)$ стартует с множества интервалов \mathcal{M} и вектора стоимостей ω . Поскольку в процессе работы алгоритма стоимости изменяются и даже могут стать отрицательными, вначале каждой итерации интервалы с неположительными стоимостями отбрасываются. Из оставшихся интервалов выбирается интервал \tilde{I} , который заканчивается не позднее других и исходная задача разбивается на две со стоимостями ω_1 и ω_2 . Вектор ω_1 имеет ненулевые компоненты только для интервалов конкурирующих с интервалом \tilde{I} . Таким образом в любом решении со стоимостями ω_1 только элементы множества $A(\tilde{I})$ вносят вклад в целевую функцию. Кроме того, любое решение, полученное алгоритмом $Approx(\mathcal{M}, \omega)$ содержит по крайней мере один интервал из множества $A(\tilde{I})$. Последние два замечания позволяют нам оценить относительную погрешность решения для задачи с вектором стоимостей ω_1 .

Лемма 1. Пусть $Res(i) \leq k$ для всех $J_i \in \mathcal{J}$, тогда решение, получаемое алгоритмом $Approx(\mathcal{M}, \omega)$ является $1/(k+1)$ -приближенным решением на векторе стоимостей ω_1 .

Доказательство. Пусть интервал \tilde{I} соответствует работе J_i . Поскольку, он заканчивается не позднее других интервалов множества \mathcal{M} , то все интервалы из множества $A(\tilde{I})$, соответствующие работам отличным от J_i пересекаются между собой. Каждый такой интервал требует как минимум одного ресурса из множества $Res(i)$. Следовательно, не более k таких интервалов присутствует в любом допустимом решении. Кроме того, допустимое решение содержит не более одного интервала работы J_i . Таким образом, стоимость любого решения ограничена сверху величиной $(k+1)\omega(\tilde{I})$. Так как любое решение, полученное алгоритмом $Approx(\mathcal{M}, \omega)$ содержит по крайней мере один интервал из множества $A(\tilde{I})$, то его стоимость как минимум не менее $\omega(\tilde{I})$. Лемма доказана.

Лемма 2. Пусть $Res(i) \geq k$ для всех $J_i \in \mathcal{J}$, тогда решение, получаемое алгоритмом $Approx(\mathcal{M}, \omega)$ является $1/(\lfloor L/k \rfloor + 1)$ -приближенным решением на векторе стоимостей ω_1 .

Доказательство. Так как общее число ресурсов равно L для доказательства достаточно заметить, что не более $\lfloor L/k \rfloor$ работ могут выполняться одновременно.

В заключение параграфа отметим, что так как решение для вектора стоимостей ω_2 находится рекурсивно, то используя теорему 1 нетрудно индукцией по числу рекурсивных вызовов показать следующее.

Замечание 3. Если решение, получаемое алгоритмом $Approx(\mathcal{M}, \omega)$ является ρ -приближенным на векторе стоимостей ω_1 , то оно является и ρ -приближенным решением на исходном векторе ω .

2. Реализация алгоритма

Несмотря на ряд достоинств алгоритма $Approx(\mathcal{M}, \omega)$, отмеченных в предыдущем параграфе, заметим, что он обладает существенным недостатком: множество \mathcal{M} имеет бесконечную мощность. Поэтому необходимо сказать несколько слов о реализации алгоритма. Конечно хранить все множество интервалов, в которых может быть расписана каждая работа, нерационально. Достаточно просто указать в каком временном отрезке начало выполнения работы не приводит к нарушению директивных сроков. При этом в ходе работы

алгоритма может возникнуть ситуация, когда прибыль получаемая от работы, зависит от момента начала ее выполнения. Перед началом работы алгоритма стоимость всех начальных моментов одной работы одинакова. То есть достаточно указать один временной промежуток для каждой работы и его стоимость. При каждой декомпозиции стоимостей в строке б алгоритма каждый такой интервал может быть разбит не более чем на два других соответствующих той же работе, но с разными стоимостями. Из целочисленности исходных данных легко получить, что число разных стоимостей ограничено величиной $W = \max_{i=1, \dots, n} w_i$. Таким образом алгоритм $Approx(\mathcal{M}, \omega)$ является "псевдополиномиальным" алгоритмом.

Для того, чтобы время работы алгоритма стало полиномиальным мы немного пожертвуем его точностью. Рассмотрим произвольное фиксированное число $0 < \varepsilon < 1$. Пусть интервал \tilde{I} соответствует работе J_i . Напомним, что вначале каждой итерации алгоритм $Approx(\mathcal{M}, \omega)$ удаляет интервалы с неположительными стоимостями. Мы изменим это условие и будем отбрасывать интервал \tilde{I} в том случае, если его стоимость не превышает εw_i . Назовем новый алгоритм $Approx(\mathcal{M}, \omega, \varepsilon)$.

Теорема 2. 1. Если $Res(i) \leq k$ для всех $J_i \in \mathcal{J}$, то алгоритм $Approx(\mathcal{M}, \omega, \varepsilon)$ является $(1 - \varepsilon)/(k + 1)$ -приближенным алгоритмом для задачи P .
 2. Если $Res(i) \geq k$ для всех $J_i \in \mathcal{J}$, то алгоритм $Approx(\mathcal{M}, \omega, \varepsilon)$ является $(1 - \varepsilon)/(\lfloor L/k \rfloor + 1)$ -приближенным алгоритмом для задачи P .

Доказательство следует из лемм 1, 2 и замечания 3.

3. Неклассическая задача Open Shop

Рассмотрим следующую многостадийную задачу MP теории расписаний. Даны множество работ $\mathcal{J} = \{J_1, J_2, \dots, J_n\}$ и m машин. Каждая работа имеет по одной операции на каждой машине. Две операции, принадлежащие одной работе или выполняющиеся на одной и той же машине, не могут обслуживаться одновременно. Для каждой операции O_{ij} задано время p_{ij} , требуемое на ее выполнение, и два директивных срока r_{ij} и d_{ij} . Все операции должны быть выполнены и операция O_{ij} не может начать выполняться до момента r_{ij} . Прерывания в выполнении операций запрещены.

В 1984 году Блажевич [3] рассмотрел новый критерий оптимизации.

$$Y = \sum_{i=1}^n \sum_{j=1}^m \max\{C_{ij} - \max\{d_{ij}, S_{ij}\}, 0\} \rightarrow \min .$$

В нем для каждой работы рассматривается время, затраченное на ее выполнение после директивного срока. Затем оно суммируется по всем работам и полученную величину требуется минимизировать. В отличие от классического критерия минимизации суммарного запаздывания штраф, за нарушение директивного срока одной работой в критерии Y не превышает длительности этой работы. Задача является NP -трудной в сильном смысле даже для случая двух машин и нулевых длительностей всех операций на одной из них.

Подробное описание результатов для многостадийных задач с критерием Y приведено в диссертации Штерны [4]. Отметим, что все известные на сегодняшний день результаты связаны либо с выделением полиномиально разрешимых случаев, либо решением задачи при наличии прерываний. Построение приближенных алгоритмов с гарантированной

оценкой точности сопряжено со следующей трудностью: значение целевой функции может равняться 0, причем проверка существования решения с таким значением сама по себе является NP -трудной задачей. Эта же проблема, в частности, возникает и для классического критерия минимизации максимального запаздывания и некоторых других.

Заметим что задача MP имеет естественную эквивалентную интерпретацию. Вместо минимизации времени, затраченного на выполнение работ после их директивных сроков, можно максимизировать общее время, потраченное на работы до наступления их директивных сроков. Тогда критерий оптимизации принимает вид

$$Y = \sum_{i=1}^n \sum_{j=1}^m \max\{\min\{d_{ij}, S_{ij}\} - C_{ij}, 0\} \rightarrow \max.$$

Обозначим новую задачу через \overline{MP} .

Рассмотрим задачу P с mn работами $\mathcal{J} = \{J_{11}, J_{12}, \dots, J_{nm}\}$ и $n + m$ ресурсами. Пусть работа J_{ij} имеет вес $w_{ij} = p_{ij}$ и требует два ресурса i и $n + i$. Можно заметить, что задача \overline{MP} близка к сформулированной задаче. Действительно, первый ресурс работы J_{ij} в задаче P отражает принадлежность операции O_{ij} работе J_i в задаче \overline{MP} , а второй ресурс работы J_{ij} указывает на какой машине выполняется операция O_{ij} . Единственное отличие состоит в том, что в задаче P работы не могут нарушать директивные сроки d_{ij} . Чтобы устранить этот недостаток мы добавим $1/\varepsilon$ работ в задаче P для каждой операции O_{ij} из \overline{MP} . Для каждого $t = 1, \dots, 1/\varepsilon$ определим работу J_{ijt} с $r_{ijt} = d_{ij} - p_{ij} + t\varepsilon p_{ij}$, $d_{ijt} = d_{ij} - p_{ij} + t\varepsilon p_{ij} + \varepsilon p_{ij}$ и $w_{ijt} = p_{ij} - t(\varepsilon + 1)p_{ij}$.

Применив алгоритм $Approx(\mathcal{M}, \omega, \varepsilon)$, получим приближенное решение с оценкой $(1 - \varepsilon)/3$.

Теорема 3. *Существует $(1 - \varepsilon)/3$ -приближенный алгоритм для задачи \overline{MP} .*

Список литературы

- [1] Bar-Noy A., Bar-Yehuda R., Freund A., Naor J., Schieber B. *A unified approach to approximating resource allocation and scheduling*. Journal of ACM, 2001, v.48, N5, p. 1069-1090.
- [2] Bar-Yehuda R., Even S. *A local-ratio theorem for approximation the weighted vertex covering problem*. Annals of Discrete Mathematics, 1985, v.25, p. 27-46.
- [3] Blazewich J., *Scheduling preemptible tasks on parallel processors with information loss*. Recherche Technique et Science Informatiques, 1984 v.3, N6, p. 415-420.
- [4] Sterna M. *Problems and algorithms in non-classical shop scheduling*. Doctoral thesis, 2000, Osrodek Wydawnictw Naukowych Scientific Publishers, Poznan.
- [5] Bo Chen, Potts C. N., Woeginger G. J. *A Review of Machine Scheduling: Complexity, Algorithms and Approximability*. in D.-Z. Du and P.M. Pardalos (Eds.) Handbook of Combinatorial Optimization, Kluwer Academic Publishers, 1998, p. 21–169.
- [6] Krawczyk H., Kubale M. *An approximation algorithm for diagnostic test scheduling in multicomputer system*. IEEE Transactions on Computing, 1985, v.34, p. 869–872.

AN APPROXIMATION ALGORITHM FOR A SCHEDULING PROBLEM WITH A BOUND ON THE NUMBER OF RESOURCES PER JOB

A. Kononov

Sobolev Institute of Mathematics, Novosibirsk
e-mail: alvenko@math.nsc.ru

Abstract. We consider a scheduling problem with a set of renewable resources. Each job has release date and deadline. The objective is to maximize the weighted number of completed jobs. We present a polynomial time approximation algorithm based on local ratio technic.

Key words: approximation algorithms, scheduling theory, renewable resources .

ВЫЧИСЛИТЕЛЬНОЕ ИССЛЕДОВАНИЕ ДВУХ ПРИБЛИЖЕННЫХ АЛГОРИТМОВ ДЛЯ ТРЕХИНДЕКСНОЙ ПЛАНАРНОЙ ПРОБЛЕМЫ ВЫБОРА

М.К. Кравцов, С.А. Дичковская

*Научно-исследовательский экономический институт Министерства экономики Республики Беларусь, Минск
e-mail: dichkovskaya@mail.ru*

Аннотация. При некоторых дополнительных условиях на коэффициенты целевой функции проведены обширные вычислительные исследования двух полиномиальных алгоритмов нахождения асимптотически оптимального решения трехиндексной планарной проблемы выбора.

Ключевые слова: планарная проблема выбора, асимптотически оптимальное решение, приближенный алгоритм.

Введение

Известно [1], что трехиндексная планарная проблема выбора является NP-трудной и имеет широкое применение на практике при решении задач, таких как распределение капиталовложений и основных инвестиций, составление расписаний [1, 2].

В [3] анализируется возможность распространения подхода к нахождению асимптотически оптимального решения, предложенного в [4, 5] для многоиндексных аксиальных транспортных задач, на r -индексную планарную проблему выбора (r -планарную ПВ). В результате, при некоторых дополнительных условиях на коэффициенты целевой функции был предложен полиномиальный алгоритм α нахождения асимптотически оптимального решения r -планарной ПВ, использующий общеизвестную схему "жадных" эвристик. Однако, позднее в [6] был построен контрпример, показывающий, что этот алгоритм находит матрицу, которая не является планом данной задачи. Поэтому, объявленный в [3] алгоритм α , в настоящее время подвергается сомнению, т.е. не является полностью обоснованным. В связи с этим возникает вопрос: как часто алгоритм α на практике строит план r -планарной ПВ.

В [7] предложен и обоснован новый алгоритм β нахождения асимптотически оптимального решения 3-планарной ПВ порядка n за $O(n^4)$ действий, постановка которой заключается в минимизации целевой функции [8]

$$f(X) = \sum_{i=1}^n \sum_{j=1}^n \sum_{t=1}^n c_{ijt} x_{ijt}$$

при условиях

$$\sum_{i=1}^n x_{ijt} = 1 \quad \forall (j, t) \in N_n^2, \quad (1)$$

$$\sum_{j=1}^n x_{ijt} = 1 \quad \forall (i, t) \in N_n^2, \quad (2)$$

$$\sum_{t=1}^n x_{ijt} = 1 \quad \forall (i, j) \in N_n^2, \quad (3)$$

$$x_{ijt} = 0 \text{ или } 1 \quad \forall (i, j, t) \in N_n^3, \quad (4)$$

где $N_n = \{1, 2, \dots, n\}$, $N_n^2 = N_n \times N_n$, $N_n^3 = N_n \times N_n \times N_n$, $C = \|c_{ijt}\|_n$ - заданная трехиндексная матрица с действительными элементами, $X = \|x_{ijt}\|_n$ - исходный план 3-планарной ПВ.

В настоящем докладе проведены экспериментальные исследования алгоритмов α и β для 3-планарной ПВ.

1. Асимптотически оптимальные алгоритмы α и β

Пусть $V(n, r)$ - множество всех матриц $C = \|c_{ijt}\|_n$, элементы которых равномерно и независимо принимают целочисленные значения из отрезка $[1, r]$, где $r = r(n)$ - либо константа, либо $r(n) \rightarrow \infty$ при $n \rightarrow \infty$. Пусть $V^\sigma(n, r)$ - множество всех тех матриц C из множества $V(n, r)$, которые обладают свойством σ . Будем говорить, что почти каждая матрица множества $V(n, r)$ обладает свойством σ , если $\lim_{n \rightarrow \infty} \frac{|V^\sigma(n, r)|}{|V(n, r)|} = 1$. Будем также говорить, что некоторый алгоритм φ почти всегда строит асимптотически оптимальный план X^* 3-планарной ПВ, если существует такая последовательность $\varepsilon_n(\varphi) \geq 0$, $\lim_{n \rightarrow \infty} \varepsilon_n(\varphi) = 0$, что для почти каждой матрицы C из множества $V(n, r)$ выполняется неравенство $f(X^*) \leq f(X^0)(1 + \varepsilon_n(\varphi))$, где X^0 - оптимальный план 3-планарной ПВ.

Алгоритм α нахождения приближенного решения 3-планарной ПВ состоит из последовательно проводимых шагов.

k -й шаг алгоритма α ($1 \leq k \leq n^2$). Находим элемент матрицы $C = \|c_{ijt}\|_n$, удовлетворяющий условию $c_{i_k, j_k, t_k} = \min\{c_{ijt} : (i, j, t) \in G_k\}$, где $G_1 = N_n^3$. Полагаем $x_{i_k, j_k, t_k}^* = 1$, $x_{i, j_k, t_k}^* = 0 \quad \forall i \in N_n \setminus \{i_k\}$, $x_{i_k, j, t_k}^* = 0 \quad \forall j \in N_n \setminus \{j_k\}$, $x_{i_k, j_k, t}^* = 0 \quad \forall t \in N_n \setminus \{t_k\}$.

Определяем множество

$$G_{k+1} = G_k \setminus Q_k,$$

где

$$Q_k = \{(i, j_k, t_k) : i \in N_n \setminus \{i_k\}\} \cup \{(i_k, j, t_k) : j \in N_n \setminus \{j_k\}\} \cup \{(i_k, j_k, t) : t \in N_n \setminus \{t_k\}\}.$$

Легко видеть, что за n^2 шагов построим матрицу X^* , которая, вообще говоря, может и не удовлетворять условиям (1)–(4). Нетрудно также убедиться, что трудоемкость алгоритма α составляет $O(n^5)$ операций.

Теорема 1 [3]. Алгоритм α почти всегда строит асимптотически оптимальный план 3-планарной ПВ, если $r \leq n^{2(1-\theta)}$, где θ - любое число, удовлетворяющее неравенствам $1/2 < \theta < 1$.

Алгоритм β нахождения приближенного решения 3-планарной ПВ состоит из последовательно проводимых шагов.

k -й шаг алгоритма β ($1 \leq k \leq n$). По матрице $C = \|c_{ijt}\|_n$ формируем двухиндексную матрицу $C^k = \|c_{ij}^k\|_n$, элементы которой определяются следующим образом:

$$c_{ij}^k = \begin{cases} c_{ijk}, & \text{если } (i, j) \in N_n^2 \setminus R_{k-1}, \\ M, & \text{если } (i, j) \in R_{k-1}, \end{cases}$$

где $R_0 = \emptyset$, M – некоторое достаточно большое положительное число. С помощью алгоритма Диница [9], трудоемкость которого равна $O(n^3)$ операций, решаем двухиндексную задачу о назначениях с матрицей C^k и находим ее оптимальный план $X^{(k)*} = \|x_{ijk}^*\|_n$.

Определяем множество

$$R_k = R_{k-1} \cup \{(i, j) : (i, j) \in N_n^2, x_{ijk}^* = 1\}.$$

Очевидно, что за n шагов алгоритм β строит план $X^* = \|x_{ijt}^*\|_n$, удовлетворяющий условиям (1)-(4). Нетрудно также убедиться, что трудоемкость алгоритма β составляет $O(n^4)$ операций.

Теорема 2 [7]. Алгоритм β почти всегда строит асимптотически оптимальный план 3-планарной ПВ, если $r \leq n^{1-\theta}$, где θ - любое число, удовлетворяющее неравенствам $1/2 < \theta < 1$.

2. Вычислительное исследование алгоритмов α и β

Алгоритмы α и β нахождения асимптотически оптимального решения 3-планарной ПВ программно реализованы на языке Object Pascal (в среде Delphi) с использованием компьютера Pentium 4, 512 Mb RAM, ОС Windows 2000 и по ним проведены вычислительные эксперименты. Исходная информация (трехиндексная матрица C) для которых формировалась с помощью датчика случайных чисел, настроенного на работу с целыми числами из отрезка $[1, r]$, где $2 \leq r \leq n$. Проверка плана X^* , построенного с помощью алгоритмов α и β , на оптимальность осуществлялась на основе следующего утверждения: если элементы матрицы C принимают целочисленные значения из отрезка $[1, r]$, то план X 3-планарной ПВ, для которого выполняется равенство $f(X) = n^2$, является оптимальным. С помощью алгоритма $\alpha(\beta)$ было решено 36000 задач (по 1000 задач для каждого $n \in \{150, 200, 250, 300, 350, 400\}$ и $r \in \{2, [\sqrt{n}], [n/4], [n/2], [3n/4], n\}$). Здесь $[m]$ - наибольшее целое число, не превосходящее числа m .

Результаты вычислительных экспериментов представлены в таблицах 1 и 2.

Таблица 1

Результаты вычислительных экспериментов на тестовых 3-планарных ПВ, полученные с помощью алгоритма α

№ серии	r	Доля задач, решаемых оптимально с помощью алгоритма α (в %) при n равном						Доля задач, для которых алгоритм α не строит плана (в %) при n равном					
		150	200	250	300	350	400	150	200	250	300	350	400
1	2	65,5	70,0	76,2	84,4	93,6	95,1	12,0	9,4	7,1	4,9	2,8	1,6
2	$[\sqrt{n}]$	46,1	54,7	59,2	66,1	72,3	74,6	14,2	11,6	8,7	6,4	3,0	2,2
3	$[n/4]$	32,1	39,3	45,4	52,1	60,0	65,4	16,2	14,3	10,8	7,4	4,6	3,5
4	$[n/2]$	21,3	27,4	33,2	39,4	47,5	52,3	20,1	17,3	13,0	10,9	6,4	5,7
5	$[3n/4]$	9,2	10,1	11,2	12,4	14,3	16,2	24,0	23,4	22,1	20,5	17,4	15,0
6	n	4,1	4,7	4,9	5,0	5,3	6,9	27,2	26,9	26,0	24,5	22,4	21,7

Результаты вычислительных экспериментов на тестовых 3-планарных ПВ, полученные с помощью алгоритма β

№ серии	r	Доля задач, решаемых оптимально с помощью алгоритма β (в %) при n равном					
		150	200	250	300	350	400
1	2	92,0	94,7	97,3	98,6	99,0	99,4
2	$[\sqrt{n}]$	78,2	82,8	88,1	93,4	94,9	96,2
3	$[n/4]$	62,1	65,6	70,7	75,2	77,8	80,4
4	$[n/2]$	45,1	49,2	53,5	58,7	63,7	67,0
5	$[3n/4]$	31,2	35,0	41,2	44,7	47,8	50,1
6	n	11,2	13,6	15,4	17,8	19,7	20,6

Исходя из таблицы 1, можно сделать следующие два вывода:

- доля оптимальных решений 3-планарной ПВ порядка n, $n \geq 150$, находимых посредством алгоритма α , составляет не менее 65,5% , если $r=2$; не менее 46,1% , если $r = [\sqrt{n}]$; не менее 32,1%, если $r = [n/4]$; не менее 21,3%, если $r = [n/2]$, не менее 9,2%, если $r = [3n/4]$ и не менее 4,1%, если $r=n$, причем с ростом n их доля увеличивается;
- доля задач, для которых алгоритм α не строит плана 3-планарной ПВ составляет не более 12,0% , если $r=2$; не более 14,2% , если $r = [\sqrt{n}]$; не более 16,2%, если $r = [n/4]$; не более 20,1%, если $r = [n/2]$, не более 24,0%, если $r = [3n/4]$ и не более 27,2%, если $r=n$, причем с ростом n их доля уменьшается.

Данные, представленные в таблице 2, позволяют сделать следующий вывод: доля оптимальных решений 3-планарной ПВ порядка n, $n \geq 150$, находимых посредством алгоритма β , составляет не менее 92,0% , если $r=2$; не менее 78,2% , если $r = [\sqrt{n}]$; не менее 62,1%, если $r = [n/4]$; не менее 45,1%, если $r = [n/2]$; не менее 31,2%, если $r = [3n/4]$ и не менее 11,2%, если $r=n$, причем с ростом n их доля увеличивается.

Так как доля 3-планарных ПВ порядка n, решаемых оптимально с помощью алгоритмов α и β , увеличивается с ростом n, то данные алгоритмы почти всегда находят асимптотически оптимальный план и могут быть применены на практике для решения задач большой размерности.

Список литературы

- [1] A.M. Frieze *Complexity of a 3-dimensional assignment problem.* - European. J. Operat. Res., 1983, N13, P.161-164.
- [2] R.E. Burkard, K. Frohlich *Some remarks on 3-dimensional assignment problems.* Methods of OR, 1980, N36, P.31-36.

- [3] М.К. Кравцов, А.П. Крачковский *Полиномиальный алгоритм для многоиндексной проблемы выбора*. - Журнал вычислительной математики и математической физики, 1999, Т.39, №6, С.1041-1044.
- [4] М.К. Кравцов, А.П. Крачковский *Асимптотический подход к решению многоиндексной аксиальной транспортной задачи*. - Журнал вычислительной математики и математической физики, 1998, Т.38, №7, С.1133-1139.
- [5] М.К. Кравцов, А.П. Крачковский *Асимптотический подход к решению многоиндексной аксиальной проблемы выбора*. - Весті НАН Беларусі, сер. фіз.-мат. навук, 1999, N2, С.123-126.
- [6] М.К. Кравцов, А.П. Крачковский *Замечания к статье "Полиномиальный алгоритм для многоиндексной проблемы выбора"*. - Журнал вычислительной математики и математической физики, 2000, Т.40, №9, С.1440.
- [7] М.К. Кравцов, А.П. Крачковский *О полиномиальном алгоритме нахождения асимптотически оптимального решения трехиндексной планарной проблемы выбора*. - Журнал вычислительной математики и математической физики, 2001, Т.41, №2, С.342-345.
- [8] В.А. Емеличев, М.М. Ковалев, М.К.Кравцов *Многогранники, графы, оптимизация*. М.: Наука, 1981, 342 с.
- [9] Е.А. Диниц, М.А. Кронрод *Один алгоритм решения задачи о назначениях*. - Докл. АН СССР, 1969, Т.189, №1, С.23-25.

COMPUTING RESEARCH OF TWO APPROXIMATED ALGORITHMS FOR A THREE-INDEX PLANAR DECISION PROBLEM

M.K. Kravtsov, S.A. Dichkovskaya

*Economic Research Institute of the Economy Ministry, Minsk
e-mail: dichkovskaya@mail.ru*

Abstract. At some additional conditions on factors of target function are conducted the extensive computing research of two polynomial algorithms of the solution of a three-index planar decision problem.

Key words: the planar decision problem, the asymptotically optimum solution, the approximated solution

ДВА МЕТОДА ЛОКАЛЬНОГО ПОИСКА С ПАРАМЕТРАМИ ДЛЯ ЗАДАЧИ О МАКСИМАЛЬНОЙ КЛИКЕ

А.А. Кузнецова, О.Н. Карпачева

Ангарская Государственная Техническая Академия, Ангарск

e-mail: pm@agta.irmail.ru

Аннотация. В работе рассматривается комбинаторная NP-трудная задача о нахождении максимальной клики в неориентированном графе. Многие алгоритмы включают в себя процедуру поиска локально максимальной клики (ЛМК) или клики максимальной по включению в некотором подграфе исходного графа, при этом предпочтение отдается так называемым жадным алгоритмам. В этой статье предлагаются две процедуры локального поиска, являющиеся обобщением жадного алгоритма и основанные на нахождении локального решения в непрерывных задачах. Кроме того, приводится оценка снизу для найденной ЛМК и тестирование данных процедур на задачах из библиотеки DIMACS.

Ключевые слова: комбинаторная оптимизация, максимальная клика, локальный поиск.

1. Постановка задачи

Пусть дан неориентированный граф $G = G(V, E)$, где $V = \{1, \dots, n\}$ — множество вершин, $E \subset V \times V$ — множество ребер. Предположим, что граф является непустым, неполным и простым, т.е. не содержит петель и кратных ребер. Граф $\bar{G} = \bar{G}(V, \bar{E})$ называется дополнительным, если его множество вершин совпадает с V , а множество ребер дополняет исходный до полного, т.е. $\bar{E} = \{(i, j) \mid i \neq j, (i, j) \in E\}$.

Подмножество вершин C множества V называется кликой, если каждая пара вершин из C является смежной, т.е. соединена ребром в G . Будем в дальнейшем обозначать произвольную клику через C , а ее мощность через \mathcal{K} и введем для произвольной клики C мощности \mathcal{K} характеристический вектор $x(C) = \frac{1}{\mathcal{K}} \sum_{i \in C} e^i$, где e^i обозначает стандартный базисный вектор в пространстве R^n .

Известно, что каждому графу можно поставить в соответствие квадратную матрицу смежности $A = [a_{ij}]$ размерности n , в которой $a_{ij} = 1$, если $(i, j) \in E$, и $a_{ij} = 0$ в противном случае. Согласно сделанным предположениям матрица A будет симметричной, знакоопределенной с нулевыми диагональными элементами. Аналогично можно построить матрицу смежности $\bar{A} = [\bar{a}_{ij}]$ для дополнительного графа \bar{G} , которая будет обладать теми же свойствами, что и матрица A .

Рассмотрим следующие две квадратичные задачи:

$$F_\alpha(x) = \langle x, H_\alpha x \rangle \uparrow \max, \quad x \in S, \quad (P_\alpha)$$

$$F_\gamma(x) = \langle x, T_\gamma x \rangle \downarrow \min, \quad x \in S, \quad (D_\gamma)$$

где $H_\alpha = A + \alpha I_n$, $T_\gamma = \bar{A} + \gamma I_n$, I_n — единичная матрица размерности n , $S = \{x = (x_1, \dots, x_n)^T \mid \sum_{i=1}^n x_i = 1, x_i \geq 0, i = 1, \dots, n\}$ — канонический симплекс. Можно показать, что эти две задачи будут иметь одинаковые решения при $\alpha + \gamma = 1$. Положим $\mathcal{V}(P_\alpha) = \max(F_\alpha, S)$, $\mathcal{V}(D_\gamma) = \min(F_\gamma, S)$ — значения задач, $Sol(P_\alpha) = \text{Argmax}(F_\alpha, S)$, $Sol(D_\gamma) = \text{Argmin}(F_\gamma, S)$ — множества решений задач.

Предложение 1. Пусть $\alpha + \gamma = 1$. Тогда значения задач связаны тем же равенством, $\mathcal{V}(P_\alpha) + \mathcal{V}(D_\gamma) = 1$, и множества решений задач совпадают, $Sol(P_\alpha) = Sol(D_\gamma)$.

2. Процедуры локального поиска

Следующие две процедуры позволяют, начиная с произвольной точки $x^0 \in S$, находить характеристический вектор некоторой клики C , причем $F_\alpha(x^0) \leq F_\alpha(x(C))$, а $F_\gamma(x^0) \geq F_\gamma(x(C))$.

C_α -процедура

Шаг 0. Положить $\Omega := \{i : x_i^0 > 0\}$, $C := \emptyset$, $k := 0$.

Шаг 1. Если $\Omega = \emptyset$, то идти на шаг 5.

Шаг 2. Выбрать номер $m \in \Omega$ такой, что

$$(H_\alpha x^k)_m = \max_{j \in \Omega} (H_\alpha x^k)_j \quad (1)$$

Шаг 3. Построить множество $W_m = \{j \in \Omega : a_{jm} = 0\}$.

Шаг 4. Положить $\Omega := \Omega \setminus W_m \setminus \{m\}$, $C := C \cup \{m\}$, $x_m^{k+1} := x_m^k + \sum_{j \in W_m} x_j^k$, $x_j^{k+1} := 0$ для $j \in W_m$, $x_j^{k+1} := x_j^k$ для $j \in \Omega$, $k := k + 1$, и вернуться на шаг 1.

Шаг 5. Положить $\mathcal{K} := |C|$, и построить характеристический вектор $x(C)$ множества C .
Stop.

C_γ -процедура

Шаг 0. Положить $\Omega := \{i : x_i^0 > 0\}$, $C := \emptyset$, $k := 0$.

Шаг 1. Если $\Omega = \emptyset$, то идти на шаг 5.

Шаг 2. Выбрать номер $m \in \Omega$ такой, что

$$(T_\gamma x^k)_m = \min_{j \in \Omega} (T_\gamma x^k)_j \quad (2)$$

Шаг 3. Построить множество $W_m = \{j \in \Omega : \bar{a}_{jm} = 1\}$.

Шаг 4. Положить $\Omega := \Omega \setminus W_m \setminus \{m\}$, $C := C \cup \{m\}$, $x_m^{k+1} := x_m^k + \sum_{j \in W_m} x_j^k$, $x_j^{k+1} := 0$ для $j \in W_m$, $x_j^{k+1} := x_j^k$ для $j \in \Omega$, $k := k + 1$, и вернуться на шаг 1.

Шаг 5. Положить $\mathcal{K} := |C|$, и построить характеристический вектор $x(C)$ множества C .
Stop.

Справедливы следующие две теоремы.

Теорема 1. Пусть $x^0 \in S$, и $\alpha < 1$. Тогда для полученной C_α -процедурой клики C выполнено неравенство:

$$\mathcal{K} \geq \begin{cases} \left(\alpha/(1-\alpha) + F_\alpha(x^0) - \alpha \min_{i:x_i>0} x_i^0 \right)^{-1}, & \text{если } \alpha < 0; \\ \left(\alpha/(1-\alpha) + F_\alpha(x^0) - \alpha \max_{i:x_i>0} x_i^0 \right)^{-1}, & \text{если } 0 \leq \alpha < 1. \end{cases} \quad (3)$$

Теорема 2. Пусть $x^0 \in S$, и $\gamma \geq 0$. Тогда для полученной C_γ -процедурой клики C выполнено неравенство:

$$\mathcal{K} \geq \begin{cases} \left(F_\gamma(x^0) + (1-\gamma) \max_{i:x_i>0} x_i^0 \right)^{-1}, & \text{если } 0 \leq \gamma \leq 1; \\ \left(F_\gamma(x^0) + (1-\gamma) \min_{i:x_i>0} x_i^0 \right)^{-1}, & \text{если } \gamma > 1. \end{cases} \quad (4)$$

Покажем, что при $x^0 = (\frac{1}{n}, \dots, \frac{1}{n})$ и $\gamma = 0$ C_γ -процедура превращается в жадный алгоритм [2], и из формулы (4) при $\gamma = 0$ можно получить известную оценку для жадного алгоритма: $\mathcal{K} \geq \frac{n}{\bar{d}+1}$, где \bar{d} — средняя степень графа.

Действительно, при $x^0 = (\frac{1}{n}, \dots, \frac{1}{n})$ и $\gamma = 0$ формула (2) превращается в проверку условия:

$$d_m = \min_{j \in \Omega} d_j,$$

где $d_j = \sum_{i:x_i^m>0} \bar{a}_{ij}$. Подставляя $\gamma = 0$ и вид x^0 в формулу (4), получим:

$$\mathcal{K} \geq \left(F_0(x^0) + \max_{1 \leq i \leq n} x_i^0 \right)^{-1} = \left(\frac{1}{n} \sum_{i=1}^n \frac{d_i}{n} + \frac{1}{n} \right) = \frac{n}{\bar{d}+1},$$

$$\text{где } \bar{d} = \sum_{i=1}^n \frac{d_i}{n}.$$

3. Численный эксперимент

C_α -процедура была тестирована при $\alpha = 0.5$ в [3]. В качестве начальной точки последовательно для $i = 1, \dots, n$ выбирались точки следующего вида. Пусть $N_i = \{l \in V \mid a_{il}^G = 1\}$ — окрестность вершины i . Обозначим через $c_i^N = \sum_{j \in N_i} 1/d_j$, $c_i = c_i^N + 1/d_i$,

$$\lambda_i = \left(c_i^N + \frac{d_i + 0.5}{d_i} \right)^{-1}. \text{ Тогда}$$

$$x_j^i = \begin{cases} (1 + \lambda_i(d_i - 0.5)c_i^N)/(d_i c_i), & \text{если } j = i; \\ (1 - \lambda_i(d_i - 0.5)/d_i)/(d_j c_i), & \text{если } j \in N_i; \\ 0, & \text{в противном случае.} \end{cases}$$

Нетрудно видеть, что при таком выборе начальной точки $i \in C$, а условие (1) имеет вид:

$$\frac{\alpha}{d_m} + \sum_{s \in W_m} \frac{1}{d_s} = \max_{j \in \Omega} \left(\frac{\alpha}{d_j} + \sum_{s \in W_j} \frac{1}{d_s} \right).$$

Тогда можно построить следующую процедуру локального поиска. Пусть $N_{ij} = N_i \cap N_j$ — множество вершин, смежных с i и j .

C_α^{ij} -процедура

Шаг 0. Положить $\Omega := N_{ij} = N_i \cap N_j$, $C = \{i, j\}$, $\mathcal{K} := 2$.

Шаг 1. Если $\Omega = \emptyset$, то идти на шаг 5.

Шаг 2. Выбрать номер $m \in \Omega$ такой, что

$$\frac{\alpha}{d_m} + \sum_{s:(m,s) \in E} \frac{1}{d_s} = \max_{j \in \Omega} \left(\frac{\alpha}{d_j} + \sum_{s:(s,j) \in E} \frac{1}{d_s} \right).$$

Шаг 3. Построить множество $W_m = \{j \in \Omega \mid a_{mj}^G = 0\}$.

Шаг 4. Положить $\Omega = \Omega \setminus W_m \setminus \{m\}$, $C = C \cup \{m\}$, $\mathcal{K} := \mathcal{K} + 1$, и вернуться на шаг 1.

Шаг 5. Положить $\mathcal{K} := |C|$, и построить характеристический вектор $x(C)$ множества C .

Аналогично можно построить локальную процедуру для задачи (D_γ) .

C_γ^{ij} -процедура

Шаг 0. Положить $\Omega := N_{ij} = N_i \cap N_j$, $C = \{i, j\}$, $\mathcal{K} := 2$.

Шаг 1. Если $\Omega = \emptyset$, то идти на шаг 5.

Шаг 2. Выбрать номер $m \in \Omega$ такой, что

$$\frac{\gamma}{n - d_m - \lfloor \gamma \rfloor} + \sum_{s:(s,m) \in \bar{E}} \frac{1}{n - d_s - \lfloor \gamma \rfloor} = \max_{j \in \Omega} \left(\frac{\gamma}{n - d_j + \lfloor \gamma \rfloor} + \sum_{s:(s,j) \in \bar{E}} \frac{1}{n - d_s - \lfloor \gamma \rfloor} \right).$$

Шаг 3. Построить множество $W_m = \{j \in \Omega \mid \bar{a}_{mj}^G = 1\}$.

Шаг 4. Положить $\Omega = \Omega \setminus W_m \setminus \{m\}$, $C = C \cup \{m\}$, $\mathcal{K} := \mathcal{K} + 1$, и вернуться на шаг 1.

Шаг 5. Положить $\mathcal{K} := |C|$, и построить характеристический вектор $x(C)$ множества C .

Применяя C_α^{ij} - и C_γ^{ij} -процедуры для всех пар $(i, j) : (i, j) \in E$ можно получить два алгоритма поиска клики в графе. В C_α -алгоритме во множестве N_{ij} с помощью C_α^{ij} -процедуры находится локально максимальная клика и среди найденных клик находится клика максимальной мощности. Аналогично можно построить C_γ -алгоритм.

graph	n	dens	\mathcal{K}_*	\mathcal{K}	St	Time	
						C_α	C_γ
p_hat300-1	300	0.244	8	8	3	0:01.87	0:02.04
c-fat500-1	500	0.035	14	14	1	0:00.50	0:00.55
c-fat500-2	500	0.073	26	26	1	0:01.76	0:02.47
c-fat500-5	500	0.186	64	64	1	0:17.74	0:32.08
c-fat500-10	500	0.374	126	126	1	2:08.91	4:11.88
p_hat500-1	500	0.253	9	9	3	0:15.11	0:15.87
hamming8-4	256	0.639	16	16	1	0:43.83	0:43.07
p_hat300-2	300	0.489	25	25	6	0:46.31	0:44.87
san400_0.5_1	400	0.500	13	13	3	3:00.81	2:44.94
sanr400_0.5	400	0.501	13	13	4	1:38.10	1:38.20
DSJC500.5	500	0.502	13	13	4	3:50.03	3:47.34
p_hat500-2	500	0.505	36	36	7	7:09.68	6:48.32
hamming8-2	256	0.968	128	128	1	7:30.28	6:37.11
p_hat300-3	300	0.744	36	36	12	3:52.34	3:34.27
MANN_a27	378	0.990	126	126	2	42:31.02	37:59.19
san400_0.7_1	400	0.700	40	40	9	10:37.03	9:13.26
san400_0.7_2	400	0.700	30	30	6	9:42.60	8:58.22
san400_0.7_3	400	0.700	22	22	6	9:26.07	8:44.82
sanr400_0.7	400	0.700	≥ 21	21	4	8:51.02	8:20.42
brock400_1	400	0.748	27	25	4	12:24.30	11:31.63
brock400_2	400	0.749	29	29	5	12:27.65	11:34.75
brock400_3	400	0.748	31	31	5	12:21.33	11:29.15
brock400_4	400	0.749	33	33	4	12:26.71	11:34.09
san400_0.9_1	400	0.900	100	100	6	32:19.47	28:49.05
p_hat500-3	500	0.752	≥ 50	50	5	34:42.23	30:17.10

Результаты тестирования C_α и C_γ -алгоритмов на 25 DIMACS benchmark graphs

C_α - и C_γ -алгоритмы были тестированы при $\alpha, \gamma = 0.5$ на задачах из библиотеки DIMACS. Результаты тестирования приведены в следующей таблице. Здесь *graph* — название файла, содержащего матрицу смежности графа; n — размерность графа; *dens* — плотность графа, т.е. отношение количества ребер к максимально возможному C_n^2 при данной размерности. Далее, \mathcal{K}_* и \mathcal{K} — размерности максимальной клики и клики, найденной обоими алгоритмами (полученные результаты для C_α - и C_γ -алгоритмов совпали). *St* — количество изменений \mathcal{K} на шаге 4. Наконец, Time — время решения алгоритмами на PC Celeron (CPU 433).

Список литературы

- [1] *Bomze I.M., Budinich M., Pardalos P.M., Pelillo M.* The maximum clique problem // Handbook of Combinatorial Optimization. – Dordrecht: Kluwer. 1999. – Suppl. Vol. A. – P.1 – 74.
- [2] *Johnson D.S.* Approximation Algorithms for Combinatorial Problems // J. Computer. and System Science. 1974. V. 9. P. 256–278.
- [3] *Kuznetsova A., Strekalovsky A.* On Solving the maximum clique problem // J. of Global Optimization. 2001. V. 21. \ N. 3. P. 265–288.

TOW PARAMETRIC LOCAL SEARCH METHODS FOR THE MAXIMUM CLIQUE PROBLEM

A.A. Kuznetsova, O.N. Karpacheva

Angarsk State Technical Academy, Angarsk

Abstract. In this work we consider the maximum clique problem. It is a well-known NP-hard combinatorial optimization problem. There are many algorithms based on the local search algorithm. The common one is the greedy algorithm. We suggest two local searches, which generalize the greedy algorithm and they are based on some continuous problems. These algorithms also provide us with lower bounds. The efficiency of this approach is proved on the DIMACS test instances.

Key words: combinatorial optimization, maximum clique, local search.

О СХОДИМОСТИ ОДНОГО АЛГОРИТМА МУРАВЬИНОЙ КОЛОНИИ ДЛЯ ЗАДАЧИ О p -МЕДИАНЕ¹

Т.В. Леванова

Омский филиал института математики СО РАН, Омск
e-mail: levanova@iitam.omsk.net.ru

М.А. Лореш

Омский государственный университет, Омск
e-mail: loresh@bk.ru

Аннотация. В статье изучается алгоритм муравьиной колонии для решения задачи о p -медиане на минимум. Найдены условия на коэффициент испарения феромона, при которых текущий рекорд стремится к оптимуму почти наверное, а для алгоритма искусственного муравья вероятность получить точное решение задачи стремится к единице с ростом числа итераций алгоритма.

Ключевые слова: методы оптимизации, p -медиана, алгоритм муравьиной колонии, сходимость.

Введение

На практике часто возникают ситуации, в которых необходимо наилучшим образом разместить предприятия бытового обслуживания, пункты автосервиса на дорогах, подстанции в электросетях, коммутаторы в телефонной сети и т.д. Значительная их часть может быть описана с помощью оптимизационных моделей и сведена к решению дискретных задач размещения. Однако вычислительная сложность многих задач не позволяет получать оптимальное решение за приемлемое время. В связи с этим особую значимость приобрела разработка методов получения приближенных решений. В последние годы особый интерес проявляется к алгоритмам, идеи которых заимствованы у живой природы или из физических процессов. К таким методам можно отнести алгоритмы муравьиной колонии, генетические алгоритмы, поиск с запретами, алгоритм имитации отжига, нейронные сети. Указанные алгоритмы, как правило, являются вероятностными итерационными методами, поэтому возникает необходимость в изучении их асимптотических свойств.

В данной работе предлагается один алгоритм муравьиной колонии для задачи о p -медиане на минимум, изучается его асимптотическая сходимость к оптимальному решению. Вопросы сходимости алгоритмов муравьиной колонии исследуются также в статье [1]. В ней найдены условия, при которых текущий рекорд стремится к точному решению с вероятностью единица. Однако автором рассматриваются лишь алгоритмы муравьиной колонии для задач на графах – так называемые *Graph-based Ant System* (к таким задачам можно отнести, например, задачу коммивояжера). К тому же в статье используется иная схема использования и переопределения феромона, что делает результаты, полученные в [1], не применимыми для предложенного в данной работе алгоритма муравьиной колонии.

¹Работа поддержана грантом Федерального агентства по образованию РФ (проект А04-2.8-581) и грантом РГНФ (проект 04-02-00238а).

1. Постановка задачи

Дано конечное множество пунктов возможного размещения предприятий и список клиентов. Предприятия производят однородный продукт в неограниченном количестве. Известны затраты на обслуживание каждого клиента, затраты на открытие предприятия в каждом пункте равны нулю. Требуется разместить ровно p предприятий и прикрепить к ним клиентов так, чтобы спрос каждого клиента был удовлетворен и суммарные затраты на обслуживание были минимальны.

Введем обозначения:

m – число пунктов возможного размещения предприятий;

i – номер пункта возможного размещения предприятия, $i \in I = \{1, \dots, m\}$;

n – количество клиентов, спрос которых необходимо удовлетворить;

j – номер клиента, $j \in J = \{1, \dots, n\}$;

t_{ij} – затраты на обслуживание клиента j предприятием, размещенным в пункте i .

Переменные

$z_i = 1$, если предприятие открыто в пункте i , и 0 в противном случае;

$x_{ij} = 1$, если предприятие в пункте i обслуживает клиента j , и 0 в противном случае.

Если предприятие открыто в пункте i , то будем говорить, что предприятие i открыто. Также будем придерживаться следующих обозначений: $z = (z_i)$, $x = (x_{ij})$, $i \in I$, $j \in J$. Тогда модель целочисленного линейного программирования для данной задачи имеет вид:

$$F(z, x) = \sum_{i \in I} \sum_{j \in J} t_{ij} x_{ij} \rightarrow \min \quad (1)$$

$$\sum_{i \in I} x_{ij} = 1, \quad j \in J, \quad (2)$$

$$\sum_{i \in I} z_i = p, \quad (3)$$

$$x_{ij} \leq z_i, \quad i \in I, \quad j \in J, \quad (4)$$

$$z_i, x_{ij} \in \{0, 1\}, \quad i \in I, \quad j \in J. \quad (5)$$

Условия (2) и (3) означают, что потребности каждого клиента должны быть удовлетворены и необходимо открыть ровно p предприятий. Неравенства (4) гарантируют, что обслуживание будет осуществляться только открытыми предприятиями.

Известно, что задача о p -медиане NP -трудна [2], поэтому одним из актуальных направлений является разработка эвристических алгоритмов для ее решения.

2. Алгоритм муравьиной колонии

Алгоритмы муравьиной колонии (МК) принадлежат к классу приближенных вероятностных алгоритмов. Алгоритмы МК моделируют поведение живых муравьев [3], способных находить кратчайший путь от муравейника до источника пищи при помощи специального пахнущего вещества – феромона. Искусственный муравей представляет собой рандомизированный жадный алгоритм. На каждом шаге алгоритма МК определенное число муравьев накапливает информацию о решениях (феромон), которая обрабатывается и используется в дальнейшем.

В статьях [4] и [5] предложены варианты алгоритмов муравьиной колонии для задачи о p -медиане на минимум, описаны экспериментальные исследования, подтверждающие

перспективность подхода. В данной работе изучаются теоретические вопросы сходимости алгоритма муравьиной колонии. Исследуется один алгоритм муравьиной колонии (АСС) с измененной по сравнению с [4] и [5] схемой, для которой удается доказать асимптотическую сходимость к точному решению. Опишем алгоритм АСС.

В следствие того, что по фиксированным значениям z_i легко восстанавливаются значения x_{ij} , решением задачи о p -медиане будем называть вектор z , удовлетворяющий ограничениям (3) и (5). Феромон для каждого потенциального предприятия i на итерации k будет храниться в векторе $\alpha^k = (\alpha_i^k), i \in I$.

В алгоритме АСС искусственный муравей (ИМ) представляет собой вероятностную модификацию жадного алгоритма спуска. Пусть \hat{I} – множество открытых предприятий; Δf_i – изменение целевой функции в результате закрытия предприятия i . Так как при закрытии какого-либо предприятия значение целевой функции f не убывает, то $\Delta f_i \geq 0$ для всех $i \in \hat{I}$. Алгоритм ИМ начинает работу с множества $\hat{I} = I$ и завершает ее при $|\hat{I}| = p$. На каждом шаге с вероятностью

$$p_i = \frac{\alpha_i^k (\Delta f_{\max} - \Delta f_i + \varepsilon)}{\sum_{t \in \hat{I}} \alpha_t^k (\Delta f_{\max} - \Delta f_t + \varepsilon)}, \quad i \in \hat{I}, \quad (6)$$

из множества \hat{I} выбирается одно предприятие i_0 , которое необходимо закрыть. Здесь $\Delta f_{\max} = \max_{t \in \hat{I}} \Delta f_t$; параметр $\varepsilon > 0$. Параметр ε гарантирует, что любое открытое предприятие может быть закрыто с ненулевой вероятностью. Опишем схему алгоритма ИМ.

Алгоритм ИМ

0. $\hat{I} = I$.

Шаг $r, r \geq 1$.

1. Если $|\hat{I}| = p$, то конец работы алгоритма.
2. С вероятностью (6) выбираем элемент $i_0 \in \hat{I}$.
3. Переопределяем $\hat{I} = \hat{I} \setminus \{i_0\}$.

Переходим на следующий шаг.

Тогда схема алгоритма АСС имеет следующий вид.

Алгоритм АСС

0. Определяем начальный вектор статистической информации α^1 и начальный рекорд $F = \infty, z^*$ – лучшее найденное решение.

Итерация $k, k \geq 1$.

1. Строим s допустимых решений алгоритмом ИМ.
2. Выбираем среди них лучшие по целевой функции;
 \hat{f} – рекорд на этой итерации, \hat{z} – решение, соответствующее \hat{f} .
3. Переопределяем значения $\alpha_i^k, i \in I$.
4. Если $\hat{f} < F$, то $F = \hat{f}$, и $z^* = \hat{z}$.
5. Для ненулевых компонент рекордного решения z^* полагаем $\alpha_i = \alpha_{\min}$.
6. Если выполняется критерий остановки, то конец работы алгоритма.

Переходим на следующую итерацию.

На шаге 2 каждой итерации алгоритма лучшими считаются только те решения, для которых значения целевой функции строго меньше рекорда F . Входной параметр α_{\min} ($\alpha_{\min} > 0$) задает минимально возможное значение $\alpha_i^k, i \in I$. На итерации k компоненты

вектора α изменяются следующим образом:

$$\alpha_i^{k+1} = \frac{\alpha_{\min} + q^{\gamma_i}(\alpha_i^k - \alpha_{\min})}{\beta_k}, \quad i \in I, \quad (7)$$

где $\beta_k \in (0, 1)$ – коэффициент затухания (испарения феромона) на итерации k , $\gamma_i \in [0, 1]$ – частота появления предприятия i в лучших решениях, выбираемых на шаге 2, параметр $q \in (0, 1)$. Таким образом, при данных значениях параметров β_k и q , чем чаще предприятие i попадает в лучшие решения, тем меньше становится соответствующее значение α_i^k , $i \in I$. Т.е. чем более выгодным представляется предприятие i , тем меньше уровень феромона для этого предприятия, и наоборот – чем хуже предприятие, тем больше соответствующее значение феромона.

3. Асимптотические свойства

Определение. Решение z будем называть просмотренным, если оно было найдено в процессе работы алгоритма АСС хотя бы одним муравьем хотя бы на одной итерации.

Пусть z^k – рекордное решение, найденное алгоритмом АСС к началу итерации k . Тогда справедливо следующее утверждение.

Теорема. Пусть в алгоритме АСС для некоторого $N \geq 1$ выполняется условие

$$\sqrt[m-p]{\frac{k+1}{k+2}} \leq \beta_k < 1 \quad \text{для всех } k \geq N.$$

Тогда $F(z^k) \rightarrow F(z^*)$ почти наверное при $k \rightarrow \infty$, где z^* – оптимальное решение.

Доказательство. Без ограничения общности можно считать, что условия теоремы выполняются, начиная с $k = 1$. Обозначим через F_k такое событие, что итерация k – это первая итерация, на которой алгоритмом АСС был просмотрен какой-либо оптимум. Рассмотрим фиксированное оптимальное решение z^* . Если выполняется событие $\neg F_1 \wedge \neg F_2 \wedge \dots$, то решение z^* никогда не будет просмотрено, и, следовательно, вероятность

$$\begin{aligned} & P(\neg F_1 \wedge \neg F_2 \wedge \dots) \leq P(z^* \text{ никогда не будет просмотрено}) = \\ & = \prod_{k=1}^{\infty} P(z^* \text{ не просмотрено на итерации } k | z^* \text{ не просмотрено на итерации } t < k). \end{aligned} \quad (8)$$

Теперь для каждого предприятия $i = 1, \dots, m$ можно построить нижнюю и верхнюю границы для соответствующего уровня феромона в начале итерации k . Из схемы алгоритма следует, что $\alpha_i \geq \alpha_{\min}$ для всех i . Верхняя граница уровня феромона достигается, если за все предыдущие итерации предприятие ни разу не попало в лучшие решения, и феромон только испарялся (т.е. все время происходило деление на β_k). Следовательно, в силу (7)

$$\alpha_i^k \leq \alpha_i^1 \prod_{t=1}^{k-1} \frac{1}{\beta_t} \leq \alpha_i^1 \prod_{t=1}^{k-1} \sqrt[m-p]{\frac{t+2}{t+1}} = \alpha_i^1 \frac{\sqrt[m-p]{k+1}}{\sqrt[m-p]{2}} = \text{const} \cdot \sqrt[m-p]{k+1}.$$

Тогда, для всех $i = 1, 2, \dots, m$ выполняется $\alpha_{\min} \leq \alpha_i^k \leq \text{const} \cdot \sqrt[m-p]{k+1}$.

Следует отметить, что решение z задачи о p -медиане можно однозначно определить не только списком, состоящим из p открытых предприятий, но и множеством $m-p$ закрытых предприятий. Поэтому в дальнейшем будем говорить, что допустимое решение задачи –

это множество закрытых предприятий. Пусть $\bar{I}^* = \{i_1, i_2, \dots, i_{m-p}\}$ – список (множество) предприятий, которые необходимо закрыть, чтобы получить решение z^* . Обозначим через A_r^k событие, что на итерации k алгоритмом ИМ на шаге r закрыто одно из предприятий i_1, i_2, \dots, i_{m-p} , и через \hat{I}_r – множество незакрытых предприятий перед началом шага r алгоритма ИМ. Тогда, вероятность того, что искусственный муравей найдет решение z^* на итерации k ,

$$P^k(z^*) = P(A_1^k \wedge A_2^k \wedge \dots \wedge A_{m-p}^k) = \prod_{r=1}^{m-p} \frac{\sum_{i \in \bar{I}^* \cap \hat{I}_r} \alpha_i^k (\Delta f_{\max} - \Delta f_i + \varepsilon)}{\sum_{i \in \hat{I}_r} \alpha_i^k (\Delta f_{\max} - \Delta f_k + \varepsilon)} \geq \frac{1}{C_m^p} \left(\frac{\text{const}}{\sqrt[m-p]{k+1}} \right)^{m-p}.$$

Очевидно, что нижняя граница для вероятности не зависит от того, что происходило перед итерацией k , поэтому оценка также выполняется на итерациях $1, \dots, k-1$. И, следовательно, верхняя граница для (8) имеет вид

$$P(\neg F_1 \wedge \neg F_2 \wedge \dots) \leq \prod_{k=1}^{\infty} \left[1 - \frac{1}{C_m^p} \cdot \frac{\text{const}}{k+1} \right]. \quad (9)$$

Логарифмируя правую часть неравенства (9), получаем

$$\sum_{k=1}^{\infty} \log \left[1 - \frac{1}{C_m^p} \cdot \frac{\text{const}}{k+1} \right] \leq - \sum_{k=1}^{\infty} \frac{1}{C_m^p} \cdot \frac{\text{const}}{k+1} = -\infty,$$

Следовательно, правая часть неравенства (9), а значит и (8), равны нулю. Это доказывает, что событие $F_1 \vee F_2 \vee \dots$ (на некоторой итерации оптимальное решение будет просмотрено) имеет вероятность, равную единице. Последнее доказывает утверждение теоремы.

Обозначим через $P^k(z)$ вероятность того, что решение z будет найдено алгоритмом ИМ на итерации k алгоритма АСС. Пользуясь доказанной теоремой, нетрудно получить

Следствие. Если в алгоритме АСС для некоторого $N \geq 1$ выполняется условие

$$\beta_k = \sqrt[m-p]{\frac{k+1}{k+2}} \quad \text{для всех } k \geq N,$$

то для каждого запуска алгоритма АСС найдется такое оптимальное решение z^* , что для алгоритма ИМ имеет место сходимость $P^k(z^*) \rightarrow 1$ при $k \rightarrow \infty$.

Список литературы

- [1] W. Gutjahr *ACO Algorithms with Guaranteed Convergence to the Optimal Solution*. – Information Processing Letters 82, 2002, pp. 145-153.
- [2] O. Kariv, S.L. Hakimi *An Algorithmic Approach to Network Location Problems*. – SIAM J. Appl. Math. 1979. V. 37, N3, pp. 513 - 560.
- [3] M. Dorigo, V. Maniezzo, A. Colorny *Ant System: An Autocatalytic Optimizing Process*. – Report No. TR-91-016. – Milan: Politecnici di Milano, 1991.

- [4] Т.В. Леванова, М.А. Лореш *Алгоритмы муравьиной колонии и имитации отжига для задачи о p -медиане.* – Автоматика и телемеханика, №3, 2004, с. 80-88.
- [5] T.V. Levanova, M.A. Loresh *Ant colony and simulated annealing algorithms for the p -median problem.* – Proceedings of The Second International Workshop "Discrete Optimization Methods in Production and Logistics", Omsk-Irkutsk, 2004, pp. 70-74.

ON CONVERGENCE OF ONE ANT COLONY ALGORITHM FOR THE p -MEDIAN PROBLEM

T.V. Levanova

*Omsk Branch of Sobolev Institute of Mathematics SB RAS, Omsk
e-mail: levanova@iitam.omsk.net.ru*

M.A. Loresh

*Omsk State University, Omsk
e-mail: loresh@bk.ru*

Abstract. In this paper we consider the classical p -median minimization problem. It is well-known that this problem is NP-hard. As approximative method of solving p -median problem we propose a variant of the Ant Colony algorithm, the Ant Colony Converging algorithm (ACC). It is shown that for some conditions on the evaporation factor the ACC algorithm converges to optimal solution with probability one.

Key words: discrete optimization, p -median problem, ant colony optimization, convergence.

АЛГОРИТМЫ ДЕКОДИРОВАНИЯ ПРЯМОУГОЛЬНЫХ УПАКОВОК: КРАТКИЙ ОБЗОР СОВРЕМЕННЫХ ПОДХОДОВ.

Э. А. Мухачева, А. С. Филиппова, Ю. И. Алентьева

Уфимский Государственный Авиационный Технический Университет
e-mail: elita@vmtk.ugatu.ac.ru

Аннотация. В статье рассматриваются задачи прямоугольной упаковки в различных постановках. Для них известны и разрабатываются новые способы кодирования входной информации, на основании которой алгоритмы-декодеры проектируют допустимые упаковки и их эскизы. Декодеры представляют однопроходные эвристики полиномиальной сложности. В настоящее время развиваются два основных направления разработки декодеров: "нижний-левый" (Европа) и "парных последовательностей" (Япония). В статье приведен "блочный" подход, который включен авторами в оба направления.

Ключевые слова: раскрой и упаковка, NP-трудные задачи, оптимизация, декодеры, пары последовательностей, блочная структура, вычислительная сложность.

1. Введение

В качестве основной принято рассматривать следующую **задачу упаковки, Packing Problem, PP**: имеются малые элементы, их необходимо разместить без взаимного перекрытия внутри больших объектов так, чтобы заданная целевая функция достигла минимума (максимума). В категории «задач раскроя и упаковки» содержится множество прикладных задач, которые изучаются с давних пор. Большая часть таких задач являются NP-трудными комбинаторными проблемами и для них не существует точных методов решения полиномиальной сложности. Классификация задач раскроя и упаковки выполнена Н. Дукhoff [1]. Одним из важных факторов разделения задач на классы является *мерность* элементов и объектов. Различаются одно, двух, трех и n-мерные задачи. Другим фактором является способ разделения малых элементов, расположенных внутри объектов. Различаются задачи *гильотинной* и *негильотинной* упаковки. Здесь рассматриваются только задачи **негильотинной прямоугольной упаковки, 2 Dimensional Rectangular Packing Problem, 2DRPP**. Среди них выделяются задачи по факторам задания объектов и целевой функции.

1. Задача упаковки прямоугольников в полосу, Strip Packing Problem, 2DSPP. Один из размеров полосы, например, ширина W , задан, второй - является переменным. Требуется найти упаковку в полосу минимальной длины L .

2. Задача упаковки прямоугольников в контейнеры, Bin Packing Problem, 2DBPP. Оба размера, ширина W и длина L контейнера заданы. Требуется найти минимальное количество контейнеров, в которые упакованы все малые прямоугольные элементы.

3. Задача упаковки прямоугольников в открытую область, Area Packing Problem, 2DAPP. Оба размера, ширина W и длина L - являются для прямоугольного объекта переменными. Требуется найти упаковку элементов в угол, для которой площадь огибающего элементы прямоугольника достигает минимума.

Подробно мы остановимся на задаче **2DSPP**. Для того чтобы проектировать алгорит-

мы упаковки, мы должны определить и выполнить следующие компоненты:

- **кодирование схемы упаковки** (представление решения).
- **алгоритм декодирования** (получение упаковки из кодированного решения).
- **стратегия поиска** (поиск хорошего кодированного решения).

2. Кодирование схемы упаковки

2.1. Прямая схема кодирования. Зададим положение каждого прямоугольника P_i , $i=1, \dots, t$ вектором $(x_i; y_i)$ с минимальными координатами. Последовательность таких векторов представляет прямую схему кодирования упаковки. Область решений такой кодировочной схемы - бесконечное множество. Поиск подмножества $\{x; y\}$ без взаимного перекрытия прямоугольников представляет нетривиальную проблему. Вместе с тем, если прямая схема кодирования обеспечивает неперекрывание прямоугольников, то восстановить упаковку по этой схеме не представляет труда. Поэтому разрабатываются схемы кодирования, снабженные декодерами - алгоритмами перехода от заданной схемы к прямой.

2.2. Кодирование перестановкой прямоугольников. Список $p = \{1(p), 2(p), \dots, i(p), \dots, t(p)\}$, $i(p)$ - номер прямоугольника, занимающего в p позицию i , - представляет наиболее популярную схему кодирования. С помощью того или иного алгоритма размещения (декодера) вычисляют координаты $(x_i; y_i)$ прямоугольника P_i и строят эскиз упаковки. Её длина зависит от перестановки p и от используемого декодера. Каждому списку p (в отличие от прямой схемы) соответствует несколько различных упаковок. Однозначный выбор схемы упаковки осуществляет декодер.

2.3. Кодирование парой последовательностей. Murata и др. [2] предложили схему кодирования, названную парой последовательностей, **Sequence Pair, SP**. Для представления решения используется пара перестановок σ_+ и σ_- из t прямоугольников. Базируясь на кодированном этими последовательностями решении, устанавливают относительные положения для каждой пары i и j прямоугольников: если i располагается перед j в обеих перестановках, следует разместить i левее j . Если i располагается перед j в σ_+ и после j в σ_- , то i помещаем выше j . Например, для пары последовательностей σ_+ : 6, 3, 1, 4, 5, 2; σ_- : 1, 2, 3, 4, 6, 5 имеем: прямоугольник P_1 расположен перед P_4 в обеих перестановках, следовательно, P_1 левее P_4 ; прямоугольник P_4 располагается перед P_2 в σ_+ и после P_2 в σ_- , и следовательно P_4 - выше чем P_2 , и так далее. Есть и другие схемы кодирования, которые описывают относительные местоположения для каждой пары прямоугольников.

2.4. Кодирование блок-структурами. Пусть имеется прямоугольная упаковка RP . Проведем через правые стороны прямоугольников вертикальные резы. Они разбивают RP на прямоугольные вертикальные блоки одной и той же ширины W и различной длины z_j . Пусть длина RP равна L . Проведем через верхние стороны прямоугольников горизонтальные линии. Тогда RP разобьется на горизонтальные блоки одной и той же длины L и различной ширины s_j , причем $\sum_j s_j \leq W$. Так мы получаем две блок-структуры для RP , вертикальную и горизонтальную. Каждому блоку j соответствует кортеж S_j (запись номеров прямоугольников, пересекающих блок) и длина z_j вертикального, ширина s_j горизонтального блоков.

Например, списки $S = \{(6, 3, 1)z_1 (6, 1)z_2 (6, 4, 2)z_3 (5, 2)z_4\}$;

$$\tilde{S} = \{(1, 2)_{s_1} (1, 4, 5)_{s_2} (3, 4, 5)_{s_3} (3, 5)_{s_4} (6, 5)_{s_5} (6)_{s_6}\};$$

отвечают приведенной ранее паре последовательностей σ_+ и σ_- . Для преобразования блок-структур S и \tilde{S} в пару σ_+ и σ_- выписывают в порядке сверху-вниз номера прямоугольников из первого блока; из следующих «определяющих» блоков записывают номера начатых или законченных в них элементов. В настоящее время разрабатывается алгоритм обратной связи кода SP к кодированию блок-структурами S и \tilde{S} .

3. Алгоритмы декодирования

Применяя алгоритм-декодер к коду упаковки, получают схему упаковки. В Европе наибольшую популярность имеет декодер **нижний-левый**, **Bottom-Left**, **BL** и его различные модификации. Основная стратегия BL состоит в упаковке следующего элемента из перестановки p в самую нижнюю возможную позицию, выравнивая по левому краю. В. Baker и др. [3] проанализировали производительность алгоритма для «наихудшего» случая и доказали, что: если не использовать упорядочивание элементов, BL может оказаться плохим; если элементы упорядочены по невозрастанию ширины, то $BL(I) = 3 \cdot OPT(I)$. J. Berkey, P. Wang [4] предложили BL для случая ограниченного контейнера. Их алгоритм «ограниченный нижний-левый» (FBL) вначале сортирует элементы по невозрастанию ширины. Затем текущий элемент упаковывается на самую нижнюю позицию любого инициализированного контейнера, с выравниванием по левому краю. Компьютерная реализация алгоритма BL сложности $O(m^2)$ была предложена В. Chazelle [5]. Метод нижний-левый до сих пор привлекает внимание ученых. В рамках реализации BL метода можно выделить следующие основные проблемы: как найти самую нижнюю возможную позицию; как выбрать элемент для упаковки в найденную позицию; как упаковать в найденную позицию выбранный элемент.

В. Chazelle [5] находит самую нижнюю позицию. Чиглинцев А. В. и Мухачева-Филиппова А. С. [6] предложили для её поиска блочный метод, основанный на блочной структуре кодирования. Этот метод дает те же результаты, что и алгоритм В. Chazelle, но он удобнее и быстрее. Далее, когда появились и начали бурно развиваться метаэвристики, стали применять простые методы ортогонального поиска нижней-левой позиции. S. Jakobs [7] предложил ортогональный вариант BL, основанный на BL-условии: *упаковка соответствует BL-условию*, если прямоугольник не может быть сдвинут дальше вниз и влево. BL-алгоритм состоит в последовательном передвижении прямоугольника, начиная от верхнего правого угла доски так далеко, как возможно вниз и потом так далеко, как возможно влево. D. Liu, H. Teng [8] обнаружили, что некоторые лучшие схемы раскроя не могут быть сконструированы BL-алгоритмом. Ими был предложен усовершенствованный алгоритм (IBL), который состоит в следующем: последовательно передвинуть прямоугольник, начиная от верхнего правого угла упакованной площади доски вниз и влево и передвижение вниз имеет преимущество. А именно влево движение продолжается пока не возникнет возможность двигаться вниз. Сложность усовершенствованного BL алгоритма не увеличилась. D. Smith [9] разработал подход, экспериментирующий с двумя эвристическими декодирующими операциями. Первая - «скольжение», помещает прямоугольник в угол, из которого он падает к противоположному дальнему углу, совершая ортогональные перемещения зигзагами в устойчивую позицию. Вторая процедура - «горизонт» - пробует все устойчивые позиции в частичном размещении. Эта комбинация генерирует лучшие размещения, но в вычислительном отношении более дорога. E. Hopper и H. Turton применяли полную программу размещения с сохранением BL-стабильности [10].

На базе кодирования упаковок парой последовательностей, SP, известно несколько эффективных алгоритмов. Их сложность от $O(m^2)$ до $O(m \cdot \log(m))$. Подробный обзор алгоритмов приведен S. Imahori (2004) в докторской диссертации Высшей Школы Информатики, Киото. В настоящее время разрабатывается блочный декодер на базе пары последовательностей SP, он совмещает в себе полный охват информации SP и мобильность блочных структур.

Выбор элемента для упаковки в найденную позицию почти все авторы осуществляют на основании перестановки p , используя в ней стратегию «первый подходящий», **First Fitting, FF**. В метаэвристиках перестановки прямоугольников меняются на каждой итерации. Норенков И. П. [11] предложил алгоритм локального поиска, оперируя со списком алгоритмов, эффективность которых оценивается различными критериями. Что касается списка p то в нем осуществляется рандомизированный перебор прямоугольников. Таким образом, список p сокращается по мере генерации упаковки. Ширгазиным Р. Р. и Мухачевой-Филипповой А. С. [12] предложен жадный алгоритм для плотной упаковки в свободную область нескольких элементов.

Выбор положения прямоугольника в свободной области представляет далеко не тривиальную задачу. В западных работах прямоугольник располагается с выравниванием слева и до упора вниз. Однако возможны и другие положения, при которых на следующих итерациях удастся частично разместить в той же области другой прямоугольник. Гибридный алгоритм основан на применении двух эвристик:

- *сравнение*: перед размещением прямоугольника сравниваются возможные потери площади выше и ниже прямоугольника, перемещение делается в сторону меньшей площади; увеличивается вероятность того, что в большей области смогут размещаться другие предметы;
- *сдвиг*: в зависимости от положения прямоугольника относительно средней линии полосы он сдвигается к ближайшему краю; таким образом, прямоугольники притягиваются к краям полосы, образуя в середине пустое пространство, в которое можно разместить другие элементы из списка p .

Мы рассмотрели основные направления, развиваемые для разработки эффективных эвристик. Это различные варианты алгоритма нижний-левый, блочного алгоритма и кодирование парой последовательностей. Все эти и другие алгоритмы - декодеры применяются в составе методов локального поиска оптимального решения, представляющих различные стратегии поиска лучшего решения. Дальнейшее повышение эффективности алгоритмов ведется в направлении повышения эффективности функции цели и сокращения времени расчета. Это удастся сделать за счет улучшающих алгоритмы приемов, гибридизации алгоритмов и снижения вычислительной сложности.

Список литературы

- [1] Dykhoff H. A. *Typology of cutting and packing problems*. European Journal of Operational research. Vol. 44. 145-149.
- [2] Murata H., Fujiyoshi K., Nakatane S., Kajitani Y. *Rectangle-Packing-Based Module Placement*. Prof. IEEE/ACM International Conf. on Computer-Aided Design. 1995. 472-479.
- [3] Baker B. S., Coffman E. G., Rives R. L. *Orthogonal Packing In Two Dimensions*. SIAM Journal on Computing. 9, 1980. 846-855.
- [4] Berkey J. O., Wang P. Y. *Two dimensional finite bin packing algorithms*. Journal of the Operational Research Society. 38, 1987. 423-429.
- [5] Chazelle B. *The bottom-left bin packing heuristic: An efficient implementation*. IEEE Transactions on Computers. 32, 1983. 697-707.
- [6] Мухачева А. С., Чиглинцев А. В., Смагин М. А., Мухачева Э. А. *Задачи двумерной упаковки: развитие генетических алгоритмов на базе смешанных процедур локального поиска оптимального решения*. Информационные технологии. Приложение. 5, 2001. 30-37.
- [7] Jakobs S. *On the genetic algorithms for the packing of polygons*. European Journal of Operational Research. 88 1996. 165-181.
- [8] Liu D., Teng H. *An improved BL-algorithm for genetic algorithm of the orthogonal packing of rectangles*. European Journal of Operation Research. 1999. Vol. 112. 413-420.
- [9] Smith D. *Bin-packing with adaptive search*. In Grefenstette(ed) Proceedings of an International Conference on Genetic Algorithms and their Applications. 1985. 202-206.
- [10] Hopper E., Turton B. C. H. *An Empirical Investigation of Meta-heuristic and Heuristic Algorithms for 2D Packing Problem*. European Journal of Operational Research. 128. 1, 2000. 34-57.
- [11] Норенков И. П. *Эвристики и их комбинации в генетических методах дискретной оптимизации*. Информационные технологии. 1, 1999. 2-7.
- [12] Мухачева А. С., Ширгазин Р. Р. *Задачи упаковки прямоугольников: рандомизированная эвристика на базе двойственной схемы локального поиска оптимума*. Информационные технологии. 5, 2003. 18-23.

DECODING ALGORITHMS FOR RECTANGULAR PACKING: A SHORT REVIEW OF PRESENT-DAY APPROACHES

E.A.Mukhacheva, A.S.Mukhacheva, U.I.Alentieva

UfaState Aviation Technological University

e-mail: elita@vmk.ugatu.ac.ru

Abstract. The paper deals with different types of rectangular problems. For the above problems there exist and are still being worked out new methods of coding information that makes it possible for algorithms-decoders to project admissible packing and their drafts. The decoders represent heuristics of polynomial complexity. Nowadays two main approaches of working out decoders are being developed: “bottom-left” and “sequence pair”. In addition to them the paper shows a block algorithm which is included in both approaches by the authors.

Key words: cutting-packing, NP-hard, optimization, decoder, sequence pair, block structure, computation complexity.

МОДИФИЦИРОВАННЫЕ АЛГОРИТМЫ БЛОЧНЫХ ДЕКОДЕРОВ В ЗАДАЧАХ ПРЯМОУГОЛЬНОЙ УПАКОВКИ¹

Э.А. Мухачева, А.С. Филиппова, Д.А. Назаров, М.С. Егорова

Уфимский Государственный Авиационный Технический Университет
e-mail: elita@vmtk.ugatu.ac.ru

Аннотация. В статье предлагается модификация блочных декодеров. Рассматриваются три декодера, использующие различные эвристические подходы и основанные на блочной структуре. Для тестирования декодеров и сравнения результатов их работы была создана оболочка Decoders Testing System (DTS). В заключении проводится сравнительный анализ работы созданных декодеров и ряда других известных декодеров.

Ключевые слова: раскрой и упаковка, NP-трудные задачи, оптимизация, декодеры, блочная структура, граф блочной структуры.

Введение

Обзор эвристических методов (декодеров) для решения задач прямоугольной упаковки представлен в статье A.Lodi, S.Martello&D.Vigo в 2002г. [5]. Там на базе уровневой стратегии приведены детерминированные эвристики: следующий подходящий (Next Fit, **NF**); первый подходящий (First Fit, **FF**); лучший подходящий (Best Fit, **BF**). Сложность алгоритмов для фиксированного приоритетного списка равна $O(m)$. Краткий обзор безуровневых стратегий приведен в статье Э.А. Мухачевой, А.С. Филипповой, Ю.И. Алентьевой [3]. Основная безуровневая стратегия известна как "нижний-левый" (Bottom-Left, **BL**) и состоит в упаковке элемента в самую нижнюю позицию с выравниванием по левому краю. Н. Murata в 1995 г. разработал схему парных последовательностей (Sequence Pair, **SeqPair**) построения упаковок с минимальной площадью огибающего ее прямоугольника [6]. **SeqPair** модифицирован и для упаковки в полубесконечную полосу. Эффективные декодеры разработаны на базе блочного кодирования Э.А. Мухачевой, А.С. Мухачевой, А.В. Чиглинцевым и Р.Р. Ширгазиным [2], [3]. Это алгоритмы "поиска парных списков" (Duality List Search, **DLS**); "замещения" (Substitution, **Sub**) и "конструирования блоков" (Block Design, **BD**).

В данной работе приведены три декодера - MBD, MBD II и IMBD. Они являются модификацией блочных декодеров и показывают как правило более высокие результаты. В ходе эксперимента декодер IMBD показал самые лучшие результаты.

1. Постановка задачи

Рассматривается задача прямоугольной упаковки в полубесконечную полосу. Ее принято именовать **2 Dimensional Strip Packing Problem, 2DSP**. Она задается информационным вектором $(W; m; w; l)$, $w = (w_1, w_2, \dots, w_m)$; $l = (l_1, l_2, \dots, l_m)$, где W - ширина полосы; m - количество прямоугольников; w_i - ширина, l_i - длина i -го прямоугольника. Введем прямоугольную систему координат: оси Ox и Oy совпадают соответственно с ниж-

¹Работа поддержана фондом Президента Российской Федерации, проект МК 145.2003.01

ней неограниченной и боковой сторонами полосы. Положение прямоугольника P_i зададим координатами (x_i, y_i) его левого нижнего угла.

Задача 2DSP. При заданных исходных данных $(W; m; w; l)$ требуется минимизировать длину $L = \max\{x_i + l_i : i = \overline{1, m}\}$ упаковки предметов в полосу при следующих условиях:

- для (x_i, y_i) , $i = \overline{1, m}$ и любой другой вершины (x_i^k, y_i^k) i -го прямоугольника

$$((x_i^k = x_i) \vee (x_i^k = x_i + l_i)) \wedge ((y_i^k = y_i) \vee (y_i^k = y_i + w_i)) \quad (1)$$

- для $i \neq j$; $i, j = \overline{1, m}$

$$([x_i \geq (x_j + l_j)] \vee [x_j \geq (x_i + l_i)]) \vee ([y_i \geq (y_j + w_j)] \vee [y_j \geq (y_i + w_i)]) \quad (2)$$

- для $i = \overline{1, m}$

$$[x_i \geq 0] \wedge [y_i \geq 0] \wedge [y_i + w_i \leq W] \quad (3)$$

Условие 1 означает параллельность ребер прямоугольников граням полосы, "ортогональность упаковки"; условие 2 - размещение прямоугольников без перекрытия друг друга; условие 3 - размещение прямоугольников без перекрытия с гранями полосы. Множество x_i, y_i , $i = \overline{1, m}$, удовлетворяющее условиям 1-3 называют допустимой прямоугольной упаковкой (Rectangle Packing, **RP**). Если длина L занятой части полосы допустимой упаковки достигает минимума, то **RP** называется оптимальной упаковкой.

Последовательность вершин (x_i, y_i) прямоугольников P_i представляет прямую схему кодирования упаковки. Если известны координаты (x_i, y_i) , отвечающие допустимой упаковке **RP**, то для них однозначно строится эскиз упаковки. Представляют интерес способы кодирования, для которых с помощью алгоритмов-декодеров можно перейти к прямой схеме.

2. Модификация блочных декодеров

Пусть имеется прямоугольная упаковка RP в полосу ширины W , и ее длина равна L . Проведем через правые стороны прямоугольников вертикальные линии. Они разделяют RP на r ($r \leq m$) прямоугольных блоков одинаковой ширины W и различной длины χ_v . При этом

$$\sum_{v=1}^r \chi_v = L \quad (4)$$

Разделяющие блоки называются вертикальными, а соответствующее разбиение упаковки - вертикальной блок-структурой. Подробное описание вертикальных и горизонтальных разделяющих блоков можно найти в работах А.С. Мухачевой, Э.А. Мухачевой [1].

Каждому блоку v поставим в соответствие S_v - список номеров прямоугольников, пересекающих v -ый блок снизу вверх, и его длину χ_v . В модификации, которая будет описана ниже, нет четкого закрепления y -координаты за прямоугольником. Текущая y -координата, помещенного в упаковку прямоугольника, и его окончательная y -координата могут различаться. На изменение y -координаты помещенного элемента влияют размещаемые позднее прямоугольники. Однако x -координата прямоугольника фиксируется и не может изменяться в процессе добавления прямоугольников в блочную структуру. Наряду со списками S_v в модифицированных блочных декодерах рассматривается ориентированный граф

$G = \langle V, E \rangle$, где $V = \{1, 2, \dots, m\}$; $E = \{(i, j) : i \succ j\}$, где $i \succ j$ означает, что прямоугольник с номером i непосредственно следует за прямоугольником с номером j хотя бы в одном из блоков. В дальнейшем будем называть G графом блочной структуры. Рассмотрим основные свойства этого графа.

Утверждение 1 (Ацикличность графа блочной структуры). *Граф блочной структуры не содержит циклов.*

Доказательство: Предположим противное: пусть граф блочной структуры содержит хотя бы один цикл. Рассмотрим этот цикл $C = (c_1, c_2, \dots, c_k)$. Так как $(c_1, c_2) \in E$, то $c_1 \succ c_2$. Из этого следует, что $y_{c_1} > y_{c_2}$. Аналогично получаем, что $y_{c_2} > y_{c_3}$ и т.д. Следовательно $y_{c_1} > y_{c_2} > y_{c_3} > \dots > y_{c_k} > y_{c_1}$. Но это невозможно. Полученное противоречие доказывает теорему. \square

Утверждение 2. *Число ребер в графе блочной структуры не превосходит $2m$, где m - число прямоугольников.*

Доказательство: Пусть у нас есть r блоков. Рассмотрим первый блок S_1 . Если в нем находится n_1 прямоугольников, то это означает, что в графе блочной структуры уже должно быть ровно $n_1 - 1$ ребер. Оценим число ребер, которые могут появиться при переходе от i -го блока к $(i+1)$ -ому блоку. В $(i+1)$ -ом блоке появляются новые прямоугольники. Пусть их будет t_{i+1} штук. Тогда для каждого нового прямоугольника в граф блочной структуры нужно добавить не более 2-х новых ребер: одно из них будет входить в вершину, соответствующую этому прямоугольнику, другое - выходить. Таким образом при переходе от i -го блока к $(i+1)$ -ому блоку добавляется не больше $2t_{i+1}$ новых ребер. Из свойств блочной структуры следует, что $\sum_{i=2}^r t_i = m - n_1$. Окончательно получаем, что число ребер в графе блочной структуре не больше $n_1 - 1 + 2 * (m - n_1) = 2 * m - n_1 - 1 < 2 * m$. \square

С помощью графа блочной структуры становится доступной операция вставки прямоугольника. Эта операция позволяет менять y -координаты уже добавленных в упаковку прямоугольников для того, чтобы вставить текущий прямоугольник.

Так как x координаты прямоугольников не изменяются при добавлении новых прямоугольников, то их можно непосредственно запоминать при добавлении нового прямоугольника. Координаты y прямоугольников, содержащихся в упаковке, находят, сделав обход графа блочной структуры в глубину или ширину. Временная сложность алгоритма $O(m)$.

Алгоритм вставки прямоугольника более трудоемкий. Во-первых, проверяют возможность вставки прямоугольника в заданное место. Во время работы этой части алгоритма запоминается дополнительная информация. Если прямоугольник можно вставить, то алгоритм переходит ко второй части - непосредственно выполняет изменения в блочной структуре: изменяет число блоков и их состав, изменяет граф блочной структуры и пересчитывает координаты прямоугольников.

3. Декодер MBD

Идея декодера MBD заключается в том, что не нужно пытаться вставлять прямоугольники левее предыдущего, т.е. для прямоугольников i и j ($i < j$) выполняется условие

$$(x_i < x_j) \vee ((x_i = x_j) \wedge (y_i < y_j)) \quad (5)$$

Если задана оптимальная последовательность прямоугольников, то MBD соберет безотходную упаковку, т.к. в ней не нарушается условие 5. Эта важная характеристика декодера называется "реставрацией".

Вычислительная сложность декодера MBD составляет $O(m^4)$. Однако, на самом деле, декодер MBD работает довольно быстро, т.к. под знаком O скрывается маленькая константа.

4. Декодер IMBD

Декодер IMBD является усовершенствованием декодера MBD. При тестировании декодера MBD на произвольных упаковках становится ясно, что он плохо работает. Действительно, декодер MBD не возвращается обратно, несмотря на то, что там могло остаться достаточно места для размещения текущего прямоугольника. Одно из решений этой проблемы заключается в декодере Improved MBD (второе решение - это декодер MBD II). IMBD пытается поместить прямоугольник как можно ближе к началу упаковки. Однако у этой стратегии есть свои минусы. В частности, если задана оптимальная последовательность прямоугольников, декодер IMBD не всегда собирает упаковку. Но в тестах, где оптимальная последовательность прямоугольников не известна, декодер IMBD показывает очень высокие результаты. Алгоритм декодера IMBD состоит в следующем:

Алгоритм 1. *Декодер IMBD*

```

MBD(m, W, w, l, x, y)
1   InitModifiedBlockStructure(W)
2   For CurrentRect = 1 To m Do
3       CurrentBlock = 1
4       CurrentPosition = 0
5       RectIsInserted = False
6       While Not RectIsInserted Do
7           If CurrentPosition > AmountOfRectsInBlock(CurrentBlock) Then
8               CurrentPosition = 0
9               CurrentBlock = CurrentBlock + 1
10          CurrentPosition = CurrentPosition + 1
11          If InsertInto(w[CurrentRect], l[CurrentRect]) Then
12              RectIsInserted = True
13  GetXY(x, y)
End-Of-MBD

```

Произведем анализ вычислительной сложности данного алгоритма. В строке 1 происходит инициализация блочной структуры. Эта инициализация требует $O(1)$ времени. Цикл 2-12 выполняется m раз. Внутренний цикл (строки 6-12) выполняется не больше чем $O(m^2)$ раз, так как суммарное число элементов во всех блоках не превосходит $\frac{m(m+1)}{2}$. Операция вставки прямоугольника в общем случае требует $O(m^2)$ времени. Следовательно, вычислительная сложность строк 2-12 составляет $O(m^5)$. Подсчет координат прямоугольников (строка 13) требует $O(m)$ времени. Окончательно, вычислительная сложность алгоритма IMBD составляет $O(m^5)$. Однако, декодер IMBD работает быстро.

5. Численный эксперимент

Задачи серии CR были получены из задач E. Норрег [4] серии С путем "перемешивания" прямоугольников. Модифицированные блочные декодеры сравнивались с базовыми алгоритмами BD и IBD. Декодер IBD на данных задачах уступает декодерам BD и IMBD. Самые худшие результаты, как и ожидалось, показывает декодер MBD. Декодер MBD II на решение задачи CR7-3 затрачивает 0.6 секунд, а декодер IMBD - всего лишь 0.3 секунды. Быстрее всех работает декодер MBD. Из полученных результатов видно, что на модифицированных тестах E. Норрег лучшие результаты показывает декодер IMBD.

Задача	M	MBD	MBD II	IMBD	BD	IBD
CR1-1	16	62.50%	76.92%	68.97%	68.97%	68.97%
CR1-2	17	76.92%	68.97%	90.91%	68.97%	68.97%
CR1-3	16	80.00%	86.96%	90.91%	86.96%	86.96%
CR2-1	25	78.95%	88.24%	88.24%	88.24%	88.24%
CR2-2	25	78.95%	68.18%	68.18%	65.22%	71.43%
CR2-3	25	71.43%	78.95%	75.00%	71.43%	78.95%
CR3-1	28	75.00%	76.92%	78.95%	81.08%	78.95%
CR3-2	29	78.95%	78.95%	81.08%	78.95%	81.08%
CR3-3	28	73.17%	73.17%	78.95%	71.43%	78.95%
CR4-1	49	76.92%	82.19%	88.24%	81.08%	84.51%
CR4-2	49	58.82%	71.43%	73.17%	76.92%	65.22%
CR4-3	49	68.18%	76.92%	82.19%	76.92%	72.29%
CR5-1	73	68.18%	75.63%	75.63%	76.92%	77.59%
CR5-2	73	66.18%	78.26%	81.08%	78.26%	78.26%
CR5-3	73	76.27%	84.11%	89.11%	88.24%	82.57%
CR6-1	97	68.97%	74.07%	74.53%	75.00%	75.00%
CR6-2	97	63.16%	71.86%	75.95%	76.92%	79.47%
CR6-3	97	75.95%	83.92%	81.63%	86.33%	83.33%
CR7-1	196	72.95%	81.08%	82.47%	79.21%	78.43%
CR7-2	197	62.34%	70.18%	72.07%	70.18%	68.97%
CR7-3	196	71.86%	80.00%	82.19%	83.92%	81.63%
Среднее		71.70%	77.47%	79.97%	77.67%	77.61%

Список литературы

- [1] Э.А. Мухачева, А.С. Мухачева. *Задача прямоугольной упаковки: методы локального поиска оптимума на базе блочных структур* // Автоматика и телемеханика. No.2, 2004. С. 101-112.

- [2] Мухачева А.С. *Технология блочных структур локального поиска оптимума в задачах прямоугольной упаковки* // Информационные технологии. Приложение No.5. 2004. с. 18-32.
- [3] Э.А. Мухачева, А.С. Филиппова, Ю.И. Алентьева. *Алгоритмы декодирования прямоугольных упаковок: краткий обзор* // Сборник трудов XIII Байкальской международной школы-семинара "МЕТОДЫ ОПТИМИЗАЦИИ И ИХ ПРИЛОЖЕНИЯ".
- [4] Hopper E. <http://home.t-online.de/home/villahost/eva>.
- [5] Lodi A., Martello S., Vigo D. *Recent advances on two-dimensional bin packing problems.* // Discrete Applied Mathematics. 2002. Vol. 123. P. 379 -396.
- [6] H.Murata, K.Fujiyoshi, S.Natake and Y.Kajitani. *Rectangle-Packing-Based Module Placement.* Proc. IEEE/ACM International Conf. on Computer-Aided Design. 1995. P. 472-479.

BLOCK DECODERS' MODIFICATION

E.A. Mukhacheva, A.S. Filippova, D.A. Nazarov, M.S. Egorova

Ufa State Technical University of Aviation
e-mail: elita@vmk.ugatu.ac.ru

Abstract. In this article there are offered new modifications of block decoder. We consider block decoder with the modification and two block decoders which use another heuristic approaches. Decoders Testing System (DTS) was created for testing the decoders and for comparison the results of their work. At the conclusion a comparative analysis of created decoders and a number of another known decoders is being carried out.

Key words: cutting and packing, NP-complete problems, optimization, decoders, block structure, block structure's graph

ПРИБЛИЖЕННЫЙ ЛОКАЛЬНЫЙ ПОИСК В ЗАДАЧЕ О p -МЕДИАНЕ ¹

А.В.Плясунов

Институт математики им. С. Л. Соболева СО РАН, Новосибирск
e-mail: apljas@math.nsc.ru

Аннотация. Для задачи о p -медиане получены следующие результаты.

1. Если окрестность N сильнее окрестности $Swap$, то необходимым условием N -оптимальности допустимого решения является существование $Swap$ -локальной седловой точки.
2. Для любой полиномиально проверяемой окрестности ε -локальный минимум может быть вычислен за время полиномиально ограниченное длиной входа задачи и величиной $1/\varepsilon$.

Ключевые слова: NP-трудные задачи, задачи размещения, локальный поиск, PLS-полные задачи, седловые точки, (ε, N) -локальные оптимумы.

Введение

Отсутствие эффективных алгоритмов решения для NP-трудных оптимизационных задач привело к появлению множества алгоритмов вычисляющих их приближенные решения. Один из общих методов построения таких алгоритмов является локальный поиск. Основная идея этого метода заключается в получении необходимых условий глобального экстремума, имеющих конструктивный характер. С этой целью каждому допустимому решению сопоставляется множество его соседей, т.е. окрестность. Конструктивность необходимых условий экстремума реализуется в случае, когда возможна эффективная проверка локальной оптимальности текущего решения относительно данной окрестности. Очевидна аналогия с необходимыми условиями экстремума в нелинейной оптимизации, где вместо окрестностей оперируют с понятием вариации [3]. В данной работе приводится результат, подчеркивающий нетривиальность этой аналогии: в задаче о p -медиане булев вектор y является локально оптимальным относительно окрестности $Swap$ тогда и только тогда, когда для подходящего вектора множителей Лагранжа γ пара (y, γ) является локальной относительно этой окрестности седловой точкой. Получается дискретный аналог теоремы Куна-Таккера.

Притягательность локального поиска в его простоте и удивительной эффективности при решении практических задач. Основная область его приложений – решение NP-трудных задач. Отметим конкурентоспособность локального поиска и при решении полиномиально разрешимых оптимизационных задач. До сих пор локальный поиск на практике оказывается лучшим при решении задач линейного программирования, ряда потоковых задач. Еще одна сфера приложения алгоритмов, основанных на этом подходе – доказательство существования тех или иных комбинаторных объектов [5, 7]. Помимо этого многие метаэвристики используют локальный поиск.

Одна из целей этой работы – продолжить исследования возможностей локального поиска в задаче о p -медиане с точки зрения худшего случая, начатые в [4]. На практике стандартный алгоритм локального спуска сходится очень быстро, за линейное или квадратичное число итераций. Вместе с тем известен широкий класс задач локального поиска,

¹Работа выполнена при поддержке РФФИ (проект 03-01-00455)

класс PLS , в котором по аналогии с NP-полными задачами удалось выделить класс наиболее трудных задач. Они названы PLS -полными [10]. Существование полиномиального алгоритма нахождения локального оптимума для любой из них, влечет существование полиномиального алгоритма для всех задач из класса PLS . Аналогичная ситуация возникает и для задачи о p -медиане. В [4] для нее показано, что задача поиска локального минимума с многими полиномиально проверяемыми окрестностями является PLS -полной. Однако, в настоящее время нет никаких прямых или косвенных аргументов в пользу совпадения или различия класса P_{LS} – полиномиально разрешимых задач локального поиска и класса PLS .

Задача о p -медиане моделирует ситуацию, когда в возможных пунктах производства нужно разместить p предприятий так, чтобы суммарное расстояние от них до потребителей было минимальным. Обозначим через I множество возможных пунктов производства, а через J – множество потребителей. Пусть g_{ij} – расстояние между j -м потребителем и i -м возможным пунктом производства.

Таким образом вход задачи определяется матрицей $(g_{ij}), i \in I, j \in J$, и целым положительным числом p . Без ограничения общности можно считать, что величины g_{ij} – положительные рациональные числа. Допустимым решением задачи является любое подмножество $S \subset I$ мощности p . Пусть $c_g(S) = \sum_{j \in J} \min_{i \in S} g_{ij}$ – значение целевой функции на этом множестве.

Сопоставим каждому решению S подмножество допустимых решений $N(S)$. Множество $N(S)$ называют окрестностью решения S , а отображение N окрестностью задачи о p -медиане. Решение S называется локальным минимумом относительно этой окрестности или N -оптимальным решением, если множество $N(S)$ не содержит решений с меньшим значением целевой функции. Будем говорить, что окрестность N_1 сильнее окрестности N_2 , если из N_1 -оптимальности следует N_2 -оптимальность.

1. Локальные седловые точки

Задачу о p -медиане можно записать как задачу минимизации псевдодобулевого полинома с одним ограничением [1]:

$$P(y) = \sum_{j \in \Pi} a_j \prod_{i \in \alpha_j} y_i \longrightarrow \min$$

$$\sum_{i=1}^m y_i = n - p,$$

$$y_i \in \{0, 1\}, i \in I = \{1, \dots, m\},$$

где $\alpha_j \subset I, j \in \Pi = \{1, \dots, n\}$.

Пусть S – произвольное допустимое решение задачи о p -медиане. Тогда $y_i = 1 \iff i \in S$. Для любого булевого вектора $y = (y_1, \dots, y_m)$ под окрестностью $Swap(y)$ будем понимать множество булевых векторов y' , отличающихся от y ровно в двух координатах и имеющих то же число единичных компонент.

Заменим условие булевости переменных на ограничения $0 \leq y_i \leq 1$. Тогда функция Лагранжа с множителями $\lambda, \mu_i \geq 0, \sigma_i \geq 0, i \in I$ и непрерывными переменными $y_i, i \in I$ имеет вид:

$$L(y, \lambda, \mu, \sigma) = P(y) + \left(\sum_{i \in I} y_i - n + p \right) \lambda + \sum_{i \in I} \sigma_i (y_i - 1) - \sum_{i \in I} \mu_i y_i.$$

Вектор $(\bar{y}, \bar{\lambda}, \bar{\mu}, \bar{\sigma})$ назовем седловой точкой относительно окрестности $Swap$, если

$$L(\bar{y}, \lambda, \mu, \sigma) \leq L(\bar{y}, \bar{\lambda}, \bar{\mu}, \bar{\sigma}) \leq L(y, \bar{\lambda}, \bar{\mu}, \bar{\sigma})$$

для всех $\lambda, \mu \geq 0, \sigma \geq 0$ и всех $y \in Swap(\bar{y})$.

Аналогично вводится понятие седловой точки относительно дискретной окрестности и для произвольной задачи математического программирования. Можно показать, что существование седловой точки относительно этой окрестности является достаточным условием локальной оптимальности соответствующего вектора. Обоснование этого утверждения следует из [8] с необходимыми модификациями, которые учитывают дискретность окрестности. Для задачи о p -медиане верен более общий результат.

Теорема 1 (дискретная форма теоремы Куна-Таккера). *Булев вектор $y = (y_1, \dots, y_m)$ является $Swap$ оптимальным решением задачи о p -медиане если и только если найдутся такие множители $\lambda, \mu_i \geq 0, \sigma_i \geq 0, i \in I$, что вектор $(y, \lambda, \mu, \sigma)$ является седловой точкой относительно окрестности $Swap$.*

Этой теореме можно придать другую форму, использующую условие Куна-Таккера [3].

Булев вектор $y^ = (y_1^*, \dots, y_m^*)$ – $Swap$ -оптимальное решение задачи о p -медиане если и только если найдутся такие множители $\lambda^*, \mu_i^* \geq 0, \sigma_i^* \geq 0, i \in I$, что вектор y^* является оптимальным решением задачи:*

$$L(y, \lambda^*, \mu^*, \sigma^*) \longrightarrow \min_{y \in Swap(y^*)}$$

$$\left(\sum_{i \in I} y_i - n + p \right) \lambda^* = 0, \quad \sigma_i^* (y_i - 1) = 0, \quad \mu_i^* y_i = 0, \quad i \in I,$$

$$\sum_{i \in I} y_i = n - p.$$

Очевидно, что необходимым условием оптимальности допустимого решения y в задаче о p -медиане является существование седловой точки относительно окрестности $Swap$. Более того, если N – произвольная окрестность, которая сильнее чем $Swap$, то необходимым условием для N -оптимальности допустимого решения y является существование седловой точки относительно окрестности $Swap$.

2. Вычисление ε -локального минимума

Для построения приближенных эффективных алгоритмов решения задач локального поиска предложен подход [9], основная идея которого состоит в переходе от локальной оптимальности к понятию ε -локальной оптимальности. Покажем как реализовать эту идею для задачи о p -медиане.

Рассмотрим задачу локального поиска (p -median, N), где N – произвольная полиномиально проверяемая окрестность. Допустимое решение S^ε назовем (ε, N) -локальным минимумом для $\varepsilon > 0$, если $c(S^\varepsilon) \leq (1 + \varepsilon)c(S)$ для всех $S \in N(S^\varepsilon)$. Семейство алгоритмов $\{A_\varepsilon\}_{\varepsilon > 0}$ назовем вполне полиномиальной (ε, N) -локальной схемой, если для каждого ε алгоритм A_ε находит (ε, N) -локальный минимум и время его работы полиномиально ограничено длиной входа и величиной $1/\varepsilon$.

Теорема 2. Для задачи поиска (ε, N) -локального минимума с полиномиально проверяемой окрестностью N существует вполне полиномиальная (ε, N) -локальная схема.

Идея доказательства состоит в следующем. Чтобы получить требуемую схему, возьмем псевдополиномиальный алгоритм (в данном случае стандартный алгоритм локального спуска [10]) и масштабируем задачу, чтобы за счет уменьшения точности добиться полиномиального времени работы алгоритма [2, 6].

Пусть ϱ – коэффициент масштабирования и (g'_{ij}) – матрица, полученная масштабированием исходной, т.е. $g'_{ij} = \lceil g_{ij}/\varrho \rceil \varrho$. Пусть алгоритм локального спуска начинает работу с решения S^0 и получает локальный минимум S^e .

Предположим, что существует величина Δ такая, что $n\varrho < \Delta \leq c_g(S^e) \leq c_g(S^0)$ – начальное решение, с которого стартует алгоритм при вычислении S^e . Тогда для $S \in N(S^e)$ имеем

$$c_g(S^e) \leq c_{g'}(S^e) \leq c_{g'}(S) \leq c_g(S) + n\varrho.$$

Отсюда следует, что

$$c_g(S^e) \leq c_g(S)(1 + n\varrho/c_g(S)) \leq c_g(S)(1 + n\varrho/(c_g(S^e) - n\varrho)) \leq c_g(S)(1 + n\varrho/(\Delta - n\varrho)).$$

Для того, чтобы множество S^e было ε -локальным минимумом нужно выбрать коэффициент масштабирования ϱ равным величине $\varepsilon\Delta/((\varepsilon + 1)n)$. Если величина Δ имеет тот же порядок величины что и величина $c_g(S^0)$, то число шагов алгоритма локального спуска ограничено некоторым полиномом от n и ε^{-1} . Действительно, для матрицы (g'_{ij}) на каждом шаге значение целевой функции убывает по меньшей мере на ϱ единиц. Поэтому число шагов оценивается величиной $O(c_{g'}(S^0)/\varrho) = O(c_g(S^0)/\varrho) = O(n\varepsilon^{-1})$.

Уточним предположение о существовании величины Δ , имеющей порядок $O(c_g(S^0))$. Из результатов полученных в [4] следует, что если задача $(p\text{-median}, N)$ – PLS -полная, то эта величина может и не существовать. Чтобы получить требуемую схему, несколько усложним алгоритм.

Пусть $c^0 = c_g(S^0)$, положим

$$\Delta = c_g(S^0)/2, \varrho = \varepsilon\Delta/((\varepsilon + 1)n)$$

и начнем поиск локального минимума для матрицы g' . Продолжаем вычисления до тех пор пока текущее решение S не локальный минимум и $c_g(S) \geq \Delta$. Если $c_g(S) < \Delta$, то пересчитываем значения параметров:

$$S^0 = S, \Delta = c_g(S^0)/2, \varrho = \varepsilon\Delta/((\varepsilon + 1)n)$$

и продолжаем вычисления с новыми значениями. Если S – локальный минимум в задаче с матрицей g' , то величина Δ имеет требуемый порядок и S – искомый ε -локальный минимум. Вычисления на этом прекращаются.

Между пересчетами значений параметров требуется $O(n\varepsilon^{-1})$ шагов, а всего пересчетов не более $O(\log c^0) \leq O(\log(n g_{max}))$, где $g_{max} = \max_{ij} g_{ij}$. Поэтому общее число шагов при вычислении ε -локального минимума $O(n\varepsilon^{-1} \log(n g_{max}))$. По условию окрестность N – полиномиально проверяемая. Поэтому вычисление начального решения и очередного приближения выполняется за полиномиальное время. Таким образом получаем вполне полиномиальную (ε, N) -локальную схему.

Как показывает следующее утверждение для задачи поиска локального минимума с полиномиально проверяемой окрестностью N , которая не слабее окрестности S_{war} , не существует константной аддитивной оценки.

Теорема 3. *Если $P_{LS} \neq PLS$, то не существует полиномиального алгоритма, который для произвольного входа задачи локального поиска (p -median, N) находит решение S^* такое, что $c_g(S^*) \leq c_g(S) + \varepsilon$ для всех S из окрестности $N(S)$, где $\varepsilon > 0$ – фиксированная константа.*

Доказательство получается модификацией конструкции, использованной в [2] при обосновании теоремы 6.6.

Список литературы

- [1] Береснев В.Л., Гимади Э.Х., Дементьев В.Т. *Экстремальные задачи стандартизации*. Новосибирск: Наука, 1978.
- [2] Гэри М., Джонсон Д. *Вычислительные машины и труднорешаемые задачи*. М: Мир, 1982.
- [3] Иоффе А.Д., Тихомиров В.М. *Теория экстремальных задач*. Москва: Наука, 1974.
- [4] Кочетов Ю., Пащенко М., Плясунов А. *Локальный поиск в задаче о p -медиане*. Новосибирск, материалы конференции Дискретный анализ и исследование операций, 2004, с. 186.
- [5] Кочетов Ю. А. *Вероятностные методы локального поиска для задач дискретной оптимизации*. Дискретная математика и ее приложения. Сборник лекций. Вып. 1. М.: Издательство центра прикл. исслед. при мех.-мате МГУ. 2001. С. 84–117.
- [6] Ausiello G., Crescenzi P., Gambosi G., Kann V. etc. *Complexity and Approximation: Combinatorial Optimization Problems and their Approximability Properties*. Berlin: Springer–Verlag, 1999.
- [7] Caporossi G., Hansen P. *Variable Neighborhood for Extremal Graphs 1. The system AutoGraphiX*. Discr. Math., 2000, V. 212, p. 29–44.
- [8] Gould F.J. *Extensions of lagrange multipliers in nonlinear programming*. SIAM. J. Appl. Math., 1969, V. 17, № 6, p. 1280–1297.
- [9] Orlin J.B., Punnen A.P., Schulz A.S., *Approximate local search in combinatorial optimization*. Proceedings of the fifteenth annual ACM-SIAM symposium on Discrete algorithms, January 11-14, 2004, New Orleans, Louisiana
- [10] Yannakakis M. *Computational Complexity*. in: E. Aarts and J. K. Lenstra (eds.) *Local Search in Combinatorial Optimization*. Chichester: John Wiley & Sons, 1997, p. 19–56.

AN APPROXIMATE LOCAL SEARCH FOR THE P-MEDIAN PROBLEM

Alexander V. Playsunov

Sobolev Institute of Mathematics, Novosibirsk
e-mail: apljas@math.nsc.ru

Abstract. For the p -median problem we present the following results.

1. If neighborhood N is stronger than neighborhood Swap then necessary condition of N -optimality for a feasible solution is existence of Swap-optimal saddle point.
2. If neighborhood N is polynomially searchable then a (ε, N) -local optimum can be founded in polynomial time both in the problem size and in $1/\varepsilon$.

Key words: NP-hard problems, facility location problems, local search, PLS-complete problems, saddle points, (ε, N) -local optimum

АЛГОРИТМ НАХОЖДЕНИЯ ПРИБЛИЖЕННОГО РЕШЕНИЯ ЗАДАЧИ ПОИСКА ОПТИМАЛЬНОГО ВЛОЖЕНИЯ ГРАФА

А.В. Пролубников

Омский государственный университет, Омск
e-mail: prolubnikov@univer.omsk.su

Аннотация. В статье рассматривается задача проверки изоморфизма подграфу с дополнительным ограничением, оставляющим задачу в классе NP -полных задач. А именно, предполагается, что количество вершин в обоих графах одинаковое. Вводится метрика, представляющая расстояние между графами. Введенная метрика позволяет рассматривать задачу как задачу поиска оптимального вложения одного графа в другой. Предлагается полиномиальный по количеству элементарных машинных операций алгоритм поиска приближенного решения задачи. Рассматриваются условия, в которых алгоритм является эффективным.

Ключевые слова: изоморфизм, метрика, оптимальное вложение, система линейных уравнений.

Введение

В задаче поиска оптимального вложения графа, являющейся модификацией задачи проверки изоморфизма подграфу, имеется некоторая целевая функция, требующая минимизации. Целевая функция представляет собой некоторую количественную характеристику близости графов. Примерами функции расстояния могут служить функции представленные в [2], [3]. В [4] вводится расстояние между классами изоморфных графов, где расстояние определяется как наименьшее возможное число вершин графа, содержащего подграфы из обоих классов, и доказывается, что классы изоморфных графов с введенным функционалом являются метрическим пространством. Однако, одна из дополнительных сложностей решения задачи в том, что задача определения расстояния между графами сама по себе, как правило, оказывается очень сложной. В [5] доказана NP -трудность вычисления некоторых естественных расстояний между классами изоморфных неориентированных графов.

Граф $G_A = \langle V_A, E_A \rangle$ изоморфен подграфу графа $G_B = \langle V_B, E_B \rangle$, если существует такая инъекция $\varphi : V_A \rightarrow V_B$, что если $(i, j) \in E_A$, то $(\varphi(i), \varphi(j)) \in E_B$. Требуется найти φ или показать, что такой инъекции не существует.

Будем в дальнейшем рассматривать случай $|V_A| = |V_B|$. Отметим, что задача проверки изоморфизма подграфу остается NP -полной и с этим дополнительным условием.

1. Постановка задачи

Вложением графа $G_A = \langle V_A, E_A \rangle$ в граф $G_B = \langle V_B, E_B \rangle$ ($|V_A| = |V_B|$, $|E_A| \leq |E_B|$) будем называть биективное отображение $\varphi : V_A \rightarrow V_B$. Граф, получаемый из графа G_A перенумерацией вершин, соответствующей биекции φ , будем обозначать G_A^φ и также называть вложением.

Любому вложению φ может быть однозначно поставлена в соответствие некоторая

матрица перестановки P , и наоборот. Элементы P следующие:

$$p_{ij} = \begin{cases} 1, & \text{если } i = \varphi(j), \\ 0, & \text{иначе.} \end{cases}$$

Под матрицей $A(\varphi)$ будем понимать матрицу PAP^{-1} .

Изоморфным вложением графа $G_A = \langle V_A, E_A \rangle$ в $G_B = \langle V_B, E_B \rangle$ ($|V_A| = |V_B|$, $|E_A| \leq |E_B|$) будем называть биективное отображение $\varphi : V_A \rightarrow V_B$ такое, что если $(i, j) \in E_A$, то $(\varphi(i), \varphi(j)) \in E_B$. Если $|E_A| = |E_B|$, то существование изоморфного вложения графа G_A в граф G_B эквивалентно изоморфизму этих графов.

Введем функцию расстояния между графами. Пусть A_0, B_0 – матрицы смежности графов G_A и G_B . Матрица $C_0 = A_0 - B_0$ – матрица смежности некоторого взвешенного графа с весами ребер равными 1 и -1. Ее элементы следующие:

$$c_{ij} = \begin{cases} 1, & (i, j) \in E_A \text{ и } (i, j) \notin E_B, \\ -1, & (i, j) \notin E_A \text{ и } (i, j) \in E_B, \\ 0, & (i, j) \in E_A \text{ и } (i, j) \in E_B, \\ & \text{или } ((i, j) \notin E_A \text{ и } (i, j) \notin E_B). \end{cases}$$

Матрица C_0 – симметрическая, поскольку матрицы A_0 и B_0 симметрические. Задаваемый ей взвешенный граф является дополнением графа G_A до графа G_B только в том случае, когда в C_0 отсутствуют элементы равные -1, то есть когда G_A – подграф G_B . Матрица C_0 задает взвешенный граф G_C , наличие ребра в котором соответствует наличие такого же ребра только в одном из графов G_A или G_B , отсутствию – либо наличие, либо отсутствие такого ребра в обоих графах. Будем считать, что чем меньше ребер в графе G_C и, соответственно, меньше ненулевых элементов в матрице C_0 , тем G_B более близок к G_A . Будем в дальнейшем граф G_C , соответствующий матрице C_0 , называть разностью графов G_A и G_B и введем метрику, конкретизирующую понятие близости графов.

Рассмотрим фробениусову (евклидову) норму матрицы F для матрицы $A = (a_{ij})$.

$$F(A) \stackrel{df}{=} \left(\sum_{i=1}^n \sum_{j=1}^n a_{ij}^2 \right)^{1/2}.$$

При перенумерации вершин графа G_A , соответствующей биекции φ , получаем граф G_A^φ с матрицей смежности PA_0P^{-1} , обозначаемой далее $A_0(\varphi)$. Матрица разности графов G_B и G_A^φ – матрица $C_0(\varphi)$, $C_0(\varphi) = B_0 - A_0(\varphi)$. Значение $F(C_0(\varphi))$ равно квадратному корню от числа несовпадающих ребер в графах G_A^φ и G_B .

Функция $\delta(G_A, G_B) \stackrel{df}{=} \min_{\varphi} F(C_0(\varphi))$, рассматриваемая как функция расстояния между графами, является метрикой, поскольку δ неотрицательна и $\delta(G_A, G_B) = 0$, если только $G_A \simeq G_B$. δ симметрична, то есть $\delta(G_A, G_B) = \delta(G_B, G_A)$. δ также удовлетворяет неравенству треугольника, что очевидно, если рассматривать $F(A)$ как евклидову норму вектора, представляющего собой последовательную конкатенацию строк матрицы A .

После введения метрики δ задача поиска оптимального вложения графа G_A в граф G_B может быть сформулирована так: необходимо найти вложение φ графа G_B в граф G_A , при котором $F(C_0(\varphi)) = \delta(G_A, G_B)$.

2. Алгоритм

Матрицы, представляющие графы, с которыми работает алгоритм, имеют следующий вид: $A = A_0 + D_{A_0}$, $B = B_0 + D_{B_0}$, где A_0, B_0 – матрицы смежности графов, D_{A_0}, D_{B_0} – диагональные матрицы, в i -й позиции каждой из которых находятся степени d_i соответствующих вершин графов. Вместо матрицы C_0 как матрицу, представляющую разность графов G_B и G_A , будем рассматривать матрицу C такую, что $C = A - B$.

Если

$$F(B_0 - A_0(\varphi)) = F(C_0(\varphi)) = \delta(G_A, G_B),$$

то и

$$F(B - A(\varphi)) = F(C(\varphi)) = \min_{\varphi} (F(B - A(\varphi))).$$

В самом деле:

$$B - A(\varphi) = (B_0 - A_0(\varphi)) + (D_B - D_A(\varphi)).$$

Если φ – оптимальное вложение, то это значит, что количество несовпадающих ребер у вложения $G_A(\varphi)$ и графа G_B минимально, что также соответствует минимуму разности степеней у соответствующих вершин графов. Следовательно, минимизация функционала $F(B_0 - A_0(\varphi))$ равносильна минимизации функционала $F(B - A(\varphi))$.

В представленном алгоритме под нормой вектора понимается евклидова норма в R^n .

Алгоритм приближенного решения задачи поиска оптимального вложения графа

Шаг 0. $A^0 := A$, $B^0 := B$; $j := 1$.

Шаг 1.1. Если $j \leq n$, то перейти на шаг 1.2, иначе перейти на шаг 6.

Шаг 1.2. Выбор ε_j .

Шаг 1.3. $A^j := A^{j-1} + \varepsilon_j E^j$.

Шаг 2. Решение системы линейных уравнений $A^j x = e_j$. x_j – полученное решение.

Шаг 3.1. $k := 1$. Если $k \leq n$, то перейти на шаг 3.2, иначе перейти на шаг 4.1.

Шаг 3.2. $B_k := B^j + \varepsilon_j E^k$.

Шаг 3.3. Решение системы линейных уравнений $B^k y = e_k$. y_k – полученное решение.

Шаг 3.4. $k := k + 1$. Перейти на шаг 3.1.

Шаг 4.1. Сравнение норм векторов x_j и y_k , где k такие, что $\exists i : k_i = k$ ($i \leq j$).

Если $k : \left| \|x_j\| - \|y_k\| \right| = \min_l \left| \|x_j\| - \|y_l\| \right|$, то $k_j := k$.

Шаг 4.2. $B^j := B^{j-1} + \varepsilon_j E^{k_j}$.

Шаг 5. $j := j + 1$. Перейти на шаг 1.1.

Шаг 6. Работу алгоритма завершить. Отображение $\varphi : V_B \rightarrow V_A$ такое, что $\varphi(k_j) = j$, $j = \overline{1, n}$ – найденное вложение графа G_A в граф G_B .

Отметим, что данный алгоритм может быть рассмотрен как эвристический алгоритм проверки изоморфизма графов. Если графы G_A и G_B близки в соответствии с введенной метрикой δ , и φ_0 – искомое оптимальное вложение, а P^0 – соответствующая ему матрица перестановки, то на каждой j -й итерации алгоритма должно выполняться следующее ключевое соотношение:

$$B^j = B^{j-1} + \varepsilon_j E^{\varphi_0(j)} \Leftrightarrow F(B^j - P_0 A^j P_0^{-1}) = \min_P \{F(B - P A P^{-1})\}, \quad (1)$$

где минимум берется по всем матрицам перестановки размерности $n \times n$.

Пусть $B - A(\varphi_0) = C(\varphi_0)$. Рассмотрим системы линейных уравнений

$$A(\varphi_0)x = e_j, \quad Bx' = e_j, \quad j = \overline{1, n}, \quad (2)$$

где векторы $\{e_j\}_{j=1}^n$ – стандартный базис в пространстве R^n . Если для некоторого θ

$$\|C(\varphi_0)\|/\|A(\varphi_0)\| \leq \theta, \quad \theta\mu(A(\varphi_0)) < 1, \quad (3)$$

где $\mu(A(\varphi_0))$ – число обусловленности матрицы $A(\varphi_0)$, то для решений x и x' систем уравнений (2) справедливо [6] неравенство

$$\frac{\|x - x'\|}{\|x\|} \leq \frac{\mu(A(\varphi_0))\theta}{1 - \mu(A(\varphi_0))\theta}. \quad (4)$$

Для матриц, представляющих графы, с которыми работает алгоритм, число обусловленности не превышает 4. В самом деле, для числа обусловленности симметрической матрицы $\mu(A) \leq \eta(A)/\chi(A)$ [6], где

$$\eta(A) = \max_{1 \leq i \leq n} (a_{ii} + \sum_{j \neq i} |a_{ij}|), \quad \chi(A) = \min_{1 \leq i \leq n} (a_{ii} - \sum_{j \neq i} |a_{ij}|).$$

При возмущении диагональных элементов матрицы A имеем: $a_{kk} = d + d_k + \varepsilon_k$, где ε_k – значение возмущения k -го диагонального элемента матрицы A . Пусть i_1 – номер строки, на котором достигается $\eta(A)$, i_2 номер строки, на котором достигается $\chi(A)$. Тогда

$$\eta(A) = a_{i_1 i_1} + \varepsilon_{i_1 i_1} + \sum_{j=1}^n |a_{i_1 j}| = d + \varepsilon_{i_1 i_1} + d_{i_1} + d_{i_1} = d + \varepsilon_{i_1 i_1} + d + d = 3d + \varepsilon_{i_1 i_1},$$

$$\chi(A) = a_{i_2 i_2} + \varepsilon_{i_2 i_2} - \sum_{j=1}^n |a_{i_2 j}| = d + \varepsilon_{i_2 i_2} + d_{i_2} - d_{i_2} = d + \varepsilon_{i_2 i_2}.$$

Следовательно,

$$\mu(A) \leq \frac{\eta(A)}{\chi(A)} = \frac{3d + \varepsilon_{i_1 i_1}}{d + \varepsilon_{i_2 i_2}}.$$

Будем считать, что ε_j , выбираемое на шаге 1.2 алгоритма, равно $1/n^p$. Вопрос о том, каким должно быть p , подробно рассмотрен в [7]. Если $\varepsilon_{i_1 i_1} = 1/n^{p_1}$, $\varepsilon_{i_2 i_2} = 1/n^{p_2}$, где p_1, p_2 – натуральные числа, то

$$\mu(A) \leq \frac{3d + \varepsilon_{i_1 i_1}}{d + \varepsilon_{i_2 i_2}} = \frac{3d + 1/n^{p_1}}{d + 1/n^{p_2}} = n^{p_2 - p_1} \frac{3dn^{p_1} + 1}{dn^{p_2} + 1} \leq n^{p_2 - p_1} \frac{4dn^{p_1}}{dn^{p_2}} = 4.$$

Поэтому если выполняется (4), и $\theta < 1/4$, то для решений x и x' систем уравнений (2) неравенство (4) принимает следующий вид:

$$\|x - x'\| \leq \frac{4\theta}{1 - 4\theta} \|x\|.$$

То есть решения систем уравнений (2), задающие обратные матрицы для матриц $A(\varphi_0)$ и B , изменяются непрерывно при малых возмущениях, задаваемых матрицей $C(\varphi_0)$. А значит

и дополнительные возмущения диагональных элементов матриц, производимые на каждой j -й итерации, с некоторой погрешностью будут сохранять ключевое соотношение (1).

Возмущая матрицы A и B с помощью матриц $\varepsilon_j E^j$ и $\varepsilon_j E^{\varphi_0(j)}$, полагая, при этом, что на каждой j -й итерации $\varphi_0(j) = k$, где k такое, что

$$\left| \|x_j\| - \|y_k\| \right| = \min_l \left| \|x_j\| - \|y_l\| \right|,$$

мы минимизируем n из n^2 слагаемых, входящих в функционал $F((B^j)^{-1} - P(A^j)^{-1}P^{-1})$.

Стремясь уменьшить на каждой итерации алгоритма значение функционала $F((B^j)^{-1} - P(A^j)^{-1}P^{-1})$, при неизвестной нам матрице перестановки P , мы исходим из того, что два функционала – функционал $F(B^j - PA^jP^{-1})$ и функционал $F((B^j)^{-1} - P(A^j)^{-1}P^{-1})$ достигают минимума на одной и той же матрице перестановки, что справедливо, в частности, когда $G_A \simeq G_B$, так как в этом случае

$$B - PAP^{-1} = 0 \Leftrightarrow B^{-1} - PA^{-1}P^{-1} = 0.$$

Если же $\delta(G_A, G_B)$ – мало, то, поскольку $\|C(\varphi_0)\| \leq F(C)$ [6], будет мало и отношение $\|C(\varphi_0)\|/\|A(\varphi_0)\|$, а значит возможно эффективное применение алгоритма поиска изоморфного вложения графа, что подтверждается вычислительным экспериментом.

Результаты проведенного вычислительного эксперимента показали, что алгоритм работает эффективно в случае, если графы, поданные на его вход, обладают достаточной реберной плотностью, которая определялась как вероятность, с которой при генерировании случайного графа между двумя вершинами графа проводится ребро. Искалось изоморфное вложение графа G_A в граф G_B , заведомо существующее. Генерировался граф G_B , по нему строился его остовный подграф G'_A , после чего случайным образом перенумеровывались его вершины и таким образом получался граф G_A . Если плотность графа G_B составляла 0,65 и выше, а количество удаленных из него при формировании подграфа G'_A ребер составляло не более 20% от числа всех ребер в G_B , то в полученном алгоритмом вложении количество ребер (его усредненное по серии экспериментов значение), несовпадающих с ребрами графа G_B , не превышало 1% от разницы в количестве ребер у графов G_A и G_B . Количество вершин в тестовых графах составляло 300.

Список литературы

- [1] Baker, B.S. *Approximation algorithms for NP-complete problems on planar graphs* // J. Assoc. Comput. Mach. 1994. V. 41. P. 153–180.
- [2] Bunke, H. *On a relation between graph edit distance and maximum common subgraph* // Pattern Recogn. Lett. 1997. V. 18, N. 8. P. 689–694.
- [3] Bunke, H., Schearer, K. *A Graph distance metric based on the maximal common subgraph* // Pattern Recogn. Lett. 1998. V. 19, N. 3–4. PP. 255–259.
- [4] Foudree, R.J., Schelp, R.H., Lesniak, L., Gyärfäs, A., Lehel, J. *On the rotation distance of graphs* Discrete Math., 126, № 1–3, 1994. PP. 121–135.
- [5] Корнеенко Н.М. *О сложности вычисления расстояния между графами* Изв. АН БССР. Сер. физ.-мат. н. № 1, 1982. С. 10–13.

- [6] Годунов С.К. и др. *Гарантированная точность решения систем линейных уравнений в евклидовых пространствах* Новосибирск: Наука, 1988.
- [7] Пролубников А.В., Файзуллин Р.Т. *Класс графов, задача проверки изоморфизма для которых разрешима за полиномиальное время алгоритмом спектрального расщепления* // Математические структуры и моделирование: Сб. научн. тр. Омск: Омск. гос. университет, 2003. Вып. 11. С. 28–57.

AN ALGORITHM FOR FINDING APPROXIMATE SOLUTION OF THE PROBLEM OF OPTIMAL GRAPH EMBEDDING

A.V. Prolubnikov

Omsk State university, Omsk
e-mail: prolubnikov@univer.omsk.su

Abstract. We consider the subgraph isomorphism problem which has the following restriction. We suppose that both of checking graphs have the same number of vertices. Note that this problem is *NP*-complete still. Proposed metric represents the distance between graphs. The metric allows to come to the problem of finding an optimal embedding of one graph into another. The supposed algorithm is an algorithm for finding an approximate solution for the problem. The algorithm is polynomial-time in sense of using elementary operations.

Key words: isomorphism, metric, optimal embedding, linear algebraic equations system.

ОБ ОЦЕНКЕ ТОЧНОСТИ ГРАДИЕНТНОГО АЛГОРИТМА В ОДНОЙ ЗАДАЧЕ ВЫПУКЛОЙ ДИСКРЕТНОЙ ОПТИМИЗАЦИИ И НЕКОТОРЫЕ СМЕЖНЫЕ ВОПРОСЫ

А. Б. Рамазанов

Бакинский Государственный Университет, Баку
e-mail: rab-unibak@rambler.ru, ram-bsu@mail.ru

Аннотация. В работе получена новая улучшенная апостериорная гарантированная оценка погрешности градиентного алгоритма в задачах максимизации строго выпуклых функций дискретного аргумента на специальном порядково-выпуклом множестве. Приведены гарантированные оценки выраженные через чебышевскую норму и собственных чисел матрицу входящие условия ограничения. Установлено устойчивости градиентного алгоритма относительно возмущения некоторых параметров допустимой области.

Ключевые слова: оценка, градиент, алгоритм, выпуклость, устойчивость.

1. Введение

Дискретный аналог градиентного алгоритма имеет многочисленные применения (см., напр., [1-4]). Нахождения и анализ оценок точности градиентного алгоритма является одной из основных ввиду невозможности точного и в то же время эффективного решения большинства таких задач [1, 3, 4] (др. ссылки см. [2]). Одна из формализаций градиентных алгоритмов в дискретной оптимизации (ДО) и методик оценки их погрешности предложена в модели порядковой выпуклости [1, 3] (см. также [4]). Актуальным является установление устойчивости градиентного алгоритма в задачах ДО относительно возмущения параметров задачи. Под устойчивостью градиентного алгоритма, как обычно понимают некоторые свойства инвариантности гарантированных оценок точности градиентного алгоритма при "малых" возмущениях параметров, входящие в эту оценку.

В настоящем докладе представлены новые улучшенные гарантированные оценки точности градиентного алгоритма покоординатного подъема в задачах максимизации ρ -координатно-выпуклых функций, понятие которого введено в [3], на специальном порядково-выпуклом множестве, а также гарантированные оценки выраженные через чебышевскую норму и собственных чисел матрицы входящие условия ограничения. Кроме того установлено устойчивости градиентного алгоритма при "малых" возмущениях некоторых параметров допустимой области.

Пусть Z_+^n - множество всех n -мерных неотрицательных целочисленных векторов. На множестве Z_+^n введем естественный порядок. Будем писать: $x = (x_1, \dots, x_n) \leq y = (y_1, \dots, y_n)$, если $x_i \leq y_i$, $i = 1, 2, \dots, n$. Если среди последних неравенств есть хотя бы одно строгое, то будем писать $x < y$. Множество $P \subseteq Z_+^n$ называется порядково-выпуклым [1], если из условия $x \leq y$, $x, y \in P$ следует включение $[x, y] = \{z : x \leq z \leq y, z \in Z_+^n\} \subseteq P$. Будем в дальнейшем предполагать, что порядково-выпуклое множество P конечное и содержит единственный минимальный элемент-нулевой вектор $0 = (0, \dots, 0)$.

Функция $f : Z_+^n \rightarrow R$ (где R множество действительных чисел) называется ρ -

координатно-выпуклой [3], если

$$\Delta_{ij}f(x) \leq 0, \quad \forall x = (x_1, \dots, x_n) \in Z_+^n, \quad i \neq j, \quad i, j \in N_n = \{1, 2, \dots, n\},$$

$$\Delta_{ii}f(x) \leq -\rho_i, \quad \forall x \in Z_+^n, \quad i \in N_n,$$

где

$$\Delta_{ij}f(x) = \Delta_i f(x + e^j) - \Delta_i f(x),$$

$$\Delta_j f(x) = f(x + e^j) - f(x), \quad e^j = (e_1^j, \dots, e_n^j), \quad e_j^j = 1, \quad e_k^j = 0, \quad j \neq k, \quad j, k \in N_n,$$

т.е. e^j — j -й единичный n -мерный орт.

Класс всех ρ -координатно-выпуклых функций на Z_+^n обозначим через $\mathfrak{R}_\rho(Z_+^n)$ [3].

Как обычно, функцию $f : Z_+^n \rightarrow R$ будем называть неубывающей (не возрастающей), если

$$\Delta_i f(x) \geq 0, \quad (\Delta_i f(x) \leq 0), \quad \forall x \in Z_+^n, \quad i \in N_n.$$

Рассмотрим задачу A выпуклой дискретной оптимизации:

$$\max \{F(x) : x \in P\},$$

где $F(x)$ — неубывающая ρ -координатно-выпуклая функция, P — порядково-выпуклое множество в Z_+^n .

Пусть x^* — оптимальное решение задачи A , т.е.

$$F(x^*) = \max \{F(x) : x \in P\}.$$

Пусть x^g — градиентное решение задачи A , т.е. точка, полученная с помощью следующего градиентного алгоритма (покоординатного подъема) [1, 3]:

$$x^{t+1} = x^t + e^{i(t)}, \quad t = 0, 1, \dots, \quad x^0 = 0 = (0, \dots, 0),$$

$$i(t) = \arg \max_i \{ \Delta_i F(x) : i \in \text{fes}(x^t, P) \},$$

$$\text{fes}(x^t, P) = \{i : x^t + e^i \in P, \quad i \in N_n\},$$

заканчивающей свою работу на шаге τ , когда $\text{fes}(x^\tau, P) = \emptyset$.

Под гарантированной оценкой погрешности градиентного алгоритма решения задачи A , как обычно понимают такое число $\varepsilon \geq 0$, что

$$\frac{F(x^*) - F(x^g)}{F(x^*) - F(0)} \leq \varepsilon$$

Пусть $A = (a_{ij})_{n \times n} \in R_+^{n \times n}$, $a_{ij} = a_{ji}$, $\forall (i, j) \in N_n \times N_n$, $x = (x_1, \dots, x_n) \in Z_+^n$, $b \in R_+^1$, $b > 0$, (Ax, x) — скалярное произведение векторов Ax и x . В пространстве $R_+^{n \times n}$ зададим чебышевскую норму [5], то есть под нормой матрицы $A = (a_{ij})_{n \times n} \in R_+^{n \times n}$ будем понимать число $\|A\| = \max \{a_{ij} : (i, j) \in N_n \times N_n\}$.

Далее будем обозначать через P_ψ^- множество

$$P_\psi^- = \left\{ x : \psi(x) = \frac{1}{2}(Ax, x) - b \leq 0, \quad x \in P \subseteq Z_+^n, \quad b > 0, \quad b \in R_+^1 \right\}.$$

В следующем пункте основной целью является получение гарантированных оценок точности градиентного алгоритма в задачах максимизации функции из класса $\mathfrak{R}_\rho(Z_+^n)$ на

множестве P_ψ^- , а также исследование устойчивости градиентного алгоритма при возмущении матрицы $A = (a_{ij})_{n \times n} \in R_+^{n \times n}$ и ее собственных чисел.

2. Оценки

Теорема 1. Если функция $\psi(x) = \frac{1}{2}(Ax, x) - b$ неубывающая, $P \subseteq Z_+^n$ - порядково-выпуклое множество, то множество P_ψ^- является порядково-выпуклым и $-\psi(x) \in \mathfrak{R}_\rho(Z_+^n)$, где $\rho_i = a_{ii}$, $i \in N_n$.

Теорема 2. Пусть $\|A\| > 0$. Тогда гарантированная оценка погрешности градиентного алгоритма решения задачи максимизации неубывающей функции $F(x) \in \mathfrak{R}_\rho(Z_+^n)$ на множестве P_ψ^- выражается формулой

$$\varepsilon = \frac{1}{1 + M(\|A\|, h)}(1 - M(\|A\|, h)B(\rho, h, r)),$$

где

$$M(\|A\|, h) = \begin{cases} (h + \frac{1}{2})\|A\|, & \text{если } (h + \frac{1}{2})\|A\| < 1, \\ \frac{2}{(2h+1)\|A\|}, & \text{если } (h + \frac{1}{2})\|A\| \geq 1, \end{cases} \quad B(\rho, h, r) = \frac{(h-r)^2\omega(\rho)}{h^2\omega_1(\rho)},$$

$$\omega(\rho) = \begin{cases} 0, & \text{если } I_\rho^+ = \emptyset, \\ \left(\sum_{i \in I_\rho^+} 1/\rho_i\right)^{-1}, & \text{если } I_\rho^+ \neq \emptyset, \end{cases} \quad , \quad I_\rho^+ = \{i : \rho_i > 0, i \in N_n\},$$

$$h = h(P) = \max \left\{ h(x) = \sum_{i=1}^n x_i : x = (x_1, \dots, x_n) \in P \right\},$$

$$r = \min \{h(x) - 1 : x \in Z_+^n \setminus P\}, \quad \omega_1(\rho) = 2\Omega(\delta_F) - \omega(\rho), \quad \Omega(\delta_F) = \sum_{i=1}^n (F(e^i) - F(0)).$$

Следствие 1. Пусть $A = (a_{ij})_{n \times n} \in Z_+^{n \times n}$, $a_{ij} = a_{ji}$, $\forall (i, j) \in N_n \times N_n$, $\|A\| > 0$, $\psi(x) = \frac{1}{2}(Ax, x) - b$ - неубывающая функция на множестве $P \subseteq Z_+^n$, $|P| > 1$, P -порядково-выпуклое множество. Тогда гарантированная оценка погрешности градиентного алгоритма решения задачи максимизации неубывающей функции $F(x) \in \mathfrak{R}_\rho(Z_+^n)$ на множестве P_ψ^- выражается формулой

$$\varepsilon = \frac{(2h+1)\|A\|}{2 + (2h+1)\|A\|} \left(1 - \frac{2B(\rho, h, r)}{(2h+1)\|A\|} \right)$$

Теорема 3. Пусть $\lambda_1, \dots, \lambda_n$ собственные числа матрицы $A = (a_{ij})_{n \times n} \in R_+^{n \times n}$, $a_{ij} = a_{ji}$, $\forall (i, j) \in N_n \times N_n$, $\Omega(\lambda) > 0$. Тогда гарантированная оценка погрешности градиентного алгоритма решения задачи максимизации неубывающей функции $F(x) \in \mathfrak{R}_\rho(Z_+^n)$ на множестве P_ψ^- выражается формулой

$$\varepsilon = \frac{1}{1 + M(\Omega(\lambda), h)}(1 - M(\Omega(\lambda), h)B(\rho, h, r)),$$

где $\Omega(\lambda) = \sum_{i=1}^n \lambda_i$, $\lambda = (\lambda_1, \dots, \lambda_n)$, остальные все обозначения имеют такой же смысл, как в теореме 2.

Следствие 2. Пусть $A = (a_{ij})_{n \times n} \in Z_+^{n \times n}$, $a_{ij} = a_{ji}$, $\forall (i, j) \in N_n \times N_n$, $\Omega(\lambda) > 0$, $\psi(x) = \frac{1}{2}(Ax, x) - b$ - неубывающая функция на множестве $P \subseteq Z_+^n$, $|P| > 1$, P - порядково-выпуклое множество. Тогда гарантированная оценка погрешности градиентного алгоритма решения задачи максимизации неубывающей функции $F(x) \in \mathfrak{R}_\rho(Z_+^n)$ на множестве P_ψ^- выражается формулой

$$\varepsilon = \frac{(2h+1)\Omega(\lambda)}{2 + (2h+1)\Omega(\lambda)} \left(1 - \frac{2B(\rho, h, r)}{(2h+1)\Omega(\lambda)} \right)$$

Замечание. Если $I_\rho^+ = \emptyset$, то полученные в теоремах 2, 3 оценки совпадают с соответствующими оценками из [1]. Однако при $I_\rho^+ \neq \emptyset$ оценки улучшаются.

Определим множество возмущающих B^δ матриц следующим образом [5]:

$$B^\delta = \{B \in R_+^{n \times n} : 0 < \|B\| < \delta, \delta \in R_+^1\}.$$

Рассмотрим следующие задачи D ДО

$$\max \{F(x) : x \in P_\psi^-\},$$

где функция $F(x)$ и множество P_ψ^- такие же, как в следствии 1.

Под возмущением задачи D по матрице $A = (a_{ij}) \in Z_+^{n \times n}$ будем понимать следующие задачи D^δ

$$\max \{F(x) : x \in P_\psi^\delta\},$$

где $P_\psi^\delta = \{x : \frac{1}{2}((A+B)x, x) - b \leq 0, B \in B^\delta, x \in P\}$, все остальные обозначения имеют такой же смысл, что и в теореме 2.

Пусть $\lambda_1, \dots, \lambda_n$ и μ_1, \dots, μ_n соответственно собственные числа матрицы $A = (a_{ij}) \in Z_+^{n \times n}$ и $C = (c_{ij}) \in R_+^{n \times n}$. Обозначим через C_s^β множество:

$$C_s^\beta = \{C \in R_+^{n \times n} : 0 < \Omega(\mu) < \beta, \beta \in R_+^1, \mu = (\mu_1, \dots, \mu_n)\}.$$

Под возмущением задачи D относительно собственных чисел в матрице $A = (a_{ij}) \in Z_+^{n \times n}$ будем понимать следующие задачи D^β

$$\max \{F(x) : x \in SP_\psi^\beta\},$$

где $SP_\psi^\beta = \{x : \frac{1}{2}((A+C)x, x) - b \leq 0, C \in C_s^\beta, x \in P\}$, все остальные обозначения имеют такой же смысл, что и в теореме 2.

Пусть ε (ε^δ) гарантированная оценка погрешности градиентного алгоритма некоторой исходной (возмущенной) задачи ДО. Как обычно, градиентный алгоритм будем называть устойчивым, если справедливо соотношение $\varepsilon^\delta < \varepsilon K(\delta)$, где $K(\delta) \rightarrow 1$ при $\delta \rightarrow 0$.

Теорема 4. Пусть ε и ε^δ (ε^β) - соответственно гарантированные оценки погрешности градиентного алгоритма для задачи D и D^δ (D^β). Тогда справедливо соотношение

$$\varepsilon^\delta < \varepsilon \left(1 + \frac{\delta}{\|A\|} \right) \left(\varepsilon^\beta < \varepsilon \left(1 + \frac{\beta}{\Omega(\lambda)} \right) \right),$$

то есть градиентный алгоритм для задачи D устойчив относительно возмущения матрицы $A = (a_{ij}) \in Z_+^{n \times n}$ (относительно возмущения собственных чисел матрицы $A = (a_{ij}) \in Z_+^{n \times n}$).

Список литературы

- [1] М.М. Ковалев *Матроиды в дискретной оптимизации*. Изд-во Университетское, Минск. 1987.
- [2] М.М. Ковалев *Исследование по математической экономике в БГУ*. // Вестник Белорусского гос. уни-та. 2001. Сер.1. No 3. С. 44-52.
- [3] V.A. Emelichev, M. M. Kovalev, A. B. Ramazanov *Errors of gradient extrema of a strictly convex function of discrete argument*. // Discrete Mathematics and Applications.1992. vol. 2. № 2. pp. 119-131.
- [4] А.Б. Рамазанов *О гарантированной оценке погрешности градиентного алгоритма для одной задачи выпуклой дискретной оптимизации*.//Вестник Бакинского уни.-та. 2004. № 3. С. 60-66.
- [5] В. А. Емеличев, В. Н. Кричко *Формула радиуса устойчивости векторной l_∞ -экстремальной траекторной задачи* // Дискретная математика. 2004. Т.16. вып. 1. С. 14-20.

ABOUT ESTIMATE OF ERROR OF GREEDY ALGORITHM IN A CONVEX DISCRETE OPTIMIZATION PROBLEM AND SOME NEAR QUESTIONS

A.B. Ramazanov

Baku State University, Baku

e-mail: rab-unibak@rambler.ru , ram-bsu@mail.ru

Abstract. On this paper have been found the new improved posterior guarantee estimate for greedy algorithm for maximization problem of discrete argument strictly coordinate convex functions on special ordered sets. Have been given the guarantee estimate of matrix which admissible solutions have been inserted constraint condition using Chebyshev norm and proper numbers. Have been proved the stability of greedy algorithm relative perturbation of some parameters of admissible domain.

Key words: estimation, greedy, algorithm, convexity, stability.

АЛГОРИТМЫ РЕШЕНИЯ ОДНОЙ ЗАДАЧИ ПОСТРОЕНИЯ ЦИКЛИЧЕСКОГО РАСПИСАНИЯ¹

А.А. Романова

Омский государственный университет, г. Омск
e-mail: romanova_ann@bk.ru

В.В. Сервах

Омский филиал института математики им. С.Л. Соболева СО РАН, г. Омск
e-mail: svv@iitam.omsk.net.ru

Аннотация. Рассматривается задача построения циклического расписания, которая для большинства известных критериев является *NP*-трудной в сильном смысле. Для одной из постановок задачи построена вполне полиномиальная аппроксимационная схема, основанная на алгоритме динамического программирования.

Ключевые слова: сложность задачи, циклические расписания, динамическое программирование, аппроксимационная схема.

Введение

Важную роль в организации производства играют циклические расписания, особенность которых состоит в том, что выполнение одних и тех же операций для двух последовательных деталей происходит через равные промежутки времени для всех операций [1].

Рассматриваемая нами задача с разными критериями исследовалась многими авторами. Для ряда естественных критериев доказано, что задача является *NP*-трудной в сильном смысле [2, 3]. Поэтому целесообразным является построение приближенных и эвристических алгоритмов решения данной задачи. Для одного из критериев для задачи в более общей постановке был предложен алгоритм локального поиска с запретами [1]. В настоящей работе основное внимание будет уделено задаче минимизации циклического времени в случае ограниченного числа деталей, участвующих в цикле. Ранее для этой задачи был предложен метод ветвей и границ [3]. Здесь предложен алгоритм ее решения, основанный на методе динамического программирования. В частном случае, когда фиксировано число деталей, участвующих в цикле, и к концу цикла нет незавершенных операций, этот алгоритм становится псевдополиномиальным. Также построена вполне полиномиальная аппроксимационная схема.

1. Постановка задачи

Рассмотрим задачу построения циклического расписания для производственной линии, состоящей из m различных машин. На этой линии необходимо произвести большую партию однотипных деталей. Все детали проходят одинаковый технологический маршрут обработки, состоящий из n операций. Операция j требует $p_j \in Z^+$ единиц времени выполнения на машине m_j ($j = 1, \dots, n$). Машины в технологическом порядке могут повторяться. Прерывания в выполнении операций недопустимы.

¹Работа поддержана грантом INTAS 03-51-5501.

Дадим определение циклического расписания. Обозначим через $t(j; k)$ время начала выполнения j -ой операции k -ой детали. Расписание является циклическим, если для любого $j = 1, \dots, n$ и для любого $k \in Z$ выполняется $t(j; k) = t(j; k-1) + C$ или $t(j; k) = t_j + Ck$, где $t_j = t(j; 0)$ и C - время цикла или циклическое время. Циклическое расписание определяется заданием времени начала выполнения каждой операции для одной из деталей. Для определенности будем искать значения t_j ($j = 1, \dots, n$).

Кроме времени цикла, введем еще две характеристики циклического расписания. Это F - время нахождения детали в системе, то есть промежуток времени между началом выполнения первой операции детали и завершением выполнения последней операции, и H - число деталей, находящихся в системе в течение цикла.

Рассмотрим следующие варианты задач построения циклического расписания: 1) минимизация циклического времени C ; 2) одновременная минимизация C и F ; 3) минимизация F при ограниченном C ; 4) минимизации C при ограниченном H .

Первая задача полиномиально разрешима [1]. Минимальное время цикла при этом равно $\max_{i=1, \dots, m} \sum_{j \in N_i} p_j$, где N_i - множество операций, выполняемых на машине i ($i = 1, \dots, m$).

Недостаток циклического расписания с минимальным временем цикла в том, что детали при выполнении часто простаивают. Это требует транспортировки и хранения незавершенных деталей, большей потребности в оборотных ресурсах. В [4] построен класс примеров, в которых время F нахождения детали в системе очень большое, причем незначительное увеличение времени цикла приводит к существенному уменьшению F . Таким образом, кроме производительности линии, целесообразно рассматривать и другие критерии.

С остальными критериями задача становится NP -трудной в сильном смысле [2, 3].

2. Задача минимизации времени цикла при ограниченном количестве обрабатываемых деталей в течение цикла

Характеристика H становится важной, когда на производственной линии нет позиций для хранения незавершенных деталей, либо число этих позиций ограничено.

Утверждение 1. В циклическом расписании величины F , C и H связаны соотношением $H = \lceil \frac{F}{C} \rceil$.

Доказательство: Величина H показывает, сколько последовательных деталей участвует в цикле. Расписание является циклическим, поэтому, если деталь начала выполняться в цикле с номером T , то закончит свое выполнение она в цикле с номером $T + H$. Таким образом, величину H можно понимать и как количество циклов, на которое растягивается выполнение детали. Отсюда следует, что время нахождения детали в системе лежит в промежутке $(H - 1)C < F \leq HC$. Поэтому $H = \lceil \frac{F}{C} \rceil$. Утверждение 1 доказано.

Таким образом, при ограничении величины H одновременно оптимизируются F и C . В связи с этим рассмотрим следующую задачу:

$$\begin{cases} C \rightarrow \min \\ H \leq h' \end{cases} \quad (1)$$

Ее NP -трудность в сильном смысле фактически означает, что поиск точного решения задачи в худшем случае требует экспоненциальных от размерности задачи затрат времени, если $P \neq NP$. Поэтому встает вопрос о построении малотрудоемких приближенных алгоритмов. Прежде всего, рассмотрим вопрос о существовании вполне полиномиальной аппроксимационной схемы (FPTAS).

Под ρ -приближенным алгоритмом будем понимать алгоритм, находящий приближенное решение, значение целевой функции которого не более, чем в ρ раз превышает значение целевой функции оптимального решения (если задача разрешима). Соответствующее решение будем называть ρ -приближенным решением.

Под вполне полиномиальной аппроксимационной схемой (*FPTAS*) будем понимать семейство $(1 + \varepsilon)$ -приближенных алгоритмов при всевозможных $0 < \varepsilon < 1$ с временной сложностью, полиномиально зависящей от длины входа задачи и $1/\varepsilon$.

Утверждение 2. *Для задачи (1) не существует FPTAS, если $P \neq NP$.*

Доказательство: Как показано в [5], всякая *NP*-трудная в сильном смысле задача не имеет *FPTAS*, если а) значение целевой функции у каждого допустимого решения целочисленно; б) существует многочлен R такой, что для любого примера I задачи выполняется $F^*(I) \leq R(|I|_{unary})$, где $F^*(I)$ - оптимальное значение целевой функции задачи I , а $|I|_{unary}$ - длина входа этой задачи в унарной записи. Первое условие в нашем случае, очевидно, выполняется. Найдем многочлен R : $|I|_{unary} = \sum_{j=1}^n p_j + h'$. $C^*(I) \leq \sum_{j=1}^n p_j$, так как в крайнем случае длина цикла равна сумме длительностей всех операций. $C^*(I) \leq \sum_{j=1}^n p_j \leq \sum_{j=1}^n p_j + h'$. Отсюда, полагая, что $R(x) = x$, заключаем, что задача не имеет *FPTAS*, если $P \neq NP$. Утверждение 2 доказано.

Утверждение 3. *При $h' \geq n$ задача (1) полиномиально разрешима.*

Доказательство очевидно, так как в этом случае все операции в цикле могут быть из разных деталей, а между этими операциями нет отношений предшествования, поэтому достаточно распределить их по машинам, на которых они должны выполняться, причем в любом порядке. Это все можно сделать за полиномиальное время.

В следующем параграфе предлагается алгоритм для решения задачи (1), который при фиксированном $h' < n$ является псевдополиномиальным.

3. Алгоритм решения задачи с ограничением на H

Вначале рассмотрим классическую задачу обработки деталей с разными технологическими маршрутами (*Job Shop*). Имеется K деталей, при обработке которых используется m машин. Деталь k имеет свой технологический маршрут обработки, состоящий из n_k последовательно выполняемых операций ($k = 1, \dots, K$). Известны длительности выполнения каждой операции и машина, на которой она выполняется. Машины в технологическом порядке могут повторяться. Прерывания в выполнении операций запрещены. Необходимо составить расписание, в котором обработка всех деталей завершается как можно раньше.

Для решения этой задачи применяется алгоритм, основанный на методе динамического программирования. Он использует геометрическую интерпретацию задачи и достаточно широко представлен в литературе [6, 7]. Далее будем обозначать этот алгоритм через ДП. Приведем оценку его сложности. Пусть S_k - суммарная длительность всех операций детали k . Тогда трудоемкость этого алгоритма составит $O(2^K K m S_1 S_2 \dots S_K)$.

Теперь рассмотрим задачу (1). Сначала опишем идею алгоритма. Нам необходимо, чтобы в течение одного цикла был выполнен весь набор операций, но разные операции могут принадлежать разным деталям. Причем в одном цикле могут присутствовать операции не более, чем h' различных деталей. Очевидно, что если в цикле выполняются операции j и k , принадлежащие одной детали, то и все операции $j + 1, j + 2, \dots, k - 2, k - 1$ должны быть выполнены в этом же цикле. Поэтому из различных деталей мы должны взять различные цепочки операций. Учитывая, что цикл можно начать с любой операции, то действуем следующим образом. Сначала в технологическом маршруте идут операции первой дета-

ли, потом в каком-то месте происходит переход на другую деталь (при этом продолжение выполнения первой детали происходит в следующем цикле), потом - на следующую и т.д. Между операциями различных деталей нет ограничений предшествования. Это значит, что технологический маршрут разбивается на h' различных цепей. Если с каждой цепью связать деталь, то получаем уже рассмотренную выше задачу *Job Shop* с h' деталями и m машинами. Ее трудоемкость составляет не более $O(h'm(2P)^{h'})$ операций, так как $S_k \leq P$, где $P = \sum_{j=1}^n p_j$. Осталось только выбрать, в каких местах делать переход. При условии, что к концу цикла нет незавершенных операций, всего мест, в которых можно его сделать, $n - 1$, а переходов $h' - 1$. Таким образом, способов выбора набора переходов $C_{n-1}^{h'-1}$ штук, то есть не более $O(n^{h'})$ штук. Теперь опишем алгоритм более формально.

Алгоритм \mathcal{A}

1. Нахождение времени цикла C^* .

$C(n; k)$ - множество выборок из n по k элементов. В начале полагаем $C^* = P$; $E^* = \emptyset$.

Цикл от $v = 1$ до $C_{n-1}^{h'-1}$

1. Взять выборку $E^v = \{j_1; j_2; \dots; j_{h'-1}\} \in C(n - 1; h' - 1)$;

2. Сформировать h' цепей $E_k^v = \{(j_{k-1} + 1) \rightarrow (j_{k-1} + 2); \dots; (j_k - 1) \rightarrow j_k\}$; $k = 1, \dots, h'$; $j_0 = 0$; $j_{h'} = n + 1$.

3. Решить задачу *Job Shop* с h' деталями, задаваемыми технологическими маршрутами E_k^v ($k = 1, \dots, h'$) и m машинами с помощью алгоритма ДП из [6, 7]. Пусть s_j - время начала выполнения j -ой операции в полученном расписании, а C - общее время обслуживания.

Если $C < C^*$, то $C^* = C$; $E^* = E^v$; (обновление рекорда) **Конец**.

2. Составление расписания. $r = 0$; $t_1 = s_1$;

Цикл от $k = 1$ до $n - 1$

Если $(k \rightarrow k + 1) \in E^*$, то $r := r + 1$;

$t_{k+1} = s_{k+1} + rC^*$; **Конец**.

Теорема 1. Алгоритм \mathcal{A} решает задачу (1) за $O(mh'(2Pn)^{h'})$ операций.

Доказательство следует из корректности оценки для алгоритма динамического программирования [6, 7] и перебора всех возможных вариантов наборов переходов. Следует заметить, что, если существует оптимальное решение, в котором в течение цикла присутствуют детали в количестве, меньшем h' , то алгоритм \mathcal{A} найдет такое же по длине расписание, просто некоторые переходы будут лишними.

Следствие 1. При фиксированном $3 \leq h' < n$ задача (1) псевдополиномиально разрешима за время $O(m(Pn)^{h'})$.

Заметим, что при $h' = 2$ результат Следствия 1 может быть усилен.

Утверждение 4. При $h' = 2$ задача (1) полиномиально разрешима за время $O(n^3 \log_2 n)$.

Доказательство: Задача *Job Shop* с двумя деталями полиномиально разрешима за время $O(r^2 \log_2 r)$, где $r = \max\{r_1, r_2\}$, r_j - число операций для j -ой детали ($j = 1, 2$) [8]. В нашем случае $r < n$. Переход в случае $h' = 2$ возможен только один, таким образом, всего вариантов переходов $n - 1$. Следовательно, итоговая трудоемкость составляет $O(n^3 \log_2 n)$. Утверждение 4 доказано.

4. Вполне полиномиальная аппроксимационная схема

Теорема 2. Для задачи (1) при фиксированном $3 \leq h' < n$ существует вполне полиномиальная аппроксимационная схема.

Доказательство: Применим подход, основанный на округлении входных данных (см.,

например, [9]). Фиксируем $\varepsilon > 0$. Пусть $p_{\max} = \max_{j=1, \dots, n} p_j$. Вычислим коэффициент округления $\delta = p_{\max} \varepsilon / n$ и рассмотрим упрощенную задачу с длительностями $\bar{p}_j = \lceil p_j / \delta \rceil$. Обозначим OPT и \overline{OPT} - оптимальные значения целевой функции исходной и упрощенной задач соответственно. Вычислим трудоемкость решения упрощенной задачи алгоритмом \mathcal{A} . $\bar{P} = \sum_{j=1}^n \bar{p}_j \leq n \overline{p_{\max}} \leq (\frac{p_{\max}}{\delta} + 1)n \leq \frac{n^2}{\varepsilon} + n$. Таким образом, упрощенная задача решается за $O(m(n\bar{P})^{h'}) = O(\frac{mn^{3h'}}{\varepsilon^{h'}})$ операций. Если \bar{s}_j - время начала выполнения операции j в упрощенной задаче, то расписание исходной задачи задается величинами $s_j = \delta \bar{s}_j$. Очевидно, оно остается допустимым, так как исходные длительности не больше измененных. Могут быть простои, от которых можно избавиться, сдвинув некоторые операции на более ранний срок. Если обозначить через L_i время окончания выполнения работ на i -ой машине в приближенном решении, то для любого $i = 1, \dots, m$ ее можно оценить следующим образом: $L_i \leq \delta \bar{L}_i \leq \delta \overline{OPT} \leq \delta n + OPT = \frac{p_{\max} \varepsilon}{n} n + OPT \leq OPT \varepsilon + OPT = OPT(1 + \varepsilon)$. Поясним неравенство $\delta \overline{OPT} \leq \delta n + OPT$. Рассмотрим оптимальное расписание исходной задачи. Заменим в этом расписании каждую операцию с длительностью p_j на операцию с длительностью $\delta \bar{p}_j$, сдвинув выполнение операций так, чтобы оставить расписание допустимым. Длина расписания увеличилась не более, чем на δn . С другой стороны, $\delta \overline{OPT}$ - минимальная длина расписания с длительностями $\delta \bar{p}_j$, поэтому $\delta \overline{OPT} \leq \delta n + OPT$. Так как $L_i \leq OPT(1 + \varepsilon)$ для любого i , то длина расписания в приближенном решении $\max_{i=1, \dots, m} L_i \leq (1 + \varepsilon)OPT$. Теорема 2 доказана.

Список литературы

- [1] P. Brucker, T. Kampmeyer *Tabu search algorithms for cyclic machine scheduling problems*. Osnabrueck, Preprint, 2002, p. 24.
- [2] R. Roundy *Cyclic schedules for job shops with identical jobs*. – Mathematics of Operations Research, 1992, Vol. 17, N 4, p. 842-865.
- [3] С. Ханен *Study of a NP-hard cyclic scheduling problem: The recurrent job-shop*. – European Journal of Operational Research, 1994, Vol. 72, p. 82-101.
- [4] А.А. Романова, В.В. Сервах *О построении циклических расписаний для задачи обработки однотипных деталей*. – Новосибирск, Материалы конференции "Дискретный анализ и исследование операций", 2004, с. 175.
- [5] M.R. Garey, D.S. Johnson *Strong NP-completeness results: Motivation, examples, and implications*. – Journal of the ACM, 1978, Vol. 25, p. 499-508.
- [6] S.E. Akers *A graphical approach to production scheduling problems*. – Operations Research, 1956, V. 4, N 2, p. 244-245.
- [7] W.W. Hardgrave, G.L. Nemhauser *A geometric model and a graphical algorithm for a sequencing problem*. – Operations Research, 1963, V. 11, N 6, p. 889-900.
- [8] Н.И. Глебов *Алгоритм составления расписания для двух работ*. – Управляемые системы, 1968, Вып. 1, с. 14–20.
- [9] Г.В. Генс, Е.В. Левнер *Эффективные приближенные алгоритмы для комбинаторных задач*. Москва: ЦЭМИ, Препринт, 1981.

ALGORITHMS FOR SOLVING SOME CYCLIC JOB SHOP PROBLEM

A.A. Romanova

Omsk State university, Omsk
e-mail: romanova_ann@mail.ru

V.V. Servakh

Omsk Branch of Sobolev Institute of Mathematics SB RAS, Omsk
e-mail: svv@iitam.omsk.net.ru

Abstract. We consider Cyclic Job Shop Problem with several objective functions. For most of them the problem is *NP*-hard in strong sense. For one special case of Cyclic Job Shop problem we propose pseudo-polynomial algorithm, which is based on the dynamic programming. Also for this special case we construct fully polynomial time approximation scheme.

Key words: complexity, Cyclic Job Shop problem, dynamic programming, approximation scheme.

ПРИБЛИЖЕННЫЙ ВЕРОЯТНОСТНЫЙ АЛГОРИТМ ДЛЯ ОДНОЙ ЗАДАЧИ АППРОКСИМАЦИИ ГРАФОВ

А.С. Талевнин

Омский государственный университет, Омск
e-mail: talants@rambler.ru

Аннотация. В работе рассматривается следующая задача аппроксимации графов: дан неориентированный граф G , требуется найти ближайший к нему граф M на том же множестве вершин, имеющий не более двух компонент связности, каждая из которых есть полный граф. Под расстоянием между графами понимается число несовпадающих в них ребер. Вычислительная сложность этой задачи неизвестна. В работе описывается вероятностный приближенный алгоритм, который находит приближенное решение этой задачи с гарантированной погрешностью, не превосходящей константы.

Ключевые слова: оптимизация, граф, разбиение, аппроксимация, вероятность, алгоритм.

Введение

Задача аппроксимации графов возникает при анализе систем взаимосвязанных объектов, в частности, в задачах классификации. При этом минимизируется число связей между классами плюс число недостающих связей внутри классов. Постановки и различные интерпретации этой задачи можно найти в работах [1],[2]. Во взвешенном эта задача NP -трудна [3], для невзвешенного варианта этот вопрос остается открытым.

В работе предлагается семейство вероятностных алгоритмов, зависящих от параметра, каждый из которых с вероятностью $1 - O(1/n^k)$, где k – некоторая константа строит решение задачи аппроксимации, отличающееся не более, чем в константу раз от оптимального. При этом, чем выше гарантированная точность получаемого решения, тем больше время работы алгоритма.

1. Постановка задачи и некоторые свойства ее решений

Обыкновенный граф называется M -графом, если каждая его компонента связности есть полный граф. Обозначим через $\mathcal{M}_2(V)$ – класс всех M -графов на множестве V , имеющих не более двух компонент связности, $|V| = n$. Если G_1 и G_2 – графы на одном и том же множестве вершин V , то *расстояние* между ними определяется как $\rho(G_1, G_2) = |E(G_1) \setminus E(G_2)| + |E(G_2) \setminus E(G_1)|$, то есть $\rho(G_1, G_2)$ равно числу несовпадающих ребер в графах G_1 и G_2 .

Сформулируем один из вариантов задачи аппроксимации графов.

Задача А. Для произвольного обыкновенного графа $G = (V, E)$ найти такой граф $M^* \in \mathcal{M}_2(V)$, что $\rho(G, M^*) = \min_{M \in \mathcal{M}_2(V)} \rho(G, M)$.

В данной работе рассматривается эквивалентная постановка этой задачи: для данного графа $G = (V, E)$ найти разбиение множества вершин (V_L, V_R) , которое минимизирует

функцию

$$f(V_L, V_R) = |\{(u, v) \in E : u \in V_L, v \in V_R\}| + |\{(u, v) \notin E : u, v \in V_i, i \in \{L, R\}\}|.$$

Каждой вершине $v \in V$ графа G при фиксированном решении (V_L, V_R) поставим в соответствие две величины:

$$b_L(v, V_L, V_R) = |N(v) \cap V_R| + |\overline{N}(v) \cap V_L|,$$

$$b_R(v, V_L, V_R) = |N(v) \cap V_L| + |\overline{N}(v) \cap V_R|,$$

где $N(v) = \{u \in V : (u, v) \in E\}$, а $\overline{N}(v) = V \setminus (N(v) \cup \{v\})$. Можно отметить, что

$$\sum_{v \in V_L} b_L(v, V_L, V_R) + \sum_{v \in V_R} b_R(v, V_L, V_R) = 2f(V_R, V_L), \quad (1)$$

а $b_i(v, V_L, V_R)$ можно понимать как штраф, вносимый вершиной $v \in V_i, i \in \{L, R\}$ в расстояние до M -графа, компоненты связности которого определяются множествами V_L и V_R .

Лемма 1. Пусть (V_L^*, V_R^*) – оптимальное разбиение множества вершин V . Тогда для любой вершины $v \in V_i^*$ верно неравенство $b_i(v, V_L^*, V_R^*) \leq (n - 1)/2$, где $i \in \{L, R\}$.

Доказательство. Покажем, что для любого решения (V_L, V_R) задачи аппроксимации выполнено 1) если $v \in V_L$, то $f(V_L \setminus \{v\}, V_R \cup \{v\}) = f(V_L, V_R) + b_R(v, V_L, V_R) - b_L(v, V_L, V_R)$; 2) если $v \in V_R$, то $f(V_L \cup \{v\}, V_R \setminus \{v\}) = f(V_L, V_R) + b_L(v, V_L, V_R) - b_R(v, V_L, V_R)$.

Докажем утверждение 1, доказательство утверждения 2 аналогично. Для вершин $u \in V \setminus \{v\}$ возможны четыре варианта:

- если $u \in V_L \cap N(v)$, то $b_L(u, V_L \setminus \{v\}, V_R \cup \{v\}) = b_L(u, V_L, V_R) + 1$;
- если $u \in V_L \cap \overline{N}(v)$, то $b_L(u, V_L \setminus \{v\}, V_R \cup \{v\}) = b_L(u, V_L, V_R) - 1$;
- если $u \in V_R \cap N(v)$, то $b_R(u, V_L \setminus \{v\}, V_R \cup \{v\}) = b_R(u, V_L, V_R) - 1$;
- если $u \in V_R \cap \overline{N}(v)$, то $b_R(u, V_L \setminus \{v\}, V_R \cup \{v\}) = b_R(u, V_L, V_R) + 1$.

Таким образом, с учетом (1) получаем, что

$$\begin{aligned} 2f(V_L \setminus \{v\}, V_R \cup \{v\}) &= \sum_{u \in V_L \setminus \{v\}} b_L(u, V_L \setminus \{v\}, V_R \cup \{v\}) + \\ &\sum_{u \in V_R \cup \{v\}} b_R(u, V_L \setminus \{v\}, V_R \cup \{v\}) = 2f(V_L, V_R) + 2(b_R(v, V_L, V_R) - b_L(v, V_L, V_R)). \end{aligned}$$

Если теперь заметить, что для любого решения (V_R, V_L) справедливо равенство $b_L(v, U) + b_R(v, U) = n - 1$, то утверждение леммы становится очевидным. ■

Приведем один из известных результатов [4] для рассматриваемой задачи:

$$f^* = f(V_L^*, V_R^*) \leq \left\lfloor \frac{(n - 1)^2}{4} \right\rfloor, \quad (2)$$

где (V_L^*, V_R^*) – оптимальное разбиение множества вершин V .

2. Приближенный алгоритм для задачи А

Идея предлагаемого метода возникла в результате анализа приближенного алгоритма для задачи о сбалансированном разрезе (bisection), предложенном в работе [5]. Для

произвольного $\delta \in (0, 1)$ и достаточно большой константы $c > 0$ опишем вероятностный алгоритм для задачи аппроксимации графов.

Алгоритм $\mathbf{A}(\delta, c)$.

1. Случайным образом, используя равномерное распределение, выбрать множество вершин $S \subset V$, $|S| = c \ln n$.
2. Для каждого разбиения (S_L, S_R) множества S , строим разбиение (V_L, V_R) множества V .
 - 2.1. $T_L = \{v \in V \setminus S : |N(v) \cap S_R| + |\overline{N}(v) \cap S_L| < \frac{1+\delta}{4}|S|\} \cup S_L$,
 $T_R = \{v \in V \setminus S : |N(v) \cap S_L| + |\overline{N}(v) \cap S_R| < \frac{1+\delta}{4}|S|\} \cup S_R$.
 - 2.2. Для любой вершины $v \in V \setminus (T_L \cup T_R)$ вычислим:
$$b_L(v, T_L, T_R) = |N(v) \cap T_R| + |\overline{N}(v) \cap T_L|,$$

$$b_R(v, T_L, T_R) = |N(v) \cap T_L| + |\overline{N}(v) \cap T_R|.$$
 - 2.3. $V_L \leftarrow T_L \cup \{v \in V \setminus (T_L \cup T_R) : b_L(v, T_L, T_R) < b_R(v, T_L, T_R)\}$,
 $V_R \leftarrow T_R \cup \{v \in V \setminus (T_L \cup T_R) : b_R(v, T_L, T_R) \leq b_L(v, T_L, T_R)\}$.
3. Выбираем лучшее среди $2^{|S|} = O(n^{c \ln 2})$ разбиений (V_L, V_R) .

Замечание 1. При выборе множества вершин S не исключается возможность того, что некоторые вершины попадут в S не единожды. При этом если находится такая вершина, то на шаге 2.1 в сумме $|N(v) \cap S_i| + |\overline{N}(v) \cap S_j|$, где $\{i, j\} = \{L, R\}$ она считается столько раз, сколько вхождений имеет во множество S .

Замечание 2. Среди всех разбиений (S_L, S_R) множества S существует пара (S_L^*, S_R^*) , соответствующая оптимальному решению (V_L^*, V_R^*) . Для нее обозначим множества, получаемые на шаге 2.1, через T_L^* и T_R^* соответственно.

Пусть (V_L, V_R) – разбиение множества V , полученное алгоритмом $\mathbf{A}(\delta, c)$. Введем еще два множества вершин $R_L = \{v \in V : b_L(v, V_L^*, V_R^*) \leq \delta(n-1)/2\}$ и $R_R = \{v \in V : b_R(v, V_L^*, V_R^*) \leq \delta(n-1)/2\}$. Из леммы 1 следует, что $R_L \subseteq V_L^*$ и $R_R \subseteq V_R^*$ для $\delta \in (0, 1)$.

Для дальнейших рассуждений нам понадобится следующая оценка Чернова (см. [6]).

Лемма 2. Пусть $c > 0$ и $d > 0$ некоторые константы. Рассмотрим совокупность независимых случайных величин ξ_j ($j = 1, \dots, c \ln n$), имеющих одинаковое распределение: $P\{\xi_j = a_i\} = 1/n$, где $a_i \in \{0, 1\}$, $i = 1, \dots, n$. Тогда имеет место следующее неравенство

$$P\left\{\left|\frac{\sum_{j=1}^s \xi_j}{s} - \frac{\sum_{i=1}^n a_i}{n}\right| > \sqrt{\frac{d}{c}}\right\} < \frac{2}{n^{d/2}}.$$

Лемма 3. Пусть T_R^* , T_L^* – множества построенные, алгоритмом $\mathbf{A}(\delta, c)$. Тогда с вероятностью не меньшей, $1 - O(1/n^k)$ выполнены включения $R_R \subseteq T_R^* \subseteq V_R^*$, а $R_L \subseteq T_L^* \subseteq V_L^*$, где k – некоторая константа.

Доказательство. Покажем, что вероятность нарушения включений достаточно мала. Вычислим вероятность того, что вершина $v \in T_R^* \setminus V_R^*$. Поставим в соответствие каждой вершине $v \in V_L^*$ набор чисел: $\{a_u : u \in V\}$, где

$$a_u = \begin{cases} 1, & u \in (N(v) \cap V_L^*) \cup (\overline{N}(v) \cap V_R^*) \cup \{v\}, \\ 0, & u \in (N(v) \cap V_R^*) \cup (\overline{N}(v) \cap V_L^*). \end{cases}$$

Положим ξ_j – исход в j -ом испытании ($j = 1, \dots, |S|$), проведенном на шаге 1 алгоритма $\mathbf{A}(\delta, c)$. Так как $v \notin V_R^*$, то $v \notin S$, а значит $\sum_{j=1}^s \xi_j = |N(v) \cap S_L^*| + |\bar{N}(v) \cap S_R^*|$. Поскольку $v \in T_R^*$, то $\sum_{j=1}^s \xi_j < (1 + \delta)|S|/4$, а так как $v \notin V_R^*$, то по лемме 1 $\sum_{u \in V} a_u = b_R(v, V_L^*, V_R^*) + 1 \geq (n + 1)/2 > n/2$. Следовательно

$$\frac{b_R(v, V_L^*, V_R^*) + 1}{n} - \frac{\sum_{j=1}^s \xi_j}{|S|} > \frac{1 - \delta}{4}.$$

По лемме 2 вероятность этого события меньше, чем $2/n^{d/2}$ при $d = c(1 - \delta)^2/16$.

Чтобы оценить вероятность того, что $v \in R_R \setminus T_R^*$, изменим лишь значение a_v в наборе $\{a_u : u \in V\}$. Положим $a_v = 0$, тогда получаем, что $\sum_{u \in V} a_u = b_R(v, V_L^*, V_R^*) \leq \delta(n - 1)/2 < \delta n/2$, а $\sum_{j=1}^s \xi_j \geq (1 + \delta)|S|/4$. Это значит, что

$$\frac{\sum_{j=1}^s \xi_j}{|S|} - \frac{b_R(v, V_L^*, V_R^*)}{n} > \frac{1 - \delta}{4}.$$

Вновь по лемме 2 при тех же значениях констант c и d вероятность такого события будет меньше $2/n^{d/2}$.

Аналогично, можно оценить вероятность выполнения соотношений $R_L \subseteq T_L^* \subseteq V_L^*$. ■

Лемма 4. Пусть $R_R \subseteq T_R^* \subseteq V_R^*$ и $R_L \subseteq T_L^* \subseteq V_L^*$. Тогда $|T_R^* \cup T_L^*| \geq n - 4f^*/(\delta(n - 1))$, где $f^* = f(V_L^*, V_R^*)$.

Доказательство. Пусть $v \in V_R^* \setminus T_R^*$, тогда по условию леммы $v \notin R_R$, а значит $b_R(v, V_L^*, V_R^*) \geq \delta(n - 1)/2$. Аналогично, если $v \in V_L^* \setminus T_L^*$, то $b_L(v, V_L^*, V_R^*) \geq \delta(n - 1)/2$. Используя эти ограничения и равенство (1), получаем:

$$2f^* \geq \sum_{u \in V_L^* \setminus T_L^*} b_L(u, V_L^*, V_R^*) + \sum_{v \in V_R^* \setminus T_R^*} b_R(v, V_L^*, V_R^*) \geq \frac{\delta(n - 1)}{2} |V \setminus (T_R^* \cup T_L^*)|,$$

из которого следует, что $|V \setminus (T_R^* \cup T_L^*)| \leq 4f^*/\delta(n - 1)$. Для завершения доказательства остается вспомнить, что $|V \setminus (T_R^* \cup T_L^*)| = n - |T_R^*| - |T_L^*|$. ■

Теорема 1. Пусть для задачи \mathbf{A} выполнено условие $f^* < \alpha(n - 1)^2$, для некоторой константы α . Тогда с вероятностью, указанной в лемме 3, для решения (V_L, V_R) , найденного алгоритмом $\mathbf{A}(\delta, c)$, справедливо неравенство $f(V_L, V_R) \leq f^*(1 + 4\alpha/\delta^2)$.

Доказательство. Фиксируем множества $T_L^* \subseteq V_L^*$ и $T_R^* \subseteq V_R^*$, оставшееся множество вершин $W = V \setminus (T_L^* \cup T_R^*)$ произвольно разобьем на две части W_L и W_R и рассмотрим решение $(T_L^* \cup W_L, T_R^* \cup W_R)$ задачи \mathbf{A} .

Определим функцию для пары множеств (W_L, W_R) :

$$b(W_L, W_R) = \sum_{v \in W_L} b_L(v, T_L^*, T_R^*) + \sum_{v \in W_R} b_R(v, T_L^*, T_R^*),$$

где $b_L(v, T_L^*, T_R^*) = |N(v) \cap T_R^*| + |\bar{N}(v) \cap T_L^*|$, а $b_R(v, T_L^*, T_R^*) = |N(v) \cap T_L^*| + |\bar{N}(v) \cap T_R^*|$. Заметим теперь, что

$$f(T_L^* \cup W_L, T_R^* \cup W_R) = f(T_L^*, T_R^*) + b(W_L, W_R) + f(W_L, W_R).$$

В этом равенстве под $f(T_L^*, T_R^*)$ понимается значение функции f на подграфе, порожденном множеством вершин $T_L^* \cup T_R^*$.

В силу критерия отбора вершин на шаге 2.3 алгоритма $\mathbf{A}(\delta, c)$, можно гарантировать, что $b(V_L \setminus T_L^*, V_R \setminus T_R^*) \leq b(V_L^* \setminus T_L^*, V_R^* \setminus T_R^*)$, а следовательно

$$f(V_L, V_R) = f(V_L^*, V_R^*) + \left(b(V_L \setminus T_L^*, V_R \setminus T_R^*) - b(V_L^* \setminus T_L^*, V_R^* \setminus T_R^*) \right) + \\ f(V_L \setminus T_L^*, V_R \setminus T_R^*) - f(V_L^* \setminus T_L^*, V_R^* \setminus T_R^*) \leq f(V_L^*, V_R^*) + f(V_L \setminus T_L^*, V_R \setminus T_R^*).$$

В силу (2) для n -вершинного графа справедлива оценка $f^* \leq (n-1)^2/4$, поэтому $f(V_L \setminus T_L^*, V_R \setminus T_R^*) \leq (|W|-1)^2/4$, а в силу леммы 4 $|W| \leq 4f^*/(\delta(n-1))$. Таким образом, при $f^* < \alpha(n-1)^2$ имеем

$$f(V_L, V_R) \leq f^* + \frac{(4f^*)^2}{4\delta^2(n-1)^2} \leq f^* \left(1 + \frac{4\alpha}{\delta^2} \right). \blacksquare$$

Теорема 2. С вероятностью, указанной в лемме 3, для решения (V_L, V_R) , полученного алгоритмом $\mathbf{A}(\delta, c)$ справедливо неравенство:

$$f(V_L, V_R) \leq \frac{2f(V_L^*, V_R^*)}{\delta\sqrt{\delta^2 + 8} - \delta^2}.$$

Доказательство. Из соотношения 1 и леммы 1 следует, что $f(V_L, V_R) \leq n(n-1)/4$, а значит при $f(V_L^*, V_R^*) \geq \alpha(n-1)^2$ выполнено следующее неравенство:

$$\frac{f(V_L, V_R)}{f(V_L^*, V_R^*)} \leq \frac{1}{2\alpha}.$$

С учетом результата теоремы 1 выберем значения константы α из соотношения $1/(2\alpha) = 1 + 4\alpha/\delta^2$. Таким образом при $\alpha = (\delta\sqrt{\delta^2 + 8} - \delta^2)/8$, можно гарантировать, что $f(V_L, V_R) \leq 4f(V_L^*, V_R^*)/(\delta\sqrt{\delta^2 + 8} - \delta^2)$ вне зависимости от значения $f(V_L^*, V_R^*)$. \blacksquare

Список литературы

- [1] А.А. Ляпунов *О строении и эволюции управляющих систем в связи с теорией классификации*. Проблемы кибернетики. М.: Наука, 1973, вып. 27, с. 7–18.
- [2] Дж. Кемени, Дж. Снелл *Кибернетическое моделирование*. М.: Сов. радио, 1972.
- [3] В.П. Ильев, А.С. Талевнин *О сложности и приближенном решении задачи аппроксимации*. Всероссийская конференция "Проблемы оптимизации и экономические приложения": Материалы конференции. 2003, с. 93.
- [4] I. Tomescu *Note sur une caractérisation des graphes dont le degré de déséquilibre est maximal*. - Mathematiques et sciences humaines, 1973, v. 11, N42, p. 37–40.
- [5] S. Arora, D. Karger, and M. Karpinski *Polynomial time approximation schemes for dense instances of NP-hard problems* Proc. 27th Ann. ACM Symp. on Theory of Comp, 1995, p. 284–293.
- [6] M. Bellare, J. Rompel *Randomness-efficient oblivious sampling* Proc. 35th Ann. Symp. on Found. of Comp. Science, 1994, p. 276–287.

A RANDOMIZED APPROXIMATION ALGORITHM FOR A GRAPH APPROXIMATION PROBLEM

A.S. Talevnin

Omsk State University, Omsk
e-mail: talants@rambler.ru

Abstract. In the paper, the following graph approximation problem is studied. Given an undirected graph G , one has to find the nearest graph M on the same vertex set with at most two connected components each of which is the complete graph. The distance between the graphs G and M is equal to the number of noncoinciding edges. Complexity of this problem is still open. We present a randomized approximation algorithm that finds a solution guaranteed to be within a constant factor of optimal one.

Key words: optimization, graph, partition, approximation, probability, algorithm

РЕЛАКСАЦИИ ЛАГРАНЖА И L-СТРУКТУРА МНОГОГРАННИКА ЗАДАЧИ ОБ УПАКОВКЕ

И.В. Уразова

Омский государственный университет, Омск
e-mail: urazovainn@mail.ru

Аннотация. В настоящей работе описаны результаты вычислительного эксперимента, в котором предпринята попытка выявить зависимость мощности L -накрытия задачи от той или иной Лагранжевой релаксации. При этом используется, описанный в [2] алгоритм перебора элементов L -накрытия в порядке лексикографического убывания.

Ключевые слова: Лагранжевы релаксации, L -структура многогранника, L -накрытие задачи, задача об упаковке.

Введение

При исследовании задач и алгоритмов целочисленного программирования весьма эффективным оказался метод оценочных разбиений. Наиболее интересные результаты были получены на основе L -разбиения пространства [2]. Поэтому представляет интерес изучение L -накрытий различных классов задач. Среди методов решения задач дискретной оптимизации большое распространение получил метод релаксации Лагранжа, при котором часть ограничений задачи включаются в целевую функцию Лагранжа с соответствующими множителями.

Пусть Ω – некоторое множество из \mathbb{R}^n . Рассматривается задача определения лексикографически максимального элемента в $\Omega \cap \mathbb{Z}^n$. Обозначим её (1). Точки $x, y \in \mathbb{R}^n, x \succ y$ эквивалентны, то есть принадлежат одному классу разбиения, если не существует такой $z \in \mathbb{Z}^n$, что $x \succeq z \succeq y$. Здесь " \succeq " – знак лексикографического сравнения. Данное отношение является отношением эквивалентности, а соответствующее фактор-пространство \mathbb{R}^n/L – L -разбиением пространства \mathbb{R}^n . Элементы L -разбиения называются L -классами. При этом [2]:

- 1) каждая целочисленная точка образует отдельный L -класс;
- 2) остальные L -классы состоят из нецелочисленных точек и называются дробными. Они представляют собой множества следующего вида: $\{x \in \mathbb{R}^n | x_i = a_i, i = 1, \dots, r-1, a_r < x_r < a_r + 1\}$, где a_i – целые числа, $r \in \{1, \dots, n\}$.
- 3) любые два L -класса сравнимы в смысле лексикографического порядка.

L -разбиения пространства \mathbb{R}^n естественным образом индуцируют L -разбиения любого его подмножества Ω . При этом для Ω/L свойства 1)-3) сохраняются. Поскольку лексикографический порядок в \mathbb{R}^n является линейным, то при решении задачи поиска $z^* = \text{lexmax}(\Omega \cap \mathbb{Z}^n)$ особое значение приобретает множество $\Omega_* = \{x : x \in \Omega, x \prec z \ \forall z \in \Omega \cap \mathbb{Z}^n\}$. Фактор-множество Ω_*/L называется L -накрытием задачи (1), а $|\Omega_*/L|$ – обозначает его мощность.

Рассмотрим лексикографическую задачу об упаковке: найти

$$\bar{u}^* = \text{lexmax}(\Omega \cap Z^{n+1}) \quad (2)$$

где множество Ω определяется условиями

$$\begin{aligned} x_0 - \sum_{i=1}^m x_i &= 0 \\ \sum_{i=1}^m a_{ij} x_i &\leq 1, j = \overline{1, n} \\ 0 \leq x_i &\leq 1, i = \overline{1, m} \end{aligned}$$

причем $a_{ij} \in \{0, 1\}$. Соответствующая ей двойственная задача называется задачей о покрытии и определяется как:

$$z^* = \text{lexmin}(\Omega \cap Z^{n+1}) \quad (3)$$

$$\begin{aligned} y_0 - \sum_{j=1}^n y_j &= 0 \\ \sum_{j=1}^n a_{ij} y_j &\geq 1, i = \overline{1, m} \\ 0 \leq y_j &\leq 1, j = \overline{1, n}, \end{aligned}$$

Пусть $A = \|a_{ij}\|, i = \overline{1, m}, j = \overline{1, n}$. Нами рассмотрен частный случай задачи об упаковке, описанный в [3] и определяемый тем, что A – блочно-диагональная матрица, каждый блок которой имеет вид:

$$\begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix}$$

Под линейной релаксацией задач понимают задачи определения лексикографически оптимального элемента множества Ω .

В [3] были получены следующие результаты для задач (2) и (3) с блочно-диагональной матрицей A :

$$\begin{aligned} u^* &= (k, 1, 0, 0, \dots, 1, 0, 0)^T \\ z^* &= (2k, 0, 1, 1, \dots, 0, 1, 1)^T \\ \bar{x} = \bar{y} &= \left(\frac{3k}{2}, \frac{1}{2}, \frac{1}{2}, \dots, \frac{1}{2}\right)^T, \end{aligned}$$

где \bar{x}, \bar{y} – оптимальные решения линейных релаксаций задач (2) и (3), k – число блоков в матрице A .

В настоящей работе описаны результаты вычислительного эксперимента, в котором принята попытка выявить зависимость мощности L -накрытия задачи от той или иной Лагранжевой релаксации. При этом используется, описанный в [2] алгоритм перебора элементов L -накрытия в порядке лексикографического убывания.

Ниже при рассмотрении задачи (2) имеется ввиду, что матрица A имеет блочно-диагональный вид. В настоящей работе рассматривается подход, связанный с применением Лагранжевых релаксаций к задаче (2).

Разобьем ограничения задачи (2) с блочно-диагональной матрицей A на два множества: $A''x \leq 1, A'x \leq 1$. При этом будем полагать, что подматрица A' состоит из k блоков. Лагранжевой релаксацией задачи (2) будет задача вида:

$$Rf_{Lagr} = x_0 \rightarrow \max \quad (4)$$

$$x_0 - \sum_{i=1}^m x_i + \lambda(A'x - 1) = 0$$

$$A''x \leq 1$$

$$0 \leq x \leq 1, \lambda \geq 0$$

Мы будем переносить k блоков матрицы A , находящихся либо внизу (обозн. \underline{k}), либо вверху (обозн. \bar{k}) диагонали. При этом $|\Omega_*/L|_{\underline{k}}$ (или $|\Omega_*/L|_{\bar{k}}$) означает мощность L -накрытия Лагранжевой релаксации с k нижними (соответственно с k верхними) блоками, перенесенными в целевую функцию.

Как известно [4], при λ равном оптимальному решению двойственной задачи, оптимальное решение релаксированной задачи Rf_{Lagr}^* совпадает с оптимумом исходной задачи (2). Причем для задачи об упаковке множители Лагранжа выбираются, исходя из условия $\lambda \geq \bar{y}$, где \bar{y} – оптимальный вектор задачи (3).

Утверждение 1. Пусть дана задача (4), \underline{k}, \bar{k} – количество нижних и верхних блоков, включаемых в функцию Лагранжа из множества ограничений задачи об упаковке с блочно-диагональной матрицей A . Тогда при $\lambda = \frac{1}{2}$ верно следующее:

1. $|\Omega_*/L|_{\underline{k}} \leq |\Omega_*/L|_{\bar{k}}, |\Omega_*/L|_{\underline{k}} \leq |\Omega_*/L|$;
причем при $\underline{k} = \bar{k} = 1$ верна оценка
 $|\Omega_*/L|_{\underline{k}} \leq |\Omega_*/L| \leq |\Omega_*/L|_{\bar{k}}$.
2. Если $\underline{k} = \bar{k} = 1$, то $Rf_{Lagr}^* = \frac{3k}{2}, f_{Lagr}^* = k$;
если $\underline{k} > 1, \bar{k} > 1$, то $Rf_{Lagr}^* = \frac{3k}{2}, f_{Lagr}^* > k$,

где Rf_{Lagr}^* – оптимальное значение целевой функции задачи (4);
 f_{Lagr}^* – оптимальное значение целевой функции задачи (4) с условием целочисленности переменных.

Ниже в таблице приведены некоторые результаты вычислительного эксперимента для задачи об упаковке (4), подтверждающие данное утверждение.

В ней используются следующие обозначения:

$|\Omega_*/L|_{Lagr}$ – мощность L -накрытия релаксированной задачи (4);

x_0^* – оптимальное значение исходной задачи об упаковке (2) с условием целочисленности переменных;

\bar{x}_0^* – оптимальное значение целевой функции задачи об упаковке (2) ЛП;

Релаксированная задача об упаковке (4) с числом блоков $k = 5$.
 Для задачи (2): $|\Omega_*/L| = 33$, $\bar{x}_0^* = 7.5$, $x_0^* = 5$

Таблица 3. $\lambda = \frac{1}{2}$

k	$ \Omega_*/L _{Lagr}$	f_{Lagr}^*	Rf_{Lagr}^*
$\underline{k} = 1$	33	5	7.5
$\bar{k} = 1$	126	5	7.5
$\underline{k} = 2$	5	6	7.5
$\bar{k} = 2$	23	6	7.5
$\underline{k} = 3$	5	6	7.5
$\bar{k} = 3$	32	6	7.5
$\underline{k} = 4$	2	7	7.5
$\bar{k} = 4$	2	7	7.5
$\underline{k} = \bar{k} = 5$	2	7	7.5

Следует отметить, что рост разницы мощностей L -накрытий в задачах (4) и (2) сопровождается увеличением разрыва двойственности между оптимальными значениями задач ЛП и ЦЛП. В качестве рекомендаций в решении задачи (2) методом релаксации Лагранжа хотелось бы добавить, что целесообразнее включать в целевую функцию Лагранжа нижние блоки из ограничений исходной задачи.

Список литературы

- [1] Jeremy F. SHAPIRO. *A survey of Lagrangean techniques for discrete optimization* - Annals of Discrete Mathematics 5(1979) pp.113-138.
- [2] Колоколов А.А. *Регулярные разбиения в целочисленном программировании*. - Методы решения и анализа задач дискретной оптимизации. Омск: 1992. С. 67-89.
- [3] Сайко Л.А. *Анализ L-структуры некоторых задач о покрытии*. - Дискретная оптимизация и анализ сложных систем. Новосибирск: 1989. С. 76-96.
- [4] Еремин И.И. *Теория линейной оптимизации*. Екатеринбург: 1998.
- [5] Схрейвер А. *Теория линейного и целочисленного программирования*. –М.:Мир, 1991. Т.1,2.

LAGRANGIAN RELAXATIONS AND L-STRUCTURE OF PACKING POLYTOPE

I.V.Urazova

Omsk State University, Omsk
e-mail: urazovainn@mail.ru

Abstract. In this article the some results of a computing experiment are shown. In the experiment had made an attempt to recognize the Lagrangian relaxation dependency of L-covering of problem cardinal number. At that it was used the L-covering of problem algorithm enumeration of elements in lexicographically descending order.

Key words: Lagrangean relaxations, L-structure of polytope, L-covering of problem, packing problem

РЕШЕНИЕ ДИСКРЕТНОЙ МИНИМАКСНОЙ ЗАДАЧИ РАЗМЕЩЕНИЯ С ДРЕВОВИДНОЙ СТРУКТУРОЙ СВЯЗЕЙ НА СЕТИ

Д.В. Филимонов

Омский государственный университет, Омск
e-mail: fdvmail@mail.ru

Аннотация. В работе рассматривается минимаксная задача размещения взаимосвязанных объектов в вершинах произвольной сети. В вершинах сети расположены фиксированные объекты. Заданы удельные стоимости связей между новыми и фиксированными объектами, а также между парами новых объектов. Доказано, что в случае произвольной структуры связей между объектами задача является NP -трудной. Предложен полиномиальный алгоритм решения задачи на произвольной сети с древовидной структурой связей между новыми объектами.

Ключевые слова: оптимизация, задача размещения, минимаксная, сеть, дерево.

Введение

Задачи оптимального размещения требуется решать при размещении пунктов обслуживания, проектировании коммуникационных и энергетических систем, конструировании технических устройств и в других ситуациях. Область, в которой размещаются объекты, может иметь различную структуру: метрическое пространство, сеть и т.д. Задачи размещения на сетях могут быть как непрерывными, когда объекты размещаются на дугах сети, так и дискретными, в которых размещение производится только в вершинах [1, 3, 4, 5, 6].

В данной работе исследуются свойства и предлагается алгоритм решения дискретной минимаксной задачи размещения взаимосвязанных точечных объектов. Фиксированные объекты расположены в вершинах сети, заданы удельные стоимости связей между объектами. Необходимо разместить объекты в вершинах так, чтобы максимальное взвешенное расстояние между объектами было минимальным.

В п.1 описана постановка задачи. В п.2 доказано, что дискретная минимаксная задача на сети с произвольной структурой связей между объектами является NP -трудной. Далее рассматриваются дискретная задача с древовидной структурой связей между размещаемыми объектами. В п.3 предложен полиномиальный алгоритм поиска размещения, удовлетворяющего ограничениям на максимальные расстояния между объектами. Решена дискретная минимаксная задача на произвольной сети (п.4).

1. Постановка задачи

Дана связная неориентированная сеть N . Требуется разместить n новых объектов в вершинах сети v_1, \dots, v_m , в каждой из которых расположен фиксированный объект. В одной вершине можно размещать любое количество новых объектов. Обозначим через $d(v_i, v_s)$ длину кратчайшего пути между вершинами v_i и v_s в сети. Введем следующие обозначения: $I = \{1, \dots, m\}$ – множество номеров вершин сети N , $J = \{1, \dots, n\}$ – множество номеров новых объектов.

Пусть w_{ij} – неотрицательные удельные стоимости связей между фиксированным объектом i и размещаемым j , $i \in I$, $j \in J$. Структура связей между новыми объектами

определяется с помощью неориентированной сети $B = (J, A)$, где J – множество ее вершин и A – множество дуг. Длина дуги $(j, k) \in A$ равна v_{jk} – удельной стоимости связи размещаемых объектов j и k между собой.

Размещением объектов назовем однозначное отображение $\pi : J \rightarrow I$, т.е. новый объект j размещается в вершину $v_{\pi(j)}$. Необходимо найти размещение, минимизирующее максимальную стоимость связи между объектами:

$$\max(\max_{(j,k) \in A} v_{jk}d(v_{\pi(j)}, v_{\pi(k)}), \max_{i \in I, j \in J} w_{ij}d(v_i, v_{\pi(j)})) \rightarrow \min_{\pi}. \quad (1)$$

Будем называть задачу (1) дискретной минимаксной задачей.

В [4, 5, 6] рассматриваются непрерывные минимаксные задачи, в которых новые объекты могут размещаться не только в вершинах, но и на дугах сети N . Непрерывная минимаксная задача размещения на сети в случае произвольной сети B является NP -трудной [6].

В работе [3] разработан полиномиальный алгоритм решения задачи (1) в случае, когда N – дерево, B – произвольная сеть, и заданы максимальные расстояния между объектами. В [4, 5] построены полиномиальные алгоритмы для непрерывного варианта такой задачи. В работе [2] предложен полиномиальный алгоритм решения задачи размещения дерева в конечном множестве, в которой минимизируется суммарная стоимость связей.

В данной работе доказана NP -трудность дискретной минимаксной задачи размещения на сети в случае произвольной структуры связей между новыми объектами. Предложен полиномиальный алгоритм решения задачи при условии, что структура связей между размещаемыми объектами определяется древовидной сетью B .

2. Минимаксная задача с произвольной структурой связей

Рассмотрим задачу поиска размещения π , удовлетворяющего ограничениям на максимальные расстояния между объектами:

$$d(v_{\pi(j)}, v_{\pi(k)}) \leq b_{jk}, \quad (j, k) \in A, \quad (2)$$

$$d(v_i, v_{\pi(j)}) \leq c_{ij}, \quad i \in I, \quad j \in J, \quad (3)$$

где c_{ij} и b_{jk} – максимально допустимые расстояния между фиксированным объектом i и новым j , а также размещаемых объектов j и k между собой, соответственно.

В работе [4] доказана следующая

ТЕОРЕМА 1. *Задача определения совместности ограничений на максимальные расстояния (2)-(3) с произвольной структурой связей на сети является NP -трудной.*

Доказательство теоремы 1 проводится путем сведения к дискретной задаче (2)-(3) ее непрерывного варианта, который является NP -трудным [6]. Нами доказана

ТЕОРЕМА 2. *Дискретная минимаксная задача для произвольных сетей N и B является NP -трудной.*

Доказательство. Сведем к задаче, указанной в формулировке теоремы, NP -трудную задачу определения совместности ограничений (2)-(3).

Рассмотрим задачу

$$F(\pi) = \max(\max_{(j,k) \in A} (1/b_{jk})d(v_{\pi(j)}, v_{\pi(k)}), \max_{i \in I, j \in J} (1/c_{ij})d(v_i, v_{\pi(j)})) \rightarrow \min. \quad (4)$$

Заметим, что задачи (1) и (4) совпадают с точностью до обозначений. Пусть π^* – оптимальное решение задачи (4).

Если $F(\pi^*) \leq 1$, то выполнены следующие неравенства:

$$d(v_{\pi^*(j)}, v_{\pi^*(k)})/b_{jk} \leq 1, (j, k) \in A, \quad (5)$$

$$d(v_i, v_{\pi^*(j)})/c_{ij} \leq 1, i \in I, j \in J. \quad (6)$$

Очевидно, из выполнения (5)-(6) вытекает, что π^* является размещением, удовлетворяющим (2)-(3).

Если $F(\pi^*) > 1$, то не существует размещения, удовлетворяющего ограничениям на максимальные расстояния (2)-(3). Действительно, если такое размещение существует, то для него будут выполнены неравенства (5)-(6), следовательно, $F(\pi^*) \leq 1$, противоречие.

Таким образом, решая задачу (1), можно решить NP -трудную задачу определения совместности ограничений на максимальные расстояния (2)-(3), т.е. задача (1) для произвольных сетей N и B также является NP -трудной. Теорема доказана.

В статье [1] предложен алгоритм ветвей и границ для решения дискретной минимаксной задачи на сети с произвольной структурой связей между объектами.

3. Поиск допустимого размещения задачи с древовидной структурой связей

Предлагаемый полиномиальный алгоритм 1 находит допустимое размещение в вершинах произвольной сети N , удовлетворяющее (2)-(3), если B – дерево. Он состоит из двух этапов: прямой ход и обратный ход. По завершении первого этапа устанавливается факт существования допустимого размещения. Если ограничения (2)-(3) совместны, то допустимое размещение находится обратным ходом алгоритма.

Обозначим через D_j множество вершин сети N , в которые допускается размещение объекта j , при условии выполнения ограничений (2)-(3). Пусть \bar{B} – поддереву дерева B с множеством вершин \bar{J} , соответствующим еще не рассмотренным новым объектам.

Этап 1. Прямой ход.

Шаг 1.0. Определим $D_j := \{i \in I : d(v_l, v_i) \leq c_{lj}, \forall l \in I\}$, $j \in J$ и $\bar{J} := J$. На шаг 1.1.

Шаг 1.1. Если $\bar{J} = \emptyset$, то на шаг 1.7, иначе на шаг 1.2.

Шаг 1.2. Пусть $j \in \bar{J}$ – концевая вершина дерева \bar{B} . Если $\exists k \in J \setminus \bar{J} : (j, k) \in A$, то на шаг 1.3, иначе на шаг 1.5.

Шаг 1.3. Если $\exists i \in D_j, k \in J \setminus \bar{J}, (j, k) \in A : \forall l \in D_k b_{jk} < d(v_i, v_l)$, то на шаг 1.4, иначе на шаг 1.5.

Шаг 1.4. $D_j := D_j \setminus \{i\}$, на шаг 1.3.

Шаг 1.5. Если $D_j = \emptyset$, то на шаг 1.6, иначе $\bar{J} := \bar{J} \setminus \{j\}$, на шаг 1.1.

Шаг 1.6. Нет допустимого размещения. Конец.

Шаг 1.7. Допустимое размещение существует. Конец.

При описании этапа 2 без ограничения общности будем полагать, что новые объекты нумеруются в порядке их удаления из множества \bar{J} на шаге 1.5 прямого хода алгоритма.

Этап 2. Обратный ход.

Шаг 2.0. $\pi(n) := i : i \in D_n$. $k := n$. На шаг 2.1.

Шаг 2.1. Если $k = 1$, то на шаг 3, иначе $k := k - 1$ и на шаг 2.2.

Шаг 2.2. $\exists! j > k : (j, k) \in A$. $\pi(k) := i : i \in D_k$ и $d(\pi(j), v_i) \leq b_{jk}$. На шаг 2.1.

Шаг 2.3. Получили допустимое размещение. Конец.

Корректность алгоритма 1 устанавливает следующая

ТЕОРЕМА 3. Алгоритм 1 находит размещение в вершинах сети N , удовлетворяющее ограничениям (2)-(3) или определяет, что допустимого размещения не существует.

Доказательство. Предположим, что первый этап алгоритма 1 закончил работу на шаге 1.6. Это означает, что либо на шаге 1.0 для некоторого нового объекта j было построено $D_j = \emptyset$ (для j не выполнены (3)), либо для j не существует вершины, в которой не нарушаются ограничения (2) на максимальные расстояния с ранее рассмотренными новыми объектами (шаг 1.3). В обоих случаях допустимое размещение в вершинах сети N , удовлетворяющее (2)-(3), не существует.

Если этап 1 закончил работу на шаге 1.7, то на этапе 2 найдено некоторое размещение. Докажем, что оно удовлетворяет ограничениям (2)-(3). Согласно шагам 2.0, 2.2, $\pi(k) \in D_k$, $k \in J$. Из шага 1.0 следует выполнение ограничений (3).

Справедливость ограничений (2) для любой пары $(j, k) \in A$ следует из шага 2.2. В свою очередь, условие $\exists! j > k: (j, k) \in A$ на шаге 2.2 выполняется потому, что на шаге 1.2 выбирается концевая вершина j , а существование $i \in D_k$ гарантируется на шагах 1.3–1.4. При удалении номеров недопустимых вершин из D_j множества D_k допустимых размещений новых объектов $k \in J \setminus \bar{J}$ не зависят друг от друга. Действительно, предположим, что $k_1, k_2 \in J \setminus \bar{J}$: $(j, k_1), (j, k_2) \in A$, и $(k_1, k_2) \in A$, следовательно, мы имеем цикл в дереве B , противоречие. Аналогично, B содержит цикл, если существует связь между новыми объектами из множества $k \in J \setminus \bar{J}$, которые находятся в разных поддеревьях дерева B , образуемых при удалении вершины j .

Далее, предположим для некоторой пары $(j, k) \in A$ не выполнено (2): $b_{jk} < d(v_{\pi(j)}, v_{\pi(k)})$. Тогда на шаге 1.4 имеем $D_j := D_j \setminus \{\pi(j)\}$, что приводит к противоречию с шагом 2.2. Теорема доказана.

УТВЕРЖДЕНИЕ. Трудоемкость алгоритма 1 равна $O(nm^2)$.

Доказательство. Трудоемкость шага 1.0 равна $O(nm^2)$. На каждой итерации (шаги 1.1–1.5) первого этапа алгоритма производится удаление нового объекта из множества \bar{J} , следовательно, количество итераций не превосходит n .

Сеть B является деревом, поэтому каждый объект $k \in J \setminus \bar{J}$ на шагах 1.2–1.3 рассматривается не более одного раза, при обработке нового объекта $j \in \bar{J}$: $(j, k) \in A$. При каждом повторении шагов 1.3–1.4 производится удаление номера вершины сети N из множества D_j , следовательно, количество таких повторений не превосходит m . Трудоемкость однократного выполнения шагов 1.3–1.4 равна $O(m)$, поэтому время одной итерации алгоритма ограничено величиной $O(m^2)$. Трудоемкость первого этапа алгоритма 1 равна $O(nm^2)$.

На этапе 2 выполняется n итераций, время выполнения одной итерации – $O(m)$ (шаг 2.2). Таким образом, итоговая трудоемкость алгоритма 1 равна $O(nm^2)$. Утверждение доказано.

4. Решение минимаксной задачи с древовидной структурой связей

Введем переменную z , равную значению целевой функции (1). Известно следующее эквивалентное представление задачи (1):

$$z \rightarrow \min \tag{7}$$

$$d(v_{\pi(j)}, v_{\pi(k)}) \leq z/v_{jk}, \quad (j, k) \in A, \tag{8}$$

$$d(v_i, v_{\pi(j)}) \leq z/w_{ij}, \quad i \in I, \quad j \in J. \tag{9}$$

При фиксированном значении z ограничения (2)-(3) и (8)-(9) идентичны с точностью до обозначений.

Для решения задачи (7)-(9) предлагается алгоритм 2, идея которого основана на бинарном поиске оптимального значения целевой функции в множестве всех возможных значений.

Схема алгоритма 2.

1) Выписываем множество возможных значений целевой функции (1):

$$\{0\} \cup \{v_{jk}d(v_i, v_l) : (j, k) \in A, i, l \in I, i < l\} \cup \{w_{ij}d(v_i, v_l) : i, l \in I, i \neq l, j \in J\}. \quad (10)$$

Количество элементов в множестве (10) ограничено сверху величиной nm^2 .

2) Упорядочиваем множество (10) по неубыванию. Трудоемкость этого этапа равна $O(nm^2 \log(nm))$.

3) Производим бинарный поиск оптимального значения целевой функции по ранжированным значениям множества (10). Проверка существования допустимого решения при фиксированном z производится на первом этапе алгоритма 1. Если для тестируемого значения z допустимое размещение не существует, то оптимальное значение $z^* > z$, в противном случае $z^* \leq z$. При этом достаточно просмотреть $O(\log(nm))$ значений, следовательно, поиск оптимального значения целевой функции также имеет трудоемкость $O(nm^2 \log(nm))$.

4) Чтобы найти оптимальное размещение, достаточно выполнить второй этап алгоритма 1 для задачи (2)-(3), соответствующей оптимальному значению целевой функции.

Таким образом, дискретная минимаксная задача с древовидной структурой связей между новыми объектами на сети является полиномиально разрешимой, трудоемкость алгоритма ее решения равна $O(nm^2 \log(nm))$.

Список литературы

- [1] Г.Г. Забудский, Д.В. Филимонов *Решение дискретной минимаксной задачи размещения на сети*. – Изв. вузов. Математика, 2004, N 5, с. 33–36.
- [2] А.В. Панюков, Б.В. Пельцвергер *Оптимальное размещение дерева в конечном множестве*. – Журнал вычислительной математики и математической физики, 1988, т. 28, с. 618–620.
- [3] Д.В. Филимонов *Решение дискретной минимаксной задачи размещения на древовидной сети*. – Материалы ежегодного научного семинара аспирантов и студентов-выпускников "Под знаком Σ ", Омск, 2003, с. 58–61.
- [4] Д.В. Филимонов *Исследование и решение минимаксных и минисуммных задач размещения на сетях*. – Автореф. канд. дисс., Иркутск, 2004, 18 с.
- [5] E. Erkut, R.L. Francis, A. Tamir *Distance-constrained multifacility minimax location problems on tree network*. – Networks, 1992, N 22, p.37–54.
- [6] A. Kolen *Location problems on trees and in rectilinear plane*. – Stichting Mathematisch Centrum, Amsterdam, 1982.

SOLVING DISCRETE MINIMAX LOCATION PROBLEM WITH TREE STRUCTURE OF CONNECTIONS ON NETWORK

D.V. Filimonov

Omsk State University, Omsk
e-mail: fdvmail@mail.ru

Abstract. A minimax location problem of connected objects at vertices of an arbitrary network is considered. The fixed objects are assumed to be located at the vertices of the network. The costs of connections between new and fixed objects and between distinct pairs of the new objects are given. It is proved that the problem is *NP*-hard when the structure of connections between objects is arbitrary. An algorithm for solving the minimax problem on arbitrary network with tree structure of connections between the new objects is proposed.

Key words: optimization, location problem, minimax, network, tree.

VARIABLE NEIGHBORHOOD BRANCHING FOR MIXED INTEGER PROGRAMMING

Pierre Hansen¹, Nenad Mladenović^{1,2}, Dragan Urošević³

¹*GERAD and Ecole des Hautes Etudes Commerciales
3000 ch. de la Cote-Sainte-Catherine, Montréal H3T 2A7, Canada
Pierre.Hansen@gerad.ca*

²*School of Mathematics, University of Birmingham
Edgbaston, Birmingham B15 2TT, United Kingdom
N.Mladenovic@bham.ac.uk*

³*Mathematical Institute, Serbian Academy of Science
Kneza Mihajla 35, 11000 Belgrade, Serbia and Montenegro
draganu@mi.sanu.ac.yu*

Introduction

Let us consider a generic Mixed integer program (MIP) with some 0-1 variables of the form [2]:

$$(P) \quad \min c^T x \quad (1)$$

$$Ax \geq b \quad (2)$$

$$x_j \in \{0, 1\} \quad \forall j \in \mathcal{B} \neq \emptyset \quad (3)$$

$$x_j \geq 0, \text{ integer} \quad \forall j \in \mathcal{G} \quad (4)$$

$$x_j \geq 0, \quad \forall j \in \mathcal{C} \quad (5)$$

The variable index set $N = \{1, 2, \dots, n\}$ is partitioned into the three subsets $(\mathcal{B}, \mathcal{G}, \mathcal{C})$, corresponding to binary, general integer and continuous variables.

Local Branching (LB) is a new MIP technique [2]. It uses a general-purpose MIP solver such as CPLEX as a subroutine. Therefore, it is, in principle, an exact method. However, if the total time (t_{max}) allocated to solve a given instance is reached before an optimal solution is found and its optimality proved, LB behaves as a heuristic, i.e., it stops at time t_{max} with the best solution known as output (or, possibly, with no feasible solution). The LB algorithm starts with the original formulation, and CPLEX is called to get a feasible solution \tilde{x}_1 . Then, a new constraint is added to reduce the solution space X , using a distance function $d(x, \tilde{x}_1)$, to the set $X_1 = X \cap N_k(\tilde{x}_1)$, where

$$N_k(\tilde{x}_1) = \{x \mid d(x, \tilde{x}_1) \leq k\}. \quad (6)$$

For the 0-1 MIP (1) - (5) given above, $d(., .)$ represents the Hamming distance and $N_k(\tilde{x}_1)$ may be expressed by the following so-called *local branching constraint*

$$\Delta(x, \tilde{x}_1) = \sum_{j \in \bar{S}} (1 - x_j) + \sum_{j \in \mathcal{B} \setminus \bar{S}} x_j \leq k, \quad (7)$$

where $\bar{S} = \{j \in \mathcal{B} \mid \tilde{x}_1 = 1\}$. If a better solution \tilde{x}_2 is found by CPLEX (within a given time limit t_{node}), a new problem is generated as follows. Relation \leq in (7) is replaced by $>$,

and a new local branching constraint (7) is added, but centered around the new incumbent \tilde{x}_2 ($\Delta(x, \tilde{x}_2) \leq k$). Thus, a feasible set X_2 for this subproblem is given as

$$X_2 = X \cap (N_k(\tilde{x}_2) \setminus N_k(\tilde{x}_1)) = (X \setminus X_1) \cap N_k(\tilde{x}_2).$$

This *branching* procedure is iterated as long as there are improvements in the objective function values within t_{node} time: ℓ new constraints are added in iteration ℓ and the feasible set is

$$X_\ell = (X \setminus X_1 \setminus X_2 \cdots \setminus X_\ell) \cap N_k(\tilde{x}_\ell).$$

If the solution is not improved within the t_{node} time limit, the neighborhood X_ℓ is further reduced by replacing k in (7) with $\frac{k}{2}$ ($\Delta(x, \tilde{x}_\ell) \leq \frac{k}{2}$). This step is called *intensification* step. If CPLEX reports proven infeasibility, or in t_{node} time a feasible solution is not found, then the so-called *diversification* step takes place. Here, the right-hand side value of (7) is increased by $\frac{k}{2}$ ($\Delta(x, \tilde{x}_\ell) \leq k + \frac{k}{2}$), a new constraint $\Delta(x, \tilde{x}_\ell) > 1$ added and all other branching constraints deleted. CPLEX is called just to get a new feasible solution for the branching step that follows again. For the diversification, another (fourth) parameter dv_{max} is used to indicate the maximum number of such steps allowed. If $dv_{max} = \infty$, LB behaves as a pure heuristic. If $dv_{max} < \infty$ and if $t_{max} = \infty$, LB switches to an exact solution method: the remaining time is used for running CPLEX.

In this paper we develop a Variable neighborhood search (VNS) heuristic [5, 4] for solving Mixed-Integer Programs (MIP's). Both LB and VNS use the same tools: CPLEX and the same definition of the neighborhoods around the incumbent. However, our VNS is simpler and more systematic in neighborhood exploration.

1. VNS for MIP

We next develop a General VNS [4] for MIP. A main fact that makes our method possible is the same as in LB: a commercial MIP solver (CPLEX) is used for getting an initial solution, and in Shaking as well as for the neighborhood exploration step. As in LB, we use total time t_{max} as a global stopping rule and the node time t_{node} as a local stopping rule for one call of the mixed integer solver (CPLEX).

We do not fix k , the size of the neighborhood around the incumbent. Instead, in the outer loop, we change it from k_{min} to k_{max} with a given step size k_{step} . In VND the neighborhood size ℓ increases from 1 to ℓ_{max} . However, we make our heuristic more user-friendly by reducing those three parameters to only one (k_{step}) in the following way: (i) we allow increase of the neighborhood size without limit, i.e., the k_{max} parameter is not specified. This size is reduced to k_{min} either when an improved or no feasible solution (in t_{node} time) are found by CPLEX; (ii) we set $k_{min} = k_{step}$; (iii) the ℓ_{max} parameter is not specified either. It gets an initial value of 1 when a better solution than the current one is found. Thus, our VNS uses three parameters: t_{max} (or `total_time_limit`), t_{node} (or `node_time_limit`) and k_{step} . A more detailed description of our heuristic, called VnsBra function is given in Figure 1.

```

function VnsBra(total_time_limit, node_time_limit, k_step, x_opt);
begin
1   TL := total_time_limit; UB := ∞; first := true;
2   stat := MIPSOLVE(TL, UB, first, x_opt, f_opt);
3   x_cur:=x_opt; f_cur:=f_opt;
4   while (elapsedtime < total_time_limit) do
5       cont := true; rhs := 1; first := false;
6       while (cont or elapsedtime < total_time_limit) do
7           TL = min(node_time_limit, total_time_limit-elapsedtime);
8           add local br. constr.  $\Delta(x, x\_cur) \leq rhs$ ; UB := f_cur;
9           stat := MIPSOLVE(TL, UB, first, x_next, f_next);
10          switch stat do
11              case "opt_sol_found":
12                  reverse last local br. constr. into  $\Delta(x, x\_cur) \geq rhs + 1$ ;
13                  x_cur := x_next; f_cur := f_next; rhs := 1;
14              end
15              case "feasible_sol_found":
16                  reverse last local br. constr. into  $\Delta(x, x\_cur) \geq 1$ ;
17                  x_cur := x_next; f_cur := f_next; rhs := 1;
18              end
19              case "proven_infeasible":
20                  remove last local br. constr.; rhs := rhs+1;
21              end
22              case "no_feasible_sol_found":
23                  cont := false
24              end
25          end
26          end
27          if f_cur < f_opt then
28              x_opt := x_cur; f_opt := f_cur; k_cur := k_step;
29          else
30              k_cur := k_cur+k_step;
31          end
32          remove all added constraints; cont := true;
33          while cont and (elapsedtime < total_time_limit) do
34              add constraints  $k\_cur \leq \Delta(x, x\_opt) < k\_cur + k\_step$ ;
35              TL := total_time_limit-elapsedtime; UB := ∞; first := true;
36              stat := MIPSOLVE(TL, UB, first, x_cur, f_cur);
37              remove last two added constraints; cont =false;
38              if stat = "proven_infeasible" or "no_feasible" then
39                  cont :=true; k_cur := k_cur+k_step;
40              end
41          end
42      end
43  end
44 end

```

Figure 1. The overall VNS function VnsBra.

Beside `total_time_limit`, `node_time_limit` and `k_step` whose meaning have already been described, other variables used in the description of `VnsBra` function from Figure 1 are:

- UB - input variable for CPLEX solver which represents the current upper bound;
- *first* - logical input variable for CPLEX solver which is `true` if the first solution lower than UB is asked for in the output; if *first* = `false`, CPLEX returns the best solution found so far;
- TL - maximum time allowed for running CPLEX;
- *rhs* - right hand side of constraint (6); it defines the size of the neighborhood within the inner or VND loop;
- *cont* - logical variable which indicates if the inner loop continues (`true`) or not (`false`);
- *x_opt* and *f_opt* - incumbent solution and corresponding objective function value;
- *x_cur*, *f_cur*, *k_cur* - current solution, objective function value and neighborhood from where VND local search starts (i.e., the distance between the new initial solution and the incumbent is equal to *k_cur*) respectively;
- *x_next* and *f_next* - solution and corresponding objective function value obtained by CPLEX in inner loop;

The initialization step is performed in lines 1 and 2. In line 3 the initial solution becomes both the incumbent (*x_opt*) and starting solution for the VND local search (*x_cur*).

The outer loop starts from line 4. In line 5, initial parameters for the inner loop are defined: the first neighborhood is 1 (*rhs* := 1); a best improvement strategy will be applied (i.e., *first* := `false`). In the computational results section we also tested first improvement VNS. This version is obtained when the command *first* := `false` is removed from line 5.

From line 7 to line 20, statements of VND local search are given. At line 9 a local branching constraint is introduced. The CPLEX solver gives four possible outputs. Only if a feasible solution is not found within the given time limit, is VND terminated. In the other three cases, the inner VND loop continues: if non feasibility is proven, then the neighborhood is increased by 1 (*rhs* := *rhs* + 1); if an optimal or feasible solution is found, which means that its value is better than *f_cur*, it becomes the new current solution, and the search starts again from the first neighborhood (*rhs* := 1).

The *move or not* step of GVNS is presented at lines 21 - 23. If the new local minimum *x_cur* is better than the incumbent, it becomes the new incumbent and the neighborhood for the next shaking step reverts to the smallest size (i.e., `k_min` = `k_step`). Otherwise, the neighborhood is increased by `k_step`.

In the *Shaking step*, given at lines 24 - 31, the first feasible solution (i.e., *first* := `true`) is searched for in the disk around the incumbent solution, with radii *k_cur* and *k_cur* + `k_step`. That solution *x_cur* becomes the initial one for the VND local search. If there is no feasible solution in the current disk, then *k_cur* is increased by `k_step`. Note that our `VnsBra` function switches to an exact algorithm if the condition from line 25 (*elapsedtime* < `total_time_limit`) is replaced by (*k_cur* < `k_max`), with an additional call of CPLEX.

The main idea of both LB and VNS for solving MIP is in fact change of the neighborhood during the search. Therefore, LB could be seen as a specialized variant of VNS (or vice versa):

(i) in LB the local search step is performed in a fixed size neighborhood $k > 1$ (a parameter), instead of $k=1$; (ii) as a consequence of (i), *backward* instead of *forward* VNS (see [3]) is used in the inner loop (i.e., instead of increasing neighborhood by 1 in VND or intensification step, its current size, initially set at k , is reduced by half); (iii) the shaking step of VNS and diversification step of LB differ only in the area from where a random feasible solution is chosen: in LB the area is a disk with radii 1 and $k + dv \cdot \lfloor \frac{k}{2} \rfloor$ where dv is the current number of diversifications (see [2] for details), while in our VNS disk is defined by radii k_{cur} and $k_{cur} + k_{step}$.

Computational results

All programs are written in C++ and experiments conducted on a Pentium 4 computer with 1800 MHz processor and 256 RAM memory.

Balas *et al.* [1] develop the Pivot and Shift (PS) heuristic for MIP, which exploits the well-known Pivot-and-Complement heuristic for pure 0-1 programs. It is "essentially a rounding procedure amended with two variants of a neighborhood search". This search is akin to LB and done with general-purpose problem solver XPRESS. The combined procedure "finds better solutions faster than the MIP solver alone". Among the 12 difficult problems for which the solution process is reported in detail are 7 considered also in [2]. We reproduce the best solution values found in 18 minutes of CPU time on a PC with a 1.7 GH Pentium 4 processor for those problems in the second column of Table 1.

Instance name	Objective value				%gap or difference		
	PS(18')	LB(5h)	VNS(18')	VNS(5h)	PS	VNS(18')	VNS(5h)
sp97ic	442866080.00	436984608.00	435761056.00	429892064.00	1.346	-0.280	-1.623
sp98ic	461222592.00	449915168.00	460654656.00	449226848.00	2.513	2.387	-0.153
rail507	199.00	175.00	192.00	176.00	24	17	1
rail2536c	767.00	691.00	701.00	690.00	76	10	-1
B2C1S1	27500.22	27622.24	33439.48	26073.78	-0.442	21.060	-5.606
roll3000	12936.00	13065.00	13201.14	12890.00	-0.987	1.042	-1.339
glass4	1900018690.00	1587515780.00	1583340670.00	1400013700.00	19.685	-0.263	-11.811

Table 1. Comparison of LB, VNS and recent Pivot and Shift (PS) heuristic.

It appears that for 2 of them, the solutions were better than those obtained by LB, and worse for the 5 other cases (but bear in mind the differences of computing times and machines). To compare the results of VNS we reproduce values obtained after 18 minutes and after five hours of computing time for those problems in columns 4 and 5 respectively and the corresponding % gaps as difference in the two last columns. While again the results of Balas *et al.* [1] are better in 2 cases and worse in 5 out of 7 when computing times are comparable, solution of VNS for long computing times are always better.

References

- [1] E. Balas, S Schmieta and C. Wallace. Pivot and shift - a mixed integer programming heuristic, *Discrete Optimization* 1 (2004) 3-12.
- [2] M. Fischetti and A. Lodi. Local branching. *Mathematical Programming, Ser. B* 98 (2003) 23-47.

- [3] P. Hansen and N. Mladenović, Developments of variable neighborhood search, C. Ribeiro, P. Hansen (eds.), *Essays and surveys in metaheuristics*, pp. 415–440, Kluwer Academic Publishers, Boston/Dordrecht/London, 2001.
- [4] P. Hansen and N. Mladenović (2003). Variable Neighborhood Search, in F. Glover and G. Kochenberger (eds.), *Handbook of Metaheuristics*, pp. 145 - 184, Kluwer Academic Publisher.
- [5] N. Mladenović and P. Hansen, Variable Neighborhood Search, *Computers Oper. Res.* 24 (1997) 1097–1100.

О ПОСТРОЕНИИ НОВЫХ ОТСЕЧЕНИЙ В ЦЕЛОЧИСЛЕННОМ ЛИНЕЙНОМ ПРОГРАММИРОВАНИИ ¹

О.В. Хамисов

Институт систем энергетики им. Л.А. Мелентьева
e-mail: khamisov@isem.sei.irk.ru

Аннотация. В статье предлагается новый способ отсечения нецелочисленной вершины, использующий информацию о всех (в том числе и неактивных в текущей вершине) линейных ограничениях, определяющих допустимое множество в задаче целочисленного линейного программирования.

Ключевые слова: отсечение, вогнутой продолжение.

Введение

Пусть заданы многогранное множество $X \subset E^n$

$$X = \{x : Ax \leq b, 0 \leq x_j \leq 1, j = \overline{1, n}\}, \quad (1)$$

где A - $m \times n$ матрица, $b \in E^m$ и невырожденная вершина $x^0 \in X$, не все компоненты которой целые числа. Требуется построить правильное отсечение, т.е. построить плоскость $\alpha^T x = \beta$ такую, что полупространство $\alpha^T x > \beta$ содержит точку x^0 , а полупространство $\alpha^T x \leq \beta$ содержит все целочисленные точки множества X .

Различные правила построения правильных отсечений известны уже давно. По-видимому, впервые правильные отсечения были построены Д. Данцигом [5] и Р. Гомори [6]. Другие правильные отсечения были предложены Э. Балашем [1] и Ф. Гловером и др. [2]. Как указано в [3] многие правильные отсечения можно построить используя точки пересечения ребер многогранного конуса с вершиной x^0 , образованного активными в точке x^0 ограничениями, с границей множества $D \supset X$ такого, что целочисленные точки множества X принадлежат границе D . В алгоритмах Гомори (см. [8]) в качестве D используется полоса

$$L = \{x : \lfloor x_k^0 \rfloor \leq x_k \leq \lceil x_k^0 \rceil\}$$

для некоторого k . Здесь $\lfloor a \rfloor$ обозначает наибольшее целое, не превосходящее a , а $\lceil a \rceil$ - наименьшее целое большее или равное a . В работах Э. Балаша и Ф. Гловера в качестве D предлагается использовать шар S и двойственный гиперкуб C

$$S = \{x : \sum_{i=1}^n x_i(1 - x_i) \geq 0\}, \quad (2)$$

$$C = \{x : \sum_{i=1}^n |x_i - \frac{1}{2}| \leq \frac{n}{2}\}.$$

¹Работа поддержана грантом РФФИ N 05-01-00587

Заметим, что при построении этих отсечений используется не вся информация о множестве X : на построение отсечений никак не влияют *неактивные* в точке x^0 ограничения. В данной статье предлагается метод построения правильных отсечений, использующий информацию о всех ограничениях множества X .

1. Отсечения по вогнутому продолжению

Обозначим функцию, с помощью которой определяется множество S в (2) через $f(x)$

$$f(x) = \sum_{i=1}^n x_i(1 - x_i). \quad (3)$$

Нетрудно видеть, что $f(x)$ - вогнутая функция. Определим функцию $F(x)$ следующим образом

$$F(x) = \min_{y \in X} \{f(y) + \nabla f(y)^T(x - y)\}, \quad (4)$$

где $\nabla f(y)$ есть градиент функции $f(y)$ в точке y . Функция $F(x)$ - непрерывная вогнутая функция и

$$F(x) = f(x), x \in X, \quad (5)$$

$$F(x) \geq f(x), x \notin X. \quad (6)$$

Функцию $F(x)$ называют вогнутым продолжением функции $f(x)$. Вогнутые продолжения использовались ранее автором при разработке методов отсечения для решения задач глобальной оптимизации [4]. В [4] и [9] доказано, что $F(x)$ есть наилучшее вогнутое продолжение, т.е. не существует другой вогнутой функции $\varphi(x)$ такой, что

$$\varphi(x) = f(x), x \in X$$

и

$$\varphi(\bar{x}) > F(\bar{x})$$

хотя бы для одной точки $\bar{x} \notin X$. Определим множество

$$Z = \{x : F(x) \geq 0\}.$$

В силу (3) и (5) целочисленные точки множества X являются граничными точками множества Z . Следовательно, отсечения, построенные с помощью множества Z , будут правильными.

Оценим сложность вычисления значений функции $F(x)$. Из (3) и (4) следует

$$F(x) = \min_{y \in X} \left\{ \sum_{i=1}^n (y_i - y_i^2 + (1 - 2y_i)(x_i - y_i)) \right\} = \min_{y \in X} \left\{ \sum_{i=1}^n (y_i^2 - 2y_i x_i + x_i) \right\}. \quad (7)$$

Задача (7) есть задача минимизации выпуклой квадратичной функции на выпуклом многогранном множестве.

Если в (7) вместо множества X взять множество

$$P = \{x : 0 \leq x_j \leq 1, j = \overline{1, n}\}, \quad (8)$$

то получим

$$F(x) \geq \min_{y \in P} \left\{ \sum_{j=1}^n (y_j^2 - 2y_j x_j + x_j) \right\} = \sum_{j=1}^n \min_{0 \leq y_j \leq 1} \{y_j^2 - 2y_j x_j + x_j\} = \Phi(x). \quad (9)$$

Минимум y_j^* функции $\varphi(y_j) = y_j^2 - 2y_j x_j + x_j$ на отрезке $[0, 1]$ находится по следующему правилу

$$y_j^* = \begin{cases} 0 & \text{если } x_j < 0 \\ x_j & \text{если } 0 \leq x_j \leq 1. \\ 1 & \text{если } x_j > 1 \end{cases}$$

Соответствующие оптимальные значения имеют вид

$$\varphi(y_j^*) = \begin{cases} x_j & \text{если } x_j < 0 \\ x_j - x_j^2 & \text{если } 0 \leq x_j \leq 1. \\ 1 - x_j & \text{если } x_j > 1 \end{cases}$$

Таким образом

$$\Phi(x) = \sum_{j: x_j < 0} x_j + \sum_{j: 0 \leq x_j \leq 1} (x_j - x_j^2) + \sum_{j: x_j > 1} (1 - x_j).$$

Поскольку $F(x) \geq \Phi(x) \geq f(x)$, то отсечения, построенные с помощью функции $F(x)$ будут глубже, чем отсечения по шару S и отсечения, построенные при помощи функции $\Phi(x)$.

References

- [1] E. Balas *Integerseccion cuts - a new type of cutting planes for integer programming*. Operations Research, vol. 19, N1, 1971, pp. 19-39
- [2] E. Balas, J. V. Bowman, F. Glover, D. Sommer *An intersection cut from the dual of the unit hypercube*. Operations Research, vol. 19, N1, 1971, pp.40-44
- [3] В.П. Булатов *Методы погружения в задачах оптимизации*. Новосибирск, Наука, 1977
- [4] V.P. Bulatov, O.V. Khamisov *The cutting method in E^{n+1} through concave extension for solving global extremum problem*. 21 JAHRESTAGUNG "Mathematische Optimierung", Berlin, 1989, pp. 16-19
- [5] G.B. Dantzig *Note on solving linear programs in integers*. Naval. Res. Log. Quart., vol. 6, N1, 1959, pp. 75-76
- [6] R.E. Gomory *An algorithm for integer solution to linear programs*. Princeton - IBM Mathematics Research Project, Technocal Report N1, November 17, 1958
- [7] R. Horst, H. Tuy *Global Optimization. Deterministic Approaches*. 2-nd edition, Springer-Verlag, 1993

- [8] А.А. Корбут, Ю.Ю. Финкельштейн *Дискретное программирование* М: Наука, 1969
- [9] Н. Туу, Р. Horst *Global optimization. Deterministic approaches* Springer-Verlag, Berlin, 1993

ON CONSTRUCTION NEW CUTS IN INTEGER LINEAR PROGRAMMING

O.V. Khamisov

Institute of Energy Systems, Irkutsk

e-mail: khamisov@isem.sei.irk.ru

Аннотация. We suggest a new kind of cuts in integer programming, which uses information on all (not only active) linear constraints describing feasible domain.

Ключевые слова: cutting plane, concave extension.

О $(n - 1)$ -МЕРНОМ ПРИБЛИЖЕННОМ РЕШЕНИИ n -МЕРНОЙ ЗАДАЧИ О РЮКЗАКЕ ¹

В.Н. Шевченко, А.Ю. Чирков

Нижегородский гос. университет им.Н.И. Лобачевского

e-mail: Shev@uic.nnov.ru

Аннотация. В работе оценивается точность $(n - 1)$ -мерного приближенного решения n -мерной целочисленной задачи о рюкзаке. Под $(n - 1)$ -мерным приближённым решением понимается наилучший по функционалу допустимый вектор, одна из компонент которого равна нулю. Показано, что отношение значения функционала на $(n - 1)$ -мерном приближенном решении к оптимальному значению не меньше $(2^n - 2)/(2^n - 1)$.

Ключевые слова: дискретная оптимизация, задача о рюкзаке, приближённое решение и его точность.

Введение

Множество вещественных чисел обозначим через R , а множество целых чисел – через Z . Пусть M – некоторое числовое множество. Через M_+ обозначим множество неотрицательных чисел из M , через M^n – множество n -элементных наборов с компонентами из M . Элементы множества M^n будем называть векторами или точками. Под скалярным произведением ax векторов a и x будем понимать сумму произведений их компонент, то есть $ax = a_1x_1 + \dots + a_nx_n$.

Для вектора a из R_+^n и числа b из R_+ определим множество $L(a, b) = \{x \mid x \in Z_+^n, ax \leq b\}$. В работе рассматривается следующая задача о рюкзаке $\max_{x \in L(a, b)} cx$. Подчеркнём, что координата вектора x – любое целое неотрицательное числа и может принимать значения больше 1. Оптимальным решением задачи о рюкзаке называется точка из $L(a, b)$, на которой достигается максимум целевой функции cx , а оптимальным значением задачи о рюкзаке – значение целевой функции на оптимальном решении. Обозначим через $L_0(a, b)$ множество точек из $L(a, b)$, имеющих по крайней мере одну нулевую компоненту. Под $(n - 1)$ -мерным приближённым решением n -мерной задачи о рюкзаке будем понимать решение задачи $\max_{x \in L_0(a, b)} cx$, а значение целевой функции на $(n - 1)$ -мерном приближённом решении будем называть $(n - 1)$ -мерным приближённым значением n -мерной задачи о рюкзаке, или просто приближённым значением.

Рассмотрим вопрос о степени близости приближённого значения задачи о рюкзаке к оптимальному значению. Введём величины: $\lambda(a, b, c) = \max_{x \in L(a, b)} cx$ – оптимальное значение задачи о рюкзаке, $\lambda_0(a, b, c) = \max_{x \in L_0(a, b)} cx$ – приближённое значение. Под точностью приближённого значения будем понимать отношение $\alpha(a, b, c) = \lambda_0(a, b, c)/\lambda(a, b, c)$ (если $\lambda(a, b, c) = 0$, или значение целевой функции cx не ограничено на множестве $L(a, b)$, то будем считать, что $\alpha(a, b, c) = 1$). Из приведённого определения следует неравенство $0 < \alpha(a, b, c) \leq 1$.

¹Работа выполнена при финансовой поддержке Российского фонда фундаментальных исследований (проект 05-01-00552-а).

Положим $\alpha_n = \inf\{\alpha(a, b, c) \mid a, c \in R_+^n, b \in R_+\}$. Величину α_n назовём гарантированной точностью $(n - 1)$ -мерного приближенного значения n -мерной задачи о рюкзаке или просто гарантированной точностью.

Целью работы является исследование поведения гарантированной точности в зависимости от размерности задачи, частные случаи которого (при $n = 2$ и $n = 3$) были получены [1]. Здесь доказано равенство $\alpha_n = (2^n - 2)/(2^n - 1)$, и построен пример задачи о рюкзаке с точностью приближённого значения, равной $(2^n - 2)/(2^n - 1)$. На основе полученных результатов построен полиномиальный от длины входа приближённый алгоритм решения задачи о рюкзаке с гарантированной точностью не менее $3/4$.

1. Общие свойства точности приближенного значения

Для вектора $a \in R_+^n$ положим $a_0 = \sum_{i=1}^n a_i$.

ЛЕММА 1. *Для любых $a, c \in R_+^n, b \in R_+$ найдутся такие векторы $a', c' \in R_+^n$, что $\alpha(a, b, c) \geq \alpha(a', a'_0, c')$.*

Доказательство Пусть y – оптимальное решение задачи о рюкзаке. Обозначим через a', c' векторы, компоненты которых вычисляются по формулам: $a'_i = a_i y_i, c'_i = c_i y_i$, где $i = 1, 2, \dots, n$. Точка $(1, \dots, 1)$ принадлежит $L(a', a'_0)$ и $c'_0 = cy = \lambda(a, b, c)$, следовательно, $\lambda(a', a'_0, c') \geq \lambda(a, b, c)$. Пусть z точка из $L_0(a', a'_0)$, тогда точка v , компоненты которой находятся по формулам $v_i = z_i y_i$, где $i = 1, \dots, n$, содержится в $L_0(a, b)$. Действительно, $av = a'z \leq a'_0 = ay \leq b, v \in Z_+^n$, и, кроме того, одна из компонент v равна 0, так как $z \in Z_0^n$. Далее, $c'z = cv$, и, значит, в силу произвольности выбора z , справедливо неравенство $\lambda_0(a, b, c) \geq \lambda_0(a', a'_0, c')$. Для доказательства утверждения осталось заметить, что справедливы соотношения $\alpha(a, b, c) = \lambda_0(a, b, c)/\lambda(a, b, c) \geq \lambda_0(a', a'_0, c')/\lambda(a, b, c) \geq \lambda_0(a', a'_0, c')/\lambda(a', a'_0, c') = \alpha(a', a'_0, c')$.

Доказанное утверждение позволяет избавиться в определении гарантированной точности от параметра b .

СЛЕДСТВИЕ 1. *Справедливо равенство $\alpha_n = \inf\{\alpha(a, a_0, c) \mid a, c \in R_+^n\}$.*

Пусть $\alpha(a, a_0, c) < 1$. Тогда все компоненты вектора a строго больше нуля. Действительно, допустим, что у вектора a найдутся нулевые компоненты. Пусть k – наименьший номер нулевой компоненты вектора a . Если $c_k > 0$, то целевая функция не ограничена на $L(a, a_0)$, и, значит, $\alpha(a, a_0, c) = 1$. Если $c_k = 0$, и $p = (p_1, \dots, p_n)$ – оптимальное решение задачи о рюкзаке, то точка p' , которая может отличаться от p только k -ой компонентой, равной нулю, является тоже оптимальным решением задачи о рюкзаке. Так как $p' \in L_0(a, a_0)$, то $\alpha(a, a_0, c) = 1$. Итак, будем считать, что все компоненты вектора a строго больше нуля. Тогда точка $(1, \dots, 1)$ является единственной точкой из множества $L(a, a_0)$, все компоненты которой отличны от нуля. Поэтому, неравенство $\lambda_0(a, a_0, c) \neq \lambda(a, a_0, c)$, реализуется только в случае, если точка $(1, \dots, 1)$ – единственное оптимальное решение задачи о рюкзаке, и, значит, $\lambda(a, a_0, c) = c_0$ и $\alpha(a, a_0, c) = \lambda_0(a, a_0, c)/c_0$. Тем самым доказано следующее следствие.

СЛЕДСТВИЕ 2. *Если $\alpha(a, a_0, c) < 1$, то $\alpha(a, a_0, c) = \lambda_0(a, a_0, c)/c_0$.*

Неравенство $\alpha(a, a_0, c) < 1$ равносильно условию, что система неравенств $ax \leq a_0, cx \geq c_0, x \in Z_+^n$ имеет единственное решение, равное $(1, \dots, 1)$. В частности справедливо

СЛЕДСТВИЕ 3. Если $\alpha(a, a_0, c) < 1$, то множества неотрицательных целочисленных решений систем $ax = a_0$, $cx = c_0$ и $x_j = 1$, где $j = 1, \dots, n$, совпадают.

В этом случае говорят, что система $ax = a_0$, $cx = c_0$ агрегирует систему $x_j = 1$, где $j = 1, \dots, n$.

Рассмотрим выпуклую оболочку точек с целочисленными неотрицательными компонентами, удовлетворяющих уравнению $ax = a_0$. Из следствия 3 вытекает, что точка $(1, \dots, 1)$ является вершиной этой выпуклой оболочки. Следовательно, можно изменить коэффициенты вектора a так, чтобы множество $L(a, a_0)$ осталось неизменным, а на гиперплоскости $ax = a_0$ лежала единственная точка из Z_+^n , равная $(1, \dots, 1)$. В таком случае уравнение $ax = a_0$ называется агрегирующим систему $x_i = 1$, где $i = 1, 2, \dots, n$ (более подробно см. [2]).

2. Пример

Определим векторы a и c , компоненты которых вычисляются по формулам $a_i = 2^{n+i-1} + 2^{n-i}$, $c_i = 2^{i-1}$, где $i = 1, \dots, n$. Покажем, что при сделанном выборе векторов выполняется равенство $\alpha(a, c) = (2^n - 2)/(2^n - 1)$.

Найдём максимальное значение целевой функции cx на множестве $\{x | x \in R_+^n, ax \leq a_0\}$. Известно (см. например [3]), что это значение равно $a_0 \max_{i=1, \dots, n} \{c_i/a_i\}$. Далее, $c_i/a_i = (2^n + 2^{n-2i+1})^{-1} \leq 2^n/(4^n + 2)$ и $a_0 = 4^n - 1$, следовательно, искомое значение равно $2^n(4^n - 1)/(4^n + 2) = 2^n - 3/(4^n + 2) < 2^n$.

Оптимальное значение задачи о рюкзаке не превосходит оптимального значения соответствующей непрерывной задачи, то есть $\lambda(a, c) < 2^n$. Величина $\lambda(a, c)$ целочисленная, и, значит, справедливо неравенство $\lambda(a, c) \leq 2^n - 1$. Точка $(1, \dots, 1)$ принадлежит $L(a)$ и $c_0 = 2^n - 1$, следовательно, установлено равенство $\lambda(a, c) = 2^n - 1$.

Покажем, что $\lambda_0(a, c) = 2^n - 2$. Во-первых, значение целевой функции cx в точке $(1, \dots, 1, 0)$ из $L_0(a)$ равно $2^n - 2$. Во-вторых, точка $(1, \dots, 1)$ единственная в $L(a)$, для которой $cx = 2^n - 1$. Чтобы убедиться в этом, рассмотрим систему

$$cx = c_0 \tag{1}$$

$$ax \leq a_0. \tag{2}$$

Покажем, что эта система не имеет решений из Z_+^n , отличных от $(1, \dots, 1)$. Для этого достаточно убедиться, что система $cx = c_0$, $ax < a_0$ несовместна в целых неотрицательных числах и кроме того, система $ax = a_0$, $cx = c_0$ агрегирует систему $x_i = 1$, где $i = 1, \dots, n$. Доказательство проведём индукцией по размерности n . При $n = 1$ утверждение очевидно. Если $n \geq 2$, то, вычитая из неравенства (2) уравнение (1), умноженное на 2^n , получим неравенство

$$\sum_{i=1}^n 2^{n-i} x_i \leq 2^n - 1, \tag{3}$$

составляющее вместе с уравнением (1) систему, равносильную исходной системе. Из неравенства (3), следует неравенство $x_1 \leq (2^n - 1)/2^{n-1} < 2$, то есть $x_1 \leq 1$. Из уравнения (1) выводим, что в любом целочисленном решении значение переменной

x_1 - нечётное число, то есть $x_1 = 1$. Нетрудно видеть, что после подстановки $x_1 = 1$ система (1), (3) несложными преобразованиями превращается в аналогичную систему с меньшим числом переменных, не имеющую по предположению индукции других решений, кроме $(1, \dots, 1)$.

Тем самым установлены равенства $\lambda_0(a, c) = 2^n - 2$, $\alpha(a, a_0, c) = (2^n - 2)/(2^n - 1)$ и, значит, справедливость следующей леммы.

ЛЕММА 2. *Справедливо неравенство $(2^n - 2)/(2^n - 1) \geq \alpha_n$.*

2. Верхние оценки

Через $[n]$ обозначим множество натуральных чисел, не превосходящих n . Для вектора $c \in R_+^n$ и подмножества $U \subseteq [n]$ положим $c(U) = \sum_{j \in U} c_j$, в частности $c_0 = c([n])$. Определим функцию $\tau(c) = \min_{U, V \subseteq [n]; U \cap V = \emptyset} |c(U) - c(V)|$.

ЛЕММА 3. *Для произвольного $c \in R_+^n$ справедливо неравенство $\tau(c) \leq (2^n - 1)^{-1} c_0$.*

Доказательство. Заметим, что $\tau(c) = \min_{U, V \subseteq [n]; U \neq V} |c(U) - c(V)|$. Действительно, пусть $U \neq V$. Обозначим через U' множество элементов из U , не принадлежащих V , а через V' - множество элементов из V , не принадлежащих U . Множества U' и V' не пересекаются и $|c(U) - c(V)| = |c(U') - c(V')|$.

Отрезок $[0, c_0]$ разбивается величинами $c(U)$ на $2^n - 1$ отрезков (длины отрезков могут быть равны 0). Следовательно, максимальная длина наименьшего из этих отрезков не больше $(2^n - 1)^{-1} c_0$, что и требовалось доказать.

Отметим, если $c = (1, 2, \dots, 2^{n-1})$, то $\tau(c) = c_0/(2^{n-1} - 1)$, и следовательно, неравенство в лемме 3 усилить невозможно.

ТЕОРЕМА. *Для любой n -мерной задачи о рюкзаке точность $(n - 1)$ -мерного приближённого решения не меньше, чем $\alpha_n = (2^n - 2)/(2^n - 1)$. Существует задача о рюкзаке, на которой эта оценка достигается.*

Доказательство. Для доказательства теоремы достаточно показать справедливость неравенства $\alpha_n \geq (2^n - 2)/(2^n - 1)$. Как установлено выше $\alpha_n = \inf \{ \lambda_0(a, a_0, c)/c_0 \mid a, c \in R_+^n \}$.

Разобьём множество натуральных чисел $[n]$ на три взаимно не пересекающихся подмножества J_0, J_1, J_2 . Обозначим через $p(J_0, J_1, J_2)$ точку, компоненты которой определяются по формулам $p_i = s$ при $i \in J_s$, где $s = 0, 1, 2$. Так как точка $(1, \dots, 1)$ является серединой отрезка, соединяющего точки $p(J_0, J_1, J_2)$ и $p(J_2, J_1, J_0)$, то по крайней мере одна из них принадлежит $L(a)$. Пусть для определённости это точка $p(J_0, J_1, J_2)$. Значение целевой функции на этой точке равно $cp(J_0, J_1, J_2) = c_0 - |c(J_0) - c(J_2)|$. Таким образом, при любых непересекающихся подмножествах J_0 и J_2 множества $[n]$ справедливо неравенство $\alpha(a, a_0, c) \geq cp(J_0, J_1, J_2)/c_0 = 1 - |c(J_0) - c(J_2)|/c_0 \geq 1 - \tau(c)/c_0$. Отсюда, по лемме 3, выводим неравенство $\alpha(a, a_0, c) \geq 1 - (2^n - 1)^{-1}$. В силу произвольности выбора векторов $a, c \in R_+^n$, из последнего неравенства выводим $\alpha_n \geq (2^n - 2)/(2^n - 1)$, что и требовалось.

Опишем алгоритм построения приближённого решения задачи о рюкзаке. Для каждой пары индексов i, j решим двумерную задачу о рюкзаке, получающуюся из исходной задачи обнулением всех переменных с номерами отличными от i и j , и из полу-

ченных решений выберем наилучшее. Полученное решение назовём 2-приближённым решением задачи о рюкзаке. Для решения двумерной задачи о рюкзаке можно использовать алгоритмы, описанные в работах [1], [4] или [5]. Трудоёмкость этих алгоритмов (при целочисленном входе) не превосходит $O(\log b)$. Таким образом 2-приближённое решение задачи о рюкзаке данным алгоритмом можно найти с трудоёмкостью не более $O(n^2 \log b)$.

СЛЕДСТВИЕ 4. Точность 2-приближённого решения n -мерной задачи о рюкзаке не меньше $(2^n - 2^{n-2})/(2^n - 1)$.

Доказательство. Поскольку 2-приближённое решение задачи о рюкзаке можно получить последовательным занулением $n - 2$ переменных, а при занулении одной переменной точность решения ухудшается не более чем в α_n раз, то точность 2-приближённого решения не меньше произведения $\alpha_n \alpha_{n-1} \dots \alpha_3$. В указанном произведении числитель каждого сомножителя (кроме последнего) равен удвоенному знаменателю следующего сомножителя. Произведём соответствующие сокращения, получим утверждение следствия.

Отметим, что следствие 4 даёт только нижнюю оценку погрешности 2-приближённого решения задачи о рюкзаке, причем указанная оценка достижима только при $n = 3$ и равна $6/7$. При $n \geq 4$ указанные оценки недостижимы, и могут быть улучшены. Тем не менее, при любом n справедливо неравенство $(2^n - 2^{n-2})/(2^n - 1) > 3/4$, и, значит, точность 2-приближённого решения, при любой размерности задачи n , строго больше $3/4$.

References

- [1] В.Н. Шевченко, А.Ю.Чирков Точность приближённого решения задачи о рюкзаке //Материалы XIV Международной школы-семинара "Синтез и сложность управляющих систем"(Нижний Новгород, 27 октября-1ноября 2003г.)-Н.Новгород Полиграфический участок АНО "МУК НГПУ С.114-117
- [2] Шевченко В.Н. Качественные вопросы целочисленного программирования. М.: Физматлит 1995 192с.
- [3] А.Схрейвер Теория линейного и целочисленного программирования. В 2-х т. Т1:- М.Мир, 1991 342с.
- [4] Веселов С.И. Нахождение выпуклой оболочки целых точек полиэдра на плоскости. Горький: ГГУ. 1988. 6с. Деп. в ВИНТИ 7.12.88 № 8624-В88.
- [5] Чирков А.Ю. Минимизация квазивыпуклых функций на двумерной целочисленной решетке.// Вестник Нижегородского университета им. Н.И. Лобачевского. Математическое моделирование и оптимальное управление. Выпуск 1(26). 2003 с.227-236.

ПРОГРАММЫ И ПРОГРАММНЫЕ КОМПЛЕКСЫ

ЯЗЫК ОПИСАНИЯ МОДЕЛЕЙ СИСТЕМЫ АВТОМАТИЧЕСКОЙ ГЛОБАЛЬНОЙ ОПТИМИЗАЦИИ

А.Р. Ершов

Институт систем энергетики им. Мелентьева СО РАН, Иркутск

Аннотация. В статье представляется детальное описание языка описания моделей и программной реализации решателя задач глобальной одномерной оптимизации.

Ключевые слова: решатель, язык описания моделей, термы, выражения.

Введение

В [1] была исследована возможность создания алгоритма решателя автоматизирующего процесс глобальной оптимизации широкого класса функций. В данной работе представляется детальное описание языка описания моделей и программной реализации решателя, основанного на вышеупомянутом алгоритме.

Исследуемая задача глобальной оптимизации состоит в нахождении глобально минимального значения одномерной явно заданной (факторизуемой) [2] функции $f(x)$ на отрезке α, β :

$$\begin{aligned} f(x) &\longrightarrow \min, \\ x &\in [a, b] \end{aligned} \quad (1)$$

В [2] было показано, как аппроксимировать исходную функцию опорными функциями: вогнутой кусочно-линейными функцией-минорантой (нижняя оценка) и выпуклой кусочно линейной функцией-мажорантой (верхняя оценка).

Иначе говоря, нужно найти опорные функции $\varphi^+(x, y)$ и $\varphi^-(x, y)$ такие, что

$$\varphi^-(x, y) \leq f(x) \leq \varphi^+(x, y) \quad \forall x \in [\alpha, \beta], \quad (1.1)$$

$$\varphi^-(y, y) = f(y) = \varphi^+(y, y), \quad (1.2)$$

имеющие вид

$$\varphi^+(x, y) = \max \{ k_1^+(y)x + b_1^+(y), \quad k_2^+(y)x + b_2^+(y) \}, \quad (1.3)$$

$$\varphi^-(x, y) = \min \{ k_1^-(y)x + b_1^-(y), \quad k_2^-(y)x + b_2^-(y) \}, \quad (1.4)$$

где $k_1^+(y), b_1^+(y), k_2^+(y), b_2^+(y), k_1^-(y), b_1^-(y), k_2^-(y), b_2^-(y)$ - некоторые числа.

Таким образом, задача нахождения глобального оптимума разбивается на следующие этапы:

```
set A;
var x;
minimize f: -sum {k in A} k*sin((k+1)*x + k);
subject to bounds: -10 <= x <= 10;
A := {1, 2, 3, 4, 5};
```

Рис. 1: Пример задания модели.

1. Формулирование модели, т.е. абстрактной системы переменных, целевой функции и функций - ограничений, представляющих общий вид рассматриваемой проблемы.
2. Разбор модели термы (переменные, константы), выражения (операции над термами или другими выражениями) и элементы модели (декларации переменных, целевой функции, ограничений и т.п.).
3. Интерпретация модели - генерация опорных функций для каждого терма и каждого выражения.
4. Уточнение опорных функций и нахождение приближенного значения глобального оптимума.

1. Формулирование модели.

Для описания модели создан язык, являющийся подмножеством языка математического программирования AMPL [3]. Пример задания модели приведен на рис. 1. Описание модели состоит из

1. объявления множеств и переменных,
2. объявления целевой функции,
3. объявления ограничений.

Множество объявляется ключевым словом **set**, именем множества и опциональным атрибутом.

объявление множества :

set *имя атрибут* ;

где единственно возможный атрибут описывает размерность множества

атрибут :

dimen *n*, $n \in \mathbb{N}$.

Множество должно быть проинициализировано. В текущей версии языка допускается только явная инициализация множества списком чисел, разделенных запятыми и заключенным в фигурные скобки, например

A := {1, 2, 3, 4, 5};

Объявление переменной состоит из ключевого слова **var**, имени переменной, индексного выражения и опциональных атрибутов.

объявление переменной :
`var имя атрибут ;`

где атрибут может быть одним из
атрибут :
`binary`
`integer`

Атрибут `integer` означает целочисленный тип переменной, `binary`, соответственно, Булев. Переменные могут быть объявлены как элементы множеств, с помощью индексных выражений.

Индексное выражение состоит из списка индексных переменных.
индексное выражение :
`{список переменных} ;`

список переменных :
индексная-переменная `in` *имя-множества* ;

Объявление целевой функции доступно в следующих вариантах:

объявление целевой функции :
`maximize имя :выражение ;`
`minimize имя :выражение ;`

Выражения могут быть арифметическими и логическими. Выражения скомбинированы из операций, переменных и констант. Если выражение не содержит переменных, то оно называется *константным* выражением.

Арифметические выражения могут состоять из арифметических операторов, встроенных функций и операторов редукции, таких как оператор суммы \sum

арифм-выражение :
число
переменная
выражение арифм-опер *выражение*
унарный-опер *выражение*
встроенная-функция (*список-выражений*)
редукции-опер *индекс* *выражение*
(*выражение* (

арифм-опер : + - * / mod div ^ **

унарный-опер : + -

редукции-опер : sum prod max min

Операторы редукции максимально приближены к традиционной математической нотации. Например, оператор суммы в формуле

$$\sum_i h_i * w_i, i \in A$$

на языке модели выглядит как:

```
sum {i in A} h[i] * w[i];
```

Логические выражения применяются с Булевыми типами данных.

лог-выражение :

арифм-выражение

лог-выражение лог-опер лог-выражение

not лог-выражение

(лог-выражение)

лог-опер : < <= = == != <> > >=

Объявление ограничений осуществляется следующим образом:

объявление ограничения :

subject to *имя : ограничение ;*

ограничение :

переменная < выражение

переменная > выражение

переменная = выражение

переменная >= выражение

переменная <= выражение

выражение <= переменная <= выражение

выражение >= переменная >= выражение

Кроме переменных и констант в выражениях могут быть использованы встроенные функции. В настоящее время разрешено использовать `abs(x)`, `sin(x)`, `cos(x)`, `max(x)` и `min(x)`.

2. Разбор модели.

Модель сформулированная на изложенном выше языке пропускается через лексический, синтаксический и семантический анализаторы.

Лексический анализатор разбивает модель на лексемы - неделимые с точки зрения модели термы, такие как ключевые слова, числа, идентификаторы.

Синтаксический анализатор проверяет модель на соответствие описанной выше грамматике. Результатом работы синтаксического анализатора является абстрактное

синтаксическое дерево (АСД) - направленный ациклический граф, в котором терминальные вершины соответствуют термам, нетерминальные - формулам, выражениям и т.д.

Семантический анализатор обходит АСД и находит переменные, контролируя правильность их применения (любая переменная должна быть продекларирована до первого использования). Также анализируются ограничения. Избыточные ограничения сокращаются, отсутствующие замещаются граничными для данной вычислительной архитектуры.

3. Интерпретация модели.

На следующем этапе АСД обходится интерпретатором. Интерпретатор параметризован типом данных. Возможны два возможных варианта параметра - численный, такой как `double`, `long double` и т.д., и кусочно-линейный (о нем ниже). Интерпретатор с численным параметром вычисляет значения целевой функции с заданными границами и дискретностью. Результат вычисления отображается в виде графика, удобного для отладки алгоритма (в настоящее время поддерживаются только функции одной переменной). На этом же графике позднее отображаются вычисленные кусочно-линейные опорные функции.

4. Уточнение кусочно-линейных опорных функций.

Алгоритм уточнения опорных функций в целом описан в [2]. Опорные функции $\varphi^+(x, y)$ и $\varphi^-(x, y)$ строятся в результате обхода АСД интерпретатором, параметризованным кусочно-линейным типом данных. В процессе обхода дерева выражения $\varphi^+(x, y)$ и $\varphi^-(x, y)$ вычисляются для каждой вершины АСД. Вычисленные коэффициенты опорных функций затем сохраняются в структуре кусочно-линейного типа (КЛТ), хранящей коэффициенты двух кусочно-линейных опорных функций; каждая функция состоит из двух линейных сегментов, мажоранты $k_1^+(y)x + b_1^+(y)$, $k_2^+(y)x + b_2^+(y)$ и миноранты $k_1^-(y)x + b_1^-(y)$, $k_2^-(y)x + b_2^-(y)$.

Для уменьшения вычислительной сложности алгоритм оптимизирован следующим образом.

1. В процессе вычисления коэффициентов опорных функций вычисляется и сохраняется вспомогательная информация о значениях и характере функции, например значения целевой функции $f(x)$ и ее производных $f'(x)$ и $f''(x)$ (если они существуют) в трех точках (на левой границе, на правой и в точке пересечения сегментов опорной функции), максимальное и минимальное значения на отрезке, факт выпуклости функции и т.д. Использование вспомогательной информации существенно снижает вычислительную сложность алгоритма построения опорных функций, позволяя выбрать его оптимальный вариант.
2. Лексический анализ модели, синтаксический разбор и построение АСД осуществляются лишь на первой итерации. На последующих итерациях производится обход уже построенного АСД, опуская лексическую, синтаксическую и семанти-

ческую проверки, что существенно снижает вспомогательные вычислительные затраты.

3. Нетрудно заметить, что в сложном выражении термы и формулы могут повторяться, что приводит к избыточным вычислениям. Для предотвращения избыточности каждая опорная функция сохраняется в индексированной таблице, каждый элемент которой состоит из ключа (символического выражения формулы или терма) и значения - КЛТ-структуры. Таблица хешируется по ключу, т.е. со значением ключа производятся некоторые арифметические действия, позволяющие вычислить функцию, указывающую на адрес в таблице, где хранится соответствующая КЛТ-структура[4] Эффект оптимизации особенно заметен на примерах формул с операторами редукции.

5. Программная реализация.

На основе указанных выше и в [2] принципов был создан прототип решателя. Прототип написан на языке C++[5], с применением стандартной библиотеки шаблонов STL[6]. В процессе описания грамматики языка и генерации интерпретатора использовались генераторы синтаксических анализаторов YACC[7], и лексических анализаторов Lex[7]. Результаты проведенных численных экспериментов приведены в [2].

References

- [1] Хамисов О.В. *Глобальная оптимизация функций с вогнутой минорантой*. Журн.вычисл.математики и матем.физики. 2004. Т.44, N9,с.1552-1563
- [2] Ершов А.Р. Хамисов О.В. *Автоматическая глобальная оптимизация*. Дискретный анализ и исследование операций. Серия 2, Т.11, N2, с.45-68
- [3] Fourer R., Gay D.M., Kernigan B.W. *AMPL A Modeling Language for Mathematical Programming*. Second Edition //Brooks/Cole-Thomson Learning., 2003.
- [4] Knuth D.E. *The Art of Computer Programming*. Second Edition, Vol. 3.//Addison-Wesley, 2000.
- [5] British Standards Institute. *The C++ Standard*. Second Edition // John Wiley & Sons Ltd, 2003.
- [6] P.J. Plauger,P.J., Lee M., Musser D., and Stepanov A.A. *C++ Standard Template Library* //Prentice Hall PTR, December 2000
- [7] Levine, D. Mason, T., Brown, D. *Lex & YACC*. Second Edition //O' Reilly, 2003.

ЭКСПЕРИМЕНТАЛЬНАЯ ОЦЕНКА ЭФФЕКТИВНОСТИ АЛГОРИТМОВ И ПРОГРАММ

Л.И. Крутова, С.В. Ленский

*Уральский государственный университет им. А.М.Горького, г.Екатеринбург
e-mail: ludmila.krutova@usu.ru*

Аннотация. В статье рассматривается экспериментальный альтернативный теоретическому подход решения оценки сложности алгоритмов и программ. При этом вычисление точной оценки $\Theta(g(n))$ может быть достигнуто с гораздо меньшими затратами и часто позволяет преодолеть возникающие теоретические затруднения.

Ключевые слова: комбинаторные алгоритмы, анализ алгоритмов, оптимизация алгоритмов и программ.

Введение

В настоящее время получают интенсивное развитие в математике экспериментальные направления. Одним из таких направлений является автоматизация определения вычислительной сложности алгоритмов и программ, для которых, в первую очередь, это трудно или невозможно сделать аналитически. Не исключая теоретических исследований, предлагаемый подход заменяет и дополняет там, где это возможно аналитические методы. Аналогичный процесс с опережением в два - три десятилетия проходил и происходит в настоящее время в области аналитических исследований систем дифференциальных уравнений.

Анализ алгоритмов проводится в основном теоретически [1,2,3], как правило, точные оценки $f(n) = \Theta(g(n))$ выводятся через получение оценок $f(n) = O(g(n))$ и $f(n) = \Theta(g(n))$. При этом, $f(n) = \Theta(g(n))$ считается точной при совпадении $f(n) = O(g(n))$ и $f(n) = \Theta(g(n))$, или при достаточно малом интервале между ними. Для некоторых алгоритмов на это были затрачены годы работы известных математиков. Экспериментальные тестирования эффективности алгоритмов часто проводятся с научными [1] или практическими целями [4,5], но, как правило, эти тестирования не проводятся до получения точной оценки и недостаточно стандартизованы. В данном исследовании показано вычисление точной оценки на заданном интервале. Данный интервал определяется постановкой решаемой алгоритмом или программой задачи и возможностями тестирующего компьютера. Намечены пути получения точной оценки в традиционном смысле и способы преодоления этих обстоятельств.

Постановка задачи.

Необходимо сопоставить время работы алгоритма или программы, вычисленное в компьютерном эксперименте с объективными показателями, например, размером данных, на входе в алгоритм или программу.

Метод решения.

1. Предварительная стандартизация перед проведением экспериментов для научных целей должна проводиться на компьютере с минимально необходимым набором программ и гарантией отсутствия вирусов. Для практических целей достаточно

выключить или деинсталлировать процессы, использующие ресурсы компьютера во время эксперимента.

2. Экспериментальный поиск максимально возможной размерности вашей задачи, выполнимой на вашем компьютере по времени (не более нескольких часов) или по емкости оперативной памяти до переполнения (то есть, не допуская обмена с внешним носителем).

3. Поиск минимально возможной для вычислений размерности, которая определяется надежностью измерений при $p < 0.01$, (то есть, если измеряются секунды, минимальное время работы программы должно быть не менее 100 секунд, если сотые доли секунды, то минимальное время должно быть не менее 1 секунды).

4. Проведение замеров в интервале времени работы программы и в выбранном интервале размерности.

5. Построение графиков зависимости: размерность - время. По оси абсцисс - размерность. По оси ординат - время.

6. Изучение характера и поведения полученных зависимостей.

7. По характеру поведения полученных зависимостей определение $\Theta(g(n))$ (экспоненциальная, полиномиальная, линейная, постоянная).

8. Если имеются аналитические оценки алгоритма, сопоставление их с полученными экспериментально оценками $\Theta(g(n))$.

Обсуждение результатов.

Экспериментальный подход выгоден для правильного выбора алгоритма или программы для задач, в которых вычислительная сложность имеет большое значение. В том числе на этапе предварительной отладки для таких задач, особенно если планируется использовать результаты решения этих задач в больших системах проектирования. При этом проводится гораздо меньше традиционной аналитической работы.

Недостатки данного подхода связаны с конкретными возможностями тестирующего компьютера, а также с интервальным ограничением оценки. Поэтому должна проводиться предварительная проверка на разных платформах, конфигурациях и для различных мощностей тестирующих данный алгоритм компьютеров. Необходимо проводить аналитическую работу по установлению прогностической асимптотической устойчивости за верхней гранью интервала.

Дополнительные теоретические разработки дали бы аналитическую поддержку определения асимптотической устойчивости по верхней грани для исследуемых экспериментально классов задач.

СПИСОК ЛИТЕРАТУРЫ

[1] Кормен Т., Лейзерсон Ч., Ривест Р. Алгоритмы: Построение и анализ. М.: МЦНМО, 1999, с. 29.

[2] Кнут Д. Искусство программирования для ЭВМ. Т.1-3. М.: Издат.дом "Вильямс", 2000.

[3] Гасфилд Дэн. Строки, деревья и последовательности в алгоритмах: Информатика и вычислительная биология. СПб.: Невский Диалект; БХВ-Петербург, 2003.

[4] Яковлев Н. Н., Асанов М. О., Баранский В. А., Беспмятных С. Н., Крутова Л. И., Петров А. Н., Харин Б. Н., Шамгунов Н. К. Развитие канального подхода к

конструированию МЭА, Свердловск, Издательство Уральского университета, 1987.

[5] Асанов М. О., Крутова Л.И. Компьютерная технология обучения алгоритмам дискретной оптимизации. Сборник "Новые информационные технологии в университетском образовании". Новосибирск, 1994.

БУЛЕВА МОДЕЛЬ ОПТИМАЛЬНОГО ПЛАНИРОВАНИЯ АСИНХРОННЫХ АБСТРАКТНЫХ ПРОГРАММ¹

Г.А. Опарин, А.П. Новопашин

Институт динамики систем и теории управления СО РАН, Иркутск

e-mail: oparin@icc.ru

Аннотация. Предложена новая формулировка задачи построения параллельных асинхронных абстрактных программ требуемой длины в параллельных вычислительных системах. Условия задачи планирования представлены в виде системы булевых уравнений (ограничений), решения которой определяют возможные планы активизации программных модулей, спецификации которых размещены в базе знаний планировщика. Учитываются ограничения на число процессоров и временные задержки, возникающие при исполнении программных модулей.

Ключевые слова: параллельные вычислительные системы, планирование, булевы уравнения.

Введение

Непроцедурные языки и системы параллельного модульного программирования [1,2] позволяют накапливать в памяти ЭВМ знания о вычислительных модулях предметной области и использовать эти знания при автоматическом решении задач заданного класса, однако эффективное использование этой технологии требует решения трудной проблемы планирования параллельной абстрактной программы (построения плана решения задачи).

Описания взаимосвязанных по информации программных модулей размещены в базе знаний планировщика. Необходимо определить, какие модули, в какой последовательности и на каких процессорах параллельной вычислительной системы (из числа доступных) должны быть исполнены для вычисления требуемого набора целевых параметров B_0 по заданному множеству параметров - исходных данных A_0 . Цель – получить параллельный план решения задачи $T = (A_0, B_0)$, который является:

- 1) допустимым (модули должны быть упорядочены таким образом, чтобы каждый из них к моменту своего запуска был обеспечен необходимыми входными данными, или, иначе говоря, для любого входного параметра модуля плана должен быть хотя бы один встречающийся ранее модуль с одноименным выходным параметром);
- 2) неповторным (каждый модуль может входить в план не более одного раза);
- 3) неизбыточным (исключение любого модуля из плана приводит к недопустимому плану);
- 4) оптимальным (длина плана должна быть равна заданной величине k).

¹Работа выполнена при поддержке РФФИ (грант № 04-07-90358).

При этом мы предполагаем, что при построении параллельного плана с перечисленными свойствами число доступных процессоров параллельной вычислительной системы ограничено некоторой величиной, имеются временные задержки, связанные с исполнением программных модулей, время передачи данных между модулями не учитывается. В такой постановке задача планирования является NP-трудной, её условия могут быть представлены в виде системы булевых уравнений (ограничений), а собственно план исполнения модулей является решением этой системы. Преимущество булева моделирования (по сравнению с традиционным дедуктивным подходом) состоит в том, что оно позволяет, во-первых, получать параллельные планы требуемой длины, во-вторых, учитывать разнообразные ограничения на план и, наконец, в-третьих, использовать существующие эффективные решатели булевых уравнений (или SAT-решатели [3]), которые в ряде случаев обгоняют по быстродействию специализированные алгоритмы планирования. В целом, как отмечается в [4], такой подход не только обеспечивает более гибкую среду для установления различных видов ограничений на параллельные планы, но также более точно отражает теорию современных систем планирования, основанных на ограничениях.

1. База знаний планировщика

В качестве базы знаний планировщика используется вычислительная модель $KB = (F, Z, In, Out)$, где $F = \{F_1, \dots, F_n\}$ - множество модулей, работающих над полем общих транзитных данных $Z = \{Z_1, \dots, Z_m\}$, являющихся входными или выходными параметрами этих модулей; $In \subset F \times Z$, $Out \subset F \times Z$ - отношения, отражающие взаимосвязь модулей с данными соответственно по входу и выходу. Таким образом, с каждым модулем F_i связано два множества параметров $A_i, B_i \subset Z$, называемых соответственно входом и выходом. Вход A_i определяет данные, которые необходимо иметь, чтобы получить результаты, представленные выходом B_i . В дальнейшем для этого будем использовать запись $F_i(A_i; B_i)$. Без потери общности будем считать, что модули F_1 и F_2 из множества F моделируют условия постановки задачи планирования $T = (A_0, B_0)$, а именно: база знаний KB содержит модули $F_1(; A_0)$ и $F_2(B_0;)$, где $A_0, B_0 \subset Z$. Отсутствие атрибута до или после точки с запятой означает пустоту соответствующего множества. Модуль F_1 назовем модулем начальных данных, а модуль F_2 - целевым модулем.

Предполагается, что, что база знаний KB является избыточной в том смысле, что для решения задачи используется только часть модулей из F и/или задача T имеет несколько альтернативных планов решения.

Отношения In и Out удобно задать в виде двух булевых матриц A и B размерности $n \times m$, элементы которых сформированы следующим образом: $a_{ij} = 1$ ($b_{ij} = 1$), если параметр Z_j является входным (выходным) для модуля F_i . Далее через A_i, B_i ($i = 1, \dots, n$) будем обозначать строки этих матриц, а через A'_i, B'_i ($i = 1, \dots, m$) - столбцы. Строки и столбцы матриц A и B являются двоичным представлением подмножеств параметров и модулей соответственно; запись $q \in S$ (S - это двоичная строка A_i, B_i, A'_i или B'_i) означает, что q принимает значения номеров единичных элементов в двоичной строке S .

2. Булево моделирование

Итак, мы рассматриваем ситуацию, когда все процессоры одинаковы и количество их ограничено величиной pr . Каждый из модулей может выполняться на любом из процессоров. Время τ_j исполнения модуля $F_j \in F$ является дискретной величиной: $\tau_j \in N = \{1, 2, \dots\}$. Время передачи данных от процессора к процессору не учитывается, в каждый момент времени процессор может исполнять только один модуль. Кроме того, задан общий директивный срок решения задачи $k \in N$. Необходимо построить асинхронный план, длина которого равна величине директивного срока k .

Определим план в виде матрицы X $k \times n$ булевых переменных x_{ij} , где $x_{ij} = 1$ означает, что модуль F_j стоит на i -ом месте в плане X . Общая длина плана равна k , его строка задает множество параллельно исполняемых модулей, а столбцы соответствуют множеству модулей F . Тогда булевы ограничения на элементы матрицы X имеют следующий вид:

Условие 1. Условие постановки задачи планирования $T = (A_0, B_0)$ (модуль начальных данных $F_1 = (; A_0)$ и целевой модуль $F_2 = (B_0;)$ расположим соответственно в первой и последней строках плана; другие модули в этих строках отсутствуют):

$$\bar{x}_{11} = 0, \quad \bigvee_{j=2}^n x_{1j} = 0, \quad \bar{x}_{k2} = 0, \quad x_{k1} = 0, \quad \bigvee_{j=3}^n x_{kj} = 0.$$

Условие 2. Условие непрерывности плана (в каждой строке плана находится хотя бы один активизированный модуль):

$$\bigvee_{t=2}^{k-1} \bigwedge_{j=1}^n \bigwedge_{i=t-\tau_j+1}^t \bar{x}_{ij} = 0.$$

Условие 3. Условие неповторности плана:

$$\bigvee_{j=1}^n \bigvee_{i=1}^{k-1} \bigvee_{p=i+1}^k (x_{ij} \wedge x_{pj}) = 0.$$

Условие 4. Условие допустимости плана:

$$\bigvee_{t=2}^k \bigvee_{p=1}^n (x_{tp} \wedge y) = 0,$$

где

$$y = \begin{cases} \bigvee_{q \in A_p} \bigwedge_{j \in B'_q} \bigwedge_{i=1}^{t-\tau_j} \bar{x}_{ij}, & \text{если } (A_p \neq 0) \wedge ((\forall q \in A_p)(B'_q \neq 0)); \\ 1, & \text{если } (A_p \neq 0) \wedge ((\exists q \in A_p)(B'_q = 0)); \\ 0, & \text{если } A_p = 0. \end{cases}$$

Условие 5. Условие избыточности плана:

$$\bigvee_{s=2}^{k-1} \bigvee_{r=1}^n (x_{sr} \wedge \bar{u}) = 0,$$

где

$$u = \bigvee_{t=2}^k \bigvee_{p=1}^n (x_{tp} \wedge y),$$

индексы t и p удовлетворяют условию $\overline{(t = s) \wedge (p = r)}$, а y определяется выражением в условии 4.

Условие 6. Условие ограниченности числа процессоров, предоставленных для решения задачи T (в каждой строке плана должно быть не более pr единиц, и существует хотя бы одна строка, в которой содержится ровно pr единиц):

$$\bigvee_{t=2}^{k-1} \left(\bigvee_{1 \leq j_1 \leq j_2 \leq \dots \leq j_{pr+1} \leq n} (x_{t,j_1} \wedge x_{t,j_2} \wedge \dots \wedge x_{t,j_{pr+1}}) \right) \vee \bigvee_{t=2}^{k-1} \left(\bigwedge_{1 \leq j_1 \leq j_2 \leq \dots \leq j_{n-pr+1} \leq n} (\bar{x}_{t,j_1} \wedge \bar{x}_{t,j_2} \wedge \dots \wedge \bar{x}_{t,j_{n-pr+1}}) \right) = 0.$$

Для вычисления решений системы уравнений, состоящей из условий 1-6, использовался решатель булевых уравнений REBUS [5]. Метод решения рассчитан на представление булевых функций левой части уравнений в общем виде (таковым является условие 5; условия 1-4 и 6 записаны в дизъюнктивной нормальной форме) и реализует технику хронологического возврата с использованием трёхзначной логики Клини.

В качестве примера рассмотрим базу знаний решателя:

$$KB = \{F_1(; z_1), F_2(z_4, z_5;), F_3(z_1; z_2), F_4(z_1; z_3), F_5(z_1; z_5), F_6(z_2, z_3; z_4)\}.$$

Решение булевых уравнений, соответствующих условиям 1-6, при $\tau_j = 1$ (для всех j) и $pr = 2$ средствами системы REBUS даёт 10 планов в строгой параллельной форме [6]: 1 план длиной 4 (№1) и 9 планов длиной 5 (№ 3-10):

$$\text{№1: } F_1, F_3 - F_4, F_5 - F_6, F_2;$$

$$\text{№2: } F_1, F_3 - F_4, F_5, F_6, F_2;$$

$$\text{№3: } F_1, F_3 - F_4, F_6, F_5, F_2;$$

$$\text{№4: } F_1, F_3 - F_5, F_4, F_6, F_2;$$

$$\text{№5: } F_1, F_4 - F_5, F_3, F_6, F_2;$$

$$\text{№6: } F_1, F_3, F_4 - F_5, F_6, F_2;$$

$$\text{№7: } F_1, F_4, F_3 - F_5, F_6, F_2;$$

$$\text{№8: } F_1, F_5, F_3 - F_4, F_6, F_2;$$

$$\text{№9: } F_1, F_3, F_4, F_5 - F_6, F_2;$$

$$\text{№10: } F_1, F_4, F_3, F_5 - F_6, F_2.$$

Символом “-” объединены параллельно исполняемые модули.

Пусть задержки при исполнении модулей F_3 и F_5 равны: $\tau_3 = 3$ и $\tau_5 = 4$. Остальные задержки сохраняются равными единице. С учётом введённых задержек будет найдено 2/10/12/13/7 асинхронных планов длиной соответственно 6/7/8/9/10.

Из множества найденных планов рассмотрим три:

А. $F_1, F_3 - F_4, F_3^+, F_3^+, F_5 - F_6, F_5^+, F_5^+, F_5^+, F_2$;

В. $F_1, F_3 - F_4, F_3^+ - F_5, F_3^+ - F_5^+, F_6 - F_5^+, F_5^+, F_2$;

С. $F_1, F_3 - F_4, F_3^+ - F_5, F_3^+ - F_5^+, F_5^+, F_6 - F_5^+, F_2$.

Символом “+” обозначены продолжения исполнения активизированных ранее модулей.

Анализ планов А, В, С показывает, что план А соответствует (при $\tau_3 = \tau_5 = 1$) плану №1 вышеприведенного списка из 10 планов, а планам В и С при том же условии соответствует план №2. Длина плана №1 в предыдущем случае равна 4, плана №2 – 5. При планировании с учётом временных задержек ситуация меняется – план А реализуется за 9 шагов, а планы В и С – за 7. Очевидно, что выбор стоит делать в пользу одного из двух последних, несмотря на то, что план А характеризуется более высокой, на первый взгляд, степенью параллелизма (две пары параллельно исполняемых модулей). Планы В и С имеют меньшую длину и более эффективно осуществляют загрузку процессоров параллельной вычислительной системы.

Количество неизвестных булевых переменных (размерность задачи) в наихудшем случае равно n^2 . В случае большой избыточности множества модулей F для сокращения размерности задачи можно рекомендовать предварительную обработку базы знаний решателя традиционно применяемыми при планировании алгоритмами прямой и обратной волны.

Заключение

Приведена новая формулировка задачи построения параллельных асинхронных абстрактных программ в параллельных вычислительных системах. При этом мы предполагали, что существуют временные задержки при исполнении модулей, ограничение на число доступных процессоров, а время передачи данных между модулями не учитывается. Эта же задача, но без учёта временных задержек в рамках концепции неограниченного параллелизма рассмотрена в [7].

Булева модель для построения последовательных планов заданной длины в системах-посредниках, обеспечивающих структурные запросы данных в сети Internet, рассмотрена в [8]. Сравнительный анализ показывает, что требование параллельности плана существенно усложняют модель, не позволяя применять известные SAT-решатели, ориентированные на представление булевых ограничений в нормальной форме.

References

- [1] В.А. Вальковский, В.Э. Малышкин. *Синтез параллельных программ и систем на вычислительных моделях*. - Новосибирск: Наука, 1988. - 129 с.
- [2] Г.А. Опарин, А.Г. Феоктистов. *Инструментальная распределенная вычислительная САТУРН-среда*. // Программные продукты и системы. - 2002, №2. - С. 27-30.
- [3] L. Simon. *The SAT-Ex Site: "The experimentation web site around the satisfiability problem"*. [<http://www.lri.fr/~simon/satex/satex.php3> 06.02.2003].
- [4] H. Kautz, V. Selman. *Planning as Satisfiability*. In Proceedings of the 10th European Conference on Artificial Intelligence (ECAI), 1992, pp. 359-363.
- [5] Г.А. Опарин, В.Г. Богданова. *Алгоритмы решения больших разреженных систем булевых уравнений*. // Методы оптимизации и их приложения: Труды 12 Байкальской межд. конф. (Иркутск, июнь 2001 г.). Секция 5. Дискретная математика. - Иркутск: ИГУ, 2001. - С. 114-118.
- [6] В.В. Воеводин, Вл.В. Воеводин. *Параллельные вычисления*. - СПб.: БХВ - Петербург, 2002. - 608 с.
- [7] Г.А. Опарин, А.П. Новопашин. *Булево моделирование планирования действий в распределенных вычислительных системах*. // Известия РАН. Теория и системы управления. - 2004. - №5. - С. 105-108.
- [8] S. Prestwich, S. Bressan. *A SAT Approach to Query Optimization in Mediator Systems*. In Proceedings of the Fifth International Symposium on the Theory and Applications of Satisfiability Testing, University of Cincinnati, 2002, pp. 252-259.

BOOLEAN MODEL OF OPTIMAL PLANNING OF ASYNCHRONOUS ABSTRACT PROGRAMS

G.A.Oparin, A.P.Novopashin

*Institute of Systems Dynamics and Control Theory SB RAS, Irkutsk, Russia
e-mail: oparin@icc.ru*

Abstract. A new formulation is proposed for the problem of constructing parallel asynchronous abstract programs of a required length in parallel computing systems. The conditions of a planning problem are represented as a system of Boolean equations (constraints), whose solutions determine the possible plans for activating the program modules. Specifications of modules are stored in the knowledge base of the planner. Constraint on number of processors and time delays at execution of modules are taken into account.

Keywords: parallel computing systems, planning, Boolean equations.

