Interval Computations
No 1, 1993

OPTIMAL SOLUTION OF INTERVAL LINEAR SYSTEMS IS INTRACTABLE (NP-HARD)

Vladik Kreinovich, Anatoly V. Lakeyev and Sergey I. Noskov

All known methods for finding optimal solutions to interval linear systems demand (in the worst case) exponential time. In this paper, we show that this problem is NP-hard, and thus (unless NP=P) faster algorithms are impossible.

ОПТИМАЛЬНОЕ РЕШЕНИЕ СИСТЕМ ИНТЕРВАЛЬНЫХ ЛИНЕЙНЫХ УРАВНЕНИЙ — NP-ТРУДНАЯ ЗАДАЧА

В.Крейнович, А.В.Лакеев, С.И.Носков

Все известные методы поиска оптимальных решений интервальных линейных систем требуют (в худшем случае) экспоненциального времени. В работе показано, что эта задача NP–трудная, и, таким образом (если только NP≠P), более быстрый алгоритм невозможен.

1. Introduction

In many real-life problems, it is necessary to solve linear systems. In many real-life problems, the desired values $x_1, ..., x_n$ must be determined from their known linear combinations $A_{i1}x_1 + ... + A_{in}x_n$. In other words, we must solve the system of linear equations $A_{i1}x_1 + ... +$

$A_{in}x_n = b_i, 1 \le i \le N$ with known b_i and A_{ij}.

Interval systems. In the ideal case, when we know b_i and A_{ij} precisely, it is sufficient to know $N = n$ (independent) linear combinations. In many real-life problems, however, we know only the intervals for the values b_i and A_{ij}. In this case, we will not be able to find precise values of all x_i, only intervals of their possible values. In this case, additional linear combinations may increase the precision (i.e., diminish the interval). In view of that, in some real-life cases, N is taken to be greater than n.

Let's give precise definitions (see, e.g., [12]):

Definition 1. Assume that $p \ge 1$ is an integer. By an *interval $p-$vector* $\mathbf{b}$, we mean a sequence of p intervals $\mathbf{b}_1, \mathbf{b}_2, ..., \mathbf{b}_p$. We say that a $p-$vector $b = (b_1, ..., b_p)$ *belongs* to $\mathbf{b}$ ($b \in \mathbf{b}$) if $b_i \in \mathbf{b}_i$ for all i. Similarly, for any integers $p \ge 1$ and $q \ge 1$, by an *interval $p \times q-$matrix* $\mathbf{A}$, we mean a $p \times q$ matrix whose elements are intervals $\mathbf{A}_{ij}$, $1 \le i \le p, 1 \le j \le q$. We say that a $p \times q$ matrix A with components A_{ij} *belongs* to $\mathbf{A}$ ($A \in \mathbf{A}$) if $A_{ij} \in \mathbf{A}_{ij}$ for all i and j.

Definition 2. Assume that integers $n > 0$ and $N \ge n$ are given. By an *interval linear system* we mean a pair $(\mathbf{A}, \mathbf{b})$, where $\mathbf{b}$ is an interval $N-$vector, and $\mathbf{A}$ is an interval $N \times n-$matrix. This pair is also denoted as $\mathbf{A}x = \mathbf{b}$. We say that an $n-$vector $x = (x_1, ..., x_n)$ is a *possible solution* of a system $\mathbf{A}x = \mathbf{b}$ if $Ax = b$ for some matrix $A \in \mathbf{A}$ and some vector $b \in \mathbf{b}$. The set of all possible solutions of an interval linear system will be denoted by $\Sigma_{\exists\exists}(\mathbf{A}, \mathbf{b})$. In other words,

$$\Sigma_{\exists\exists}(\mathbf{A}, \mathbf{b}) = \{x \in R^n \ |(\exists A \in \mathbf{A})(\exists b \in \mathbf{b})(Ax = b)\}.$$

Comment. This denotation was introduced by S. P. Shary (private communication) to distinguish this notion from other notions of a solutions set (see, e.g., [4]).

Definition 3. We say that an interval linear system is *consistent* if it has a possible solution, and that it is *non-singular* if its set of possible solutions is bounded.

Comment. For example, a system is non-singular if $N = n$, and all matrices $A \in \mathbf{A}$ are non-singular [12]. Another case is when we have a non-singular system, and add additional equations to it.

Definition 4. An *optimal* (*exact*) solution of an interval linear system $\mathbf{A}x = \mathbf{b}$ is a set of n intervals $[x_j^-, x_j^+]$, where $1 \le j \le n$,

$$x_j^- = \min\{x_j |\ x \in \Sigma_{\exists\exists}(\mathbf{A},\mathbf{b})\} \text{ and } x_j^+ = \max\{x_j |\ x \in \Sigma_{\exists\exists}(\mathbf{A},\mathbf{b})\}.$$

There exist several algorithms that find an optimal solution to a consistent non-singular interval linear system (see [8], [1], [7], [10], [6], [11], [12], and references therein). These algorithms handle the case of the square matrix, when $N = n$. The main problem with these algorithms is as follows: If we know A and b precisely, then one can compute the components $x_1, ..., x_n$ in *polynomial* time, namely, in time that grows as $\le Cn^3$. Even for large n, this is feasible. However, for all known interval algorithms, the running time increases exponentially with n (i.e., as a^n) even for $N = n$, and is, therefore, infeasible for large n.

In this paper, we prove that the problem of finding an optimal solution to a consistent non-singular interval linear system is in the general case intractable (or, using the precise mathematical notion from complexity theory [3], NP-hard).

Therefore, we cannot expect polynomial-time algorithms for interval linear systems (unless, of course, someone finds a way to solve all intractable problems).

2. Main result

Problem. *Given a consistent non-singular interval linear system, find its optimal solution.*

What is NP-hard: a brief informal explanation. We want to prove that this problem is NP-hard. This notion (see, e.g., [3]) means that if there exists an algorithm solving interval systems in polynomial time (i.e., whose running time does not exceed some polynomial of the input length), then the polynomial-time algorithm would exist for practically all discrete problems such as propositional satisfiability problem, discrete optimization problems, etc, – and it is a common belief that for at least some of these discrete problems no polynomial-time algorithm is possible (this belief is formally described as P≠NP). So, the fact that the problem is NP-hard means that no matter what algorithm we use, there will always be some cases for which the running time grows faster than any polynomial, and therefore, for these cases the problem is intractable. In other

words: no practical algorithm is possible that finds the optimal solution to any non-singular interval linear system.

Theorem. *The problem of computing an optimal solution to a consistent non-singular interval linear system is NP-hard.*

Comment. A similar results was announced in [5]. It has also been recently proved [9] that checking whether a square matrix is non-singular is NP-hard.

Another case when computing an optimal interval estimate is NP-hard is given in [2]: namely, it is proved there that computing the range $P(\mathbf{x}_1, ..., \mathbf{x}_n)$ of a given polynomial $P(x_1, ..., x_n)$ of several variables $x_1, ..., x_n$ from given intervals of values $\mathbf{x}_1, ..., \mathbf{x}_n$ is NP-hard.

3. Proof

To prove that our problem is NP-hard, we will prove that if it were possible to solve it in polynomial time, then it would be possible to solve in polynomial time a problem that is already known to be NP-hard: the so-called satisfiability problem for 3-CNF (see, e.g., [3]). This problem consists of the following: suppose that an integer v is fixed, and a formula F of the type $F_1\&F_2\&...\&F_k$ is given, where each of the expressions F_i has the form $a \vee b$ or $a \vee b \vee c$, and a, b, c are either the variables $z_1, ..., z_v$, or their negations $\tilde{z}_1, ..., \tilde{z}_v$ (these $a, b, c, ...$ are called *literals*). If we assign arbitrary logical values ("true" or false") to v variables $z_1, ..., z_v$, then, applying the standard logical rules, we get the truth value of F. We say that a formula F is *satisfiable* if there exist truth values $z_1, ..., z_v$ for which the truth value of the expression F is "true". The problem is, given F, to check whether it is satisfiable.

The reduction will be as follows. Let us start with a 3-CNF propositional formula F of the type $F_1\&F_2\&...\&F_k$ with v Boolean variables $z_1, ..., z_v$ (i.e., variables that can take only two values: "true" and "false"). Let us build an interval linear system as follows. This system will have $n = 2v + 2$ variables $x_1, ..., x_v, x_{v+1}, ..., x_{2v}, x_{2v+1}, x_n$, and the following equations:

1) $v + 1$ equations $[-2, 2]x_i = [1, 2]$, $1 \le i \le v + 1$;
2) $v + 1$ equations $[-1, -1]x_i + [1, 1]x_{v+i+1} = [0.5, 0.5]$, $1 \le i \le v + 1$;
3) $v + 1$ equations $[1, 1]x_{v+i+1} = [0, 1]$, $1 \le i \le v + 1$;
4) k equations that correspond to $F_1, ..., F_k$: namely, if $F_j = a \vee b \vee c$, then the equation $t(a) + t(b) + t(c) + [0, 1]x_n = [1, 3]$, where $t(z_i) =$

x_{v+1+i} and $t(\tilde{z}_i) = 1 - x_{v+1+i}$, and if $F = a \vee b$, then the equation $t(a) + t(b) + [0,1]x_n = [1,2]$.

As a result, we get an interval linear system with $n = 2v + 2$ variables and $N = 3(v+1) + k$ equations. The time that it took us to design this system is evidently bounded by a polynomial of v.

Example. Let us take $F = (z_1 \vee z_2)\&(z_1 \vee \tilde{z}_2)$. Here, $k = v = 2$, so we have the following linear system:

$$[-2,2]x_1 = [1,2]$$
$$[-2,2]x_2 = [1,2]$$
$$[-2,2]x_3 = [1,2]$$
$$[-1,-1]x_1 + [1,1]x_4 = [0.5,0.5]$$
$$[-1,-1]x_2 + [1,1]x_5 = [0.5,0.5]$$
$$[-1,-1]x_3 + [1,1]x_6 = [0.5,0.5]$$
$$[1,1]x_4 = [0,1]$$
$$[1,1]x_5 = [0,1]$$
$$[1,1]x_6 = [0,1]$$
$$[1,1]x_4 + [1,1]x_5 + [0,1]x_6 = [1,2]$$

$x_4 + (1 - x_5) + [0,1]x_6 = [1,2]$, or $[1,1]x_4 + [-1,-1]x_5 + [0,1]x_6 = [0,1]$. End of example.

We will now prove the following three statements:

i) for every formula F this system is consistent and non-singular;
ii) if a formula F is satisfiable, then $[x_n^-, x_n^+] = [0,1]$;
iii) if a formula F is not satisfiable, then $[x_n^-, x_n^+] = [1,1]$.

If we prove that, then we will be able to prove our theorem. Indeed, suppose that there exists an algorithm that finds an optimal solution of any consistent non-singular interval linear system in polynomial time (i.e., in time that does not exceed some polynomial of n). Let us show that this algorithm will enable us to check satisfiability in polynomial time. Indeed, for any 3-CNF formula F, we form an interval linear system (as above; it takes a polynomial time) and apply the hypothetic algorithm to compute its optimal solution. If $x_n^- = 0$, then F is satisfiable; if $x_n^- = 1$, then F is not satisfiable. The running time of this algorithm is polynomial in $N = 3(v+1) + k$ and thus polynomial in v.

So, to complete the proof of our theorem, it is sufficient to prove the above three statements *i*) – *iii*).

1. The above-described system is consistent.

To prove that, let us show that the following x is a possible solution: $x_i = -0.5$, $1 \le i \le v$, $x_{v+1} = 0.5$, $x_{v+1+i} = 0, 1 \le i \le v$, and $x_n = 1$. Indeed,

1) The equations $[-2,2]x_i = [1,2]$, $1 \le i \le v+1$, are satisfied because $(-2)x_i = 1$ for $i \le v$ (where $-2 \in [-2,2]$ and $1 \in [1,2]$), and $2x_{v+1} = 1$.

2) The equations $[-1,-1]x_i + [1,1]x_{v+i+1} = [0.5,0.5]$ are satisfied for all $1 \le i \le v+1$.

3) The equations $[1,1]x_{v+i+1} = [0,1]$, $1 \le i \le v+1$, are evidently satisfied;

4) Each equation $t(a) + t(b) + t(c) + [0,1]x_n = [1,3]$ is satisfied for the following reason: each of the values $t(a)$, $t(b)$, and $t(c)$, is equal either to 0, or to 1. Therefore, $t(a) + t(b) + t(c)$ is equal to either 0, or 1, or 2, or 3. If this sum is equal to 1, 2, or 3, then $t(a) + t(b) + t(c) + 0 \cdot x_n \in [1,3]$. If $t(a) + t(b) + t(c) = 0$, then $t(a) + t(b) + t(c) + 1 \cdot x_n = 1 \in [1,3]$.

Similarly, the equations $t(a) + t(b) + [0,1]x_n = [1,2]$ are satisfied. So, the system is consistent.

2. Let us now prove that this system is non-singular.

Indeed, according to equations 3), $x_{v+1+i} \in [0,1]$, and from this and equations 2), we conclude that $x_i = x_{v+1+i} - 0.5 \in [-0.5, 0.5]$ for $i \le v+1$. Therefore, for each of the variables x_i, its area of possible values is bounded. So, the system is non-singular.

3. Before we start proving two other properties, let us first prove that for any possible solution of this system, $x_{v+1+i} \in \{0,1\}$ for $i \le v+1$.

Indeed, according to equations 1), $[-2,2]x_i = [1,2]$. Therefore, if x_i is a possible solution, there exists values r and s such that $rx_i = s$, $r \in [-2,2]$, and $s \in [1,2]$.

Since $s = rx_i \in [1,2]$, we have $rx_i \ne 0$, hence $x_i \ne 0$. If $x_i > 0$, then from $rx_i > 0$, we conclude that $r > 0$, so $0 < r \le 2$. From $s \ge 1$ and $0 < r \le 2$, we conclude that $x = s/r \ge 1/2$. Likewise, if $x_i < 0$, we can conclude that $x_i \le -0.5$.

Therefore, $x_i \in (-\infty, -0.5] \cup [0.5, \infty)$ for $i \le v+1$.

According to equations 2), $x_{v+1+i} = x_i + 0.5$. Therefore, $x_{v+1+i} \in (-\infty, 0] \cup [1, \infty)$, i.e., either $x_{v+1+i} \le 0$, or $x_{v+1+i} \ge 1$.

According to equations 3), $x_{v+1+i} \in [0,1]$. So, values < 0 and > 1 are not possible. Therefore, either $x_{v+1+i} = 0$, or $x_{v+1+i} = 1$.

4. In particular, 3. means that for possible solution x, x_n can take only the values 0 and 1. We have already proved (in 1.) that 1 is a possible value of x_n. Let us now prove that 0 is a possible value of x_n if and only if F is satisfiable. This will prove *ii*) and *iii*), and thus complete the proof of the theorem.

4.1. First, assume that F is satisfiable, and z_i are corresponding truth values. Let us show that in this case the following vector x is a possible solution: $x_n = 0$, $x_{v+1} = -0.5$; for $1 \le i \le v$, $x_{v+1+i} = 1$ iff $z_i =$"true", and $x_i = x_{v+1+i} - 0.5$.

1) $[-2,2]x_i = [1,2]$ is satisfied, because either $x_i = -0.5$ (then $(-2)x_i = 1$), or $x_i = 0.5$, then $2x_i = 1$.

2)$[-1,-1]x_i + [1,1]x_{v+i+1} = [0.5, 0.5]$ is satisfied.

3) Equations $[1,1]x_{v+i+1} = [0,1]$ are trivially true.

4) Each of the values $t(a)$, $t(b)$, $t(c)$ equals 0 or 1. Therefore, the sum $t(a) + t(b) + t(c) + [0,1]x_n = t(a) + t(b) + t(c)$ is equal to one of the 4 numbers 0, 1, 2, and 3. Since the values $z_1, z_2, ..., z_k$ satisfy F, the truth value of F is "true". Therefore, each of the subformulas F_j is true, which means that for each j, at least one of the expressions a, b, or c, is true. If a is true, then, according to our assignment, $t(a) = 1$. Therefore, $[0,1]x_n + t(a) + t(b) + t(c)$ is at least 1. Hence, $t(a) + t(b) + t(c) + [0,1]x_n \in [1,3]$. So, these equations are also satisfied.

4.2. Now, assume that x_i is a possible solution, and $x_n = 0$. Let us show that the formula F is satisfiable. We will show, that the following set of Boolean value makes it true: $z_i =$"true" iff $x_{v+1+i} = 1$.

Indeed, according to 3., for every $i \le v$, x_{v+1+i} is equal either to 0, or to 1. Hence, for every a, either $t(a) = 0$ or $t(a) = 1$, and $t(a) = 1$ iff a is true. Since $x_n = 0$, for every F_j, the corresponding sum is equal to

$t(a) + t(b) + t(c) + [0,1]x_n = t(a) + t(b) + t(c)$. Because of the equation 4), this sum is ≥ 1. This means that at least one of its terms $t(a)$ is equal to 1. This, in its turn, means that at least one of the literals a is true. Therefore, the formula $F_j = a \vee b \vee c$ is true for all j. Therefore, $F = F_1 \& ... \& F_j \& ... \& F_k$ is true. Q.E.D.

Acknowledgments. This work was sponsored by NSF grant No. CDA-9015006, NASA Research Grant No. 9-482, and a Grant No. PF90–018 from the General Services Administration (GSA), administered by the Materials Research Institute. The authors are thankful to S. P. Shary, Jiri Rohn for interesting reprints and stimulating discussions, and also to the anonymous referee for valuable comments.

References

1. Beeck, H. *Über die Struktur und Abschätzungen der Lösungsmenge von linearen Gliechungssystem mit Intervallkoeffizienten*. Computing **10** (1972), pp. 231–244.

2. Gaganov, A.A. *Computational complexity of the range of the polynomial in several variables*. Kibernetika (1985), pp. 418–421 (in Russian).

3. Garey, M. and Johnson, D. *Computers and intractability: a guide to the theory of NP-completeness*. Freeman, San Fransisco, 1979.

4. Lakeyev, A.V. and Noskov, S.I. *On controlled solution set to interval algebraic systems* (to appear).

5. Lakeyev, A.V. and Noskov, S.I. *A description of the set of solutions of a linear equation with intervally defined operator and right-hand side*. Doklady RAN, Matematika **330** (4) (1993), pp. 430–433 (in Russian).

6. Neumaier, A. *Interval methods for systems of equations*. Cambridge University Press, Cambridge, 1990.

7. Nickel, K. *Die Überschätzung des Wertebereichs einer Funktion in der Intervall-rechnung mit Anwendungen auf lineare Gleichungssysteme*. Computing **18** (1977), pp. 15–36.

8. Oettli, W. *On the solution set of a linear system with inaccurate coefficients*. SIAM J. Numer. Anal. **2** (1965), pp. 115–118.

9. Poljak, S. and Rohn, J. *Checking robust non-singularity is NP-hard*. Mathematics of Control. Signals, and Systems **6** (1993), pp. 1–9.

10. Rohn, J. *Systems of linear interval equations*. Linear Algebra and its Applications **126** (1989), pp. 39–78.

11. Shary, S.P. *Optimal solution of interval linear algebraic systems. I.* Interval Computations 2 (1991), pp. 7–30.

12. Shary, S.P. *A new class of algorithms for optimal solution of interval linear systems.* Interval Computations 2(4) (1992) pp. 18–29.

Received: May, 5, 1993

Revised version: June 15, 1993

V.Kreinovich
Department of
Computer Science,
University of Texas
at El Paso, El Paso,
TX 79968
USA

A.V.Lakeyev, S.I.Noskov
Irkutsk Computing
Center, Russian
Academy of Sciences,
Siberian Division
Lermontov Str. 134,
Irkutsk 664033,
Russia

Interval Computations
No 1, 1993

A PRECONDITIONER SELECTION HEURISTIC FOR EFFICIENT ITERATION WITH DECOMPOSITION OF ARITHMETIC EXPRESSIONS FOR NONLINEAR ALGEBRAIC SYSTEMS

R. Baker Kearfott and Xiaofa Shi*

We have recently considered decomposing a system of nonlinear equations by defining new variables corresponding to the intermediate results in the evaluation process. In that previous work, we applied both a derivative-free component solution process and an interval Gauss–Seidel method to the large, sparse system of equations so obtained.

An analysis of the component solution process indicates when a linearized Gauss–Seidel step is necessary, and how to make it more effective. In this paper, we will present preliminary results on an improved, efficient hybrid algorithm combining the component solution process with only an occasional Gauss–Seidel step on a single component.

ЭВРИСТИКА ВЫБОРА ПРЕДОБУСЛАВЛИВАТЕЛЯ ДЛЯ ЭФФЕКТИВНОЙ ИТЕРАЦИИ С ДЕКОМПОЗИЦИЕЙ АРИФМЕТИЧЕСКИХ ВЫРАЖЕНИЙ ДЛЯ НЕЛИНЕЙНЫХ АЛГЕБРАИЧЕСКИХ СИСТЕМ

Р.Б. Кирфотт, К. Ши

Ранее нами было рассмотрено разложение системы нелинейных уравнений путем определения новых переменных в соответствии с промежуточными результатами в процессе вычисления. В этой предыдущей работе мы применили покомпонентный процесс решения без производных и интервальный метод Гаусса-Зайде-

*This work was supported in part by National Science Foundation grants CCR-9203730 and DMS 9216120.