Solving Linear Interval Systems Is NP-Hard Even If We Exclude Overflow and Underflow

GERHARD HEINDL

Bergische Universität GH Wuppertal, Fachbereich Mathematik, Gaussstr. 20, 42097 Wuppertal, Germany, e-mail: heindl@math.uni-wuppertal.de

VLADIK KREINOVICH

Department of Computer Science, University of Texas at El Paso, El Paso, TX 79968, USA, e-mail: vladik@cs.utep.edu

and

ANATOLY V. LAKEYEV

Irkutsk Computing Center, Russian Academy of Science, Siberian Branch, Lermontov Str. 134, Irkutsk 664033, Russia, e-mail: lakeyev@icc.ccsoan.irkutsk.su

Motivation. It has been recently proven that the problem of finding exact (or even ε-approximate) componentwise bounds for the solution set X of an linear interval system $\sum \mathbf{a}_{ij} x_j = \mathbf{b}_i$ with interval coefficients $\mathbf{a}_{ij} = [a_{ij}^-, a_{ij}^+]$ and $\mathbf{b}_i = [b_i^-, b_i^+]$ is NP-hard [2]–[4]. Here, each interval coefficient is assumed to be rational (i.e., both bounds of this interval are rational numbers), and the solution set is defined as the set of all possible x_j for which $\sum a_{ij} x_j = b_i$ for some $a_{ij} \in \mathbf{a}_{ij}$ and $b_i \in \mathbf{b}_i$.

This result is true even if we restrict ourselves to systems with *square* interval matrices that are *regular*, i.e., for which each matrix $a_{ij} \in \mathbf{a}_{ij}$ is regular.

In the proof of this result, a hard-to-solve class C of interval linear systems is described such that if we can solve systems from the class C in polynomial time, then we will be able to solve all problems from a class NP in polynomial time; this proves that the problem of solving linear interval systems is NP-hard. For this class C, the bounds $a_{ij}^{\pm}$ and $b_i^{\pm}$ of the corresponding intervals $\mathbf{a}_{ij}$ and $\mathbf{b}_i$ are *rational* numbers, i.e., exactly the numbers that can be represented in the existing computers. In this sense, the problems from the class C are realistic. However, they are not completely realistic because in systems from C, some bounds of coefficient intervals grow (tend to ∞) with the number n of equations. So, if we try to represent these coefficients in the existing computers, we will, for large n, get an *overflow*. (Or, if we re-scale the coefficients so as to avoid the overflow, we will get an *underflow* for the other coefficients that will be re-scaled to almost 0.)

Problem and the main result. A natural question is: *If we restrict ourselves to interval linear systems with no underflow and no overflow*, i.e., to systems in which all the bounds of all coefficient intervals are binary numbers of the type

$d_{-s} \ldots d_{-1} d_0 . d_1 d_2 \ldots d_t$ for fixed s and t, *will the solution problem still be NP-hard?* Our answer is: "yes."

Moreover, we show that the answer is "yes, NP-hard" even if we restrict ourselves to regular linear interval systems in which every bound of an interval coefficient is either equal to 0, or to 1; in other words, to *regular linear interval systems in which each coefficient interval coefficient is equal to* [0, 0], [1, 1], *or to* [0, 1].

Comment. It is known that NP-hard numerical problems can be, crudely speaking, of two types (see, e.g., [1], Section 4.2):

• Problems that are, in general, NP-hard, but for which a polynomial time algorithm is possible if we restrict ourselves to instances in which the lengths of all numerical coefficients are bounded by a constant C. Such problems are called *pseudo-polynomial.*

• Problems that remain NP-hard even if we restrict ourselves to instances in which the lengths of all numerical coefficients are bounded by some constant C. Such problems are called *NP-hard in the strong sense.*

In these terms, our result shows that *the problem of finding exact (or ε-approximate) componentwise bounds for the solution set of a linear interval system is NP-hard in the strong sense.*

Proof. To prove our result, we will start with the known result (mentioned above) that the problem of solving regular linear interval systems with rational interval coefficients is NP-hard. We will then describe a general transformation of such systems into systems in which each interval coefficient is [0, 0], [1, 1], or [0, 1]. This transformation will be done in several steps.

On some steps, we will introduce new variables in addition to the variables $x_1, \ldots, x_n$ used in the original system. We will make sure that for each of the original variables x_j, the set of possible values of x_j will remain the same. Thus, we will be sure that the resulting final system has exactly the same bounds for $x_1, \ldots, x_n$ as the original system. Thus, if there is a polynomial-time algorithm that can find these bounds for an arbitrary linear interval system with coefficients [0, 0], [1, 1], or [0, 1], then by applying this algorithm to the transformation result, we would be able to compute the bounds for the original system, and this computation is an NP-hard problem. Thus, the problem of computing solution bounds for linear interval systems with coefficients [0, 0], [1, 1], or [0, 1] is also NP-hard.

We will also make sure that each transformation step preserves the number of non-degenerate interval coefficients (i.e., coefficients that are not of the type [a, a]), and that when we choose some values inside these intervals, all the variables of the resulting systems are uniquely determined. In other words, we will make sure that the systems obtained on each transformation step are *regular.* Thus, the problem of computing solution bounds for *regular* linear interval systems with coefficients [0, 0], [1, 1], or [0, 1] is also NP-hard.

1) The first transformation simplifies the right-hand side of the linear equation. Namely, we introduce a new variable x_0, replace each of n equations $\sum a_{ij}x_j = b_i$ by an equation $a_{i0}x_0 + \sum a_{ij}x_j = 0$ with $a_{i0} = -b_i$, and add a new equation $x_0 = 1$.

If the vector $(x_0, ..., x_n)$ belongs to the solution set of the new system, then $x_0 = 1$, and thus, the values $x_1, ..., x_n$ satisfy the original equations. Vice versa, if the values $x_1, ..., x_n$ satisfy the original equations, then for $x_0 = 1$, the transformed equations also hold. Thus, for each of the variables $x_1, ..., x_n$, the bounds are the same for the original and for the transformed equations.

If we fix the values $a_{ij} \in \mathbf{a}_{ij}$, then, since the original system was regular, the values $x_1, ..., x_n$ would be uniquely determined. The only missing value x_0 can be now uniquely determined from the equation $x_0 = 1$. Thus, the new system is regular.

Hence, this transformation preserves both the bounds on x_i and the uniqueness (regularity) property.

2) For every i, all the bounds of all the coefficients $\mathbf{a}_{ij}$ in i-th equation are rational numbers (i.e., fractions). If we multiply all coefficients of this equation by the least common denominator of the corresponding fractions, we get a new equation in which all bounds of all coefficients are integers.

In the second transformation, we apply this procedure to all equations. As a result, we get an equivalent (thus regular) linear interval systems, in which all bounds of all interval coefficients are integers.

In particular, the only equation with a the non-zero right-hand side (namely, the equation $x_0 = 1$) stays the same.

3) On the third step, for each $j = 0, ..., n$, we introduce two new variables n_j and p_j (here, n stands for negative, and p for positive).

For each j, we add two new equations $x_j + n_j = 0$ and $n_j + p_j = 0$. In each original equation, we replace each term $[a_{ij}^-, a_{ij}^+]x_j$ by one of the following six expressions:

- If the interval $[a_{ij}^-, a_{ij}^+]$ is non-degenerate (i.e., if $a_{ij}^- < a_{ij}^+$), then we use one of the following three expressions:

  - If $a_{ij}^- < 0$, then we replace the term $[a_{ij}^-, a_{ij}^+]x_j$ by the sum $|a_{ij}^-|n_j + [0, a_{ij}^+ - a_{ij}^-]x_j$.

  - If $a_{ij}^- = 0$, then we leave the term $[a_{ij}^-, a_{ij}^+]x_j$ $(= [0, a_{ij}^+]x_j)$ unchanged.

  - If $a_{ij}^- > 0$, then we replace this term by the sum $a_{ij}^- p_j + [0, a_{ij}^+ - a_{ij}^-]x_j$.

- If the interval $[a_{ij}^-, a_{ij}^+]$ is degenerate (i.e., if $a_{ij}^- = a_{ij}^+$), then we use one of the following two expressions:

  - If $a_{ij}^- < 0$, then we replace the term $[a_{ij}^-, a_{ij}^+]x_j$ by $|a_{ij}^-|n_j$.

  - If $a_{ij}^- = 0$, then we delete the term $[a_{ij}^-, a_{ij}^+]x_j$ (because it is equal to 0).

  - If $a_{ij}^- > 0$, then we replace the term $[a_{ij}^-, a_{ij}^+]x_j$ by $a_{ij}^- p_j$.

As a result, we get a linear interval system in which each interval coefficient is either a positive integer, or an interval of the type $[0, z]$ for some positive integer z.

Let us show that this transformation does not change the bounds of the solution set for $x_1, ..., x_n$, and preserves regularity.

Indeed, if $x_0, x_1, ..., x_n$ is a solution of the system that we had before this step, then, by adding $n_j = -x_j$ and $p_j = x_j$, we get a solution of the transformed system. Vice versa, if we have a solution of the transformed system, then from the new equations $x_j + n_j = 0$ and $n_j + p_j = 0$, we conclude that $n_j = -x_j$, $p_j = -n_j = x_j$. If $a_{ij}^- < 0$ and $a_{ij}^- < a_{ij}^+$, then, for every $z_{ij} \in [0, a_{ij}^+ - a_{ij}^-]$, we have

$$|a_{ij}^-| \, |n_j + z_{ij}x_j = a_{ij}^- x_j + z_{ij}x_j = a_{ij}x_j,$$

where $a_{ij} = a_{ij}^- + z_{ij} \in a_{ij}^- + [0, a_{ij}^+ - a_{ij}^-] = [a_{ij}^-, a_{ij}^+]$. Thus, $x_0 = 1, x_1, ..., x_n$ form a solution of the system that we had before this step. (A similar proof holds in all five other cases: $a_{ij}^- = 0 < a_{ij}^+$, $0 < a_{ij}^- < a_{ij}^+$, $a_{ij}^- = a_{ij}^+ < 0$, $a_{ij}^- = a_{ij}^+ = 0$, and $a_{ij}^- = a_{ij}^+ > 0$.)

Since the system that we had before this step was regular, for each choice of coefficients in their intervals, the variables $x_0(= 1), x_1, ..., x_n$ are uniquely determined; the new variables n_j and p_j are uniquely determined from x_j. Thus, the transformed system is also regular.

4) Let us now describe the fourth (and final) transformation step. To simplify the description of this step, let us first rename the variables x_j, n_j, and p_j into $x_j^{[0]}$, $n_j^{[0]}$, and $p_j^{[0]}$.

Let N denote the largest integer bound in the system obtained after the third step, let d denote the number of binary digits in the binary representation of N (i.e., $d = \lfloor \log_2(N) \rfloor$), and let I denote the set of all pairs (i, j) for which i-th equation of the system that we had before this step contains the term $[0, z_{ij}]x_j$. The transformed system will have the following variables:

- For each $j = 0, ..., n$, and for each $k = 0, ..., d$, variables $x_j^{[k]}$, $n_j^{[k]}$, and $p_j^{[k]}$.

- For each pair $(i, j) \in I$, and for each $k = 0, ..., d$, variables $x_{ij}^{[k]}$, $n_{ij}^{[k]}$, and $p_{ij}^{[k]}$.

This system will consist of the following equations:

- Equations $x_j^{[k]} + n_j^{[k]} = 0$ and $p_j^{[k]} + n_j^{[k]} = 0$ (for all $j \leq n$).

- Equations $x_{ij}^{[k]} + n_{ij}^{[k]} = 0$ and $p_{ij}^{[k]} + n_{ij}^{[k]} = 0$ (for all $(i, j) \in I$).

- Equations $x_j^{[k]} + p_j^{[k]} + n_j^{[k+1]} = 0$ (for all $j \leq n$ and $k < d$).

- Equations $x_{ij}^{[k]} + p_{ij}^{[k]} + n_{ij}^{[k+1]} = 0$ (for all $(i, j) \in I$ and $k < d$).

- Equations $n_{ij}^{[0]} + [0, 1]x_j^{[0]} = 0$ (for all $(i, j) \in I$).

- Equations that are obtained from the equations of the system that we had before this step by the following replacement:

- Each term of the type zn_j is replaced by the sum of the terms $n_j^{[k]}$ for all places k on which the binary expansion of z has 1 (i.e., for which $\varepsilon_k = 1$ in the binary expansion $z = \sum \varepsilon_k 2^k$).

- Each term of the type zp_j is replaced by the sum of the terms $p_j^{[k]}$ for all places k on which the binary expansion of z has 1.

- Each term of the type $[0, z_{ij}]x_j$ is replaced by the sum of the terms $x_{ij}^{[k]}$ for all places k on which the binary expansion of z_{ij} has 1.

As a result, we get a linear interval system in which each interval coefficient is either 0, or 1, or an interval $[0, 1]$.

Let us show that this transformation does not change the bounds of the solution set for $x_1, \ldots, x_n$, and preserves regularity.

Indeed, if the values x_j, n_j, and p_j form a solution of the system that we had before this step, a solution that corresponds to the coefficients $c_{ij} \in [0, z_{ij}]$, then, as one can check, the values $x_j^{[k]} = p_j^{[k]} = 2^k \cdot x_j$, $n_j^{[k]} = -2^k \cdot x_j$, $c'_{ij} = c_{ij} / z_{ij}$, $x_{ij}^{[k]} = p_{ij}^{[k]} = 2^k \cdot c'_{ij} \cdot x_j$, and $n_{ij}^{[k]} = -2^k \cdot c'_{ij} \cdot x_j$ form a solution of the transformed system, for $c'_{ij} \in [0, 1]$.

Indeed, e.g., for this choice of variables, the sum of the terms $p_j^{[k]}$ for all places k on which the binary expansion of z has 1, is equal to the sum of the terms $2^k x_j$, i.e., to the product of x_j and the sum of the terms 2^k that corresponds to all places k on which the binary expansion of z has 1. This sum is exactly the binary expansion of z, and hence, the sum is equal to zx_j.

Vice versa, if we have a solution of the transformed system, for $c_{ij} \in [0, 1]$, then:

- From the equations $x_j^{[k]} + n_j^{[k]} = 0$ and $p_j^{[k]} + n_j^{[k]} = 0$, we conclude that $n_j^{[k]} = -x_j^{[k]}$ and $p_j^{[k]} = -n_j^{[k]} = x_j^{[k]}$.

- From the equations $x_j^{[k]} + p_j^{[k]} + n_j^{[k+1]} = 0$, we can now conclude that $n_j^{[k+1]} = -2x_j^{[k]}$, hence, $x_j^{[k+1]} = -n_j^{[k+1]} = 2x_j^{[k]}$. By induction over k, we can now conclude that $x_j^{[k]} = 2^k \cdot x_j^{[0]}$, and hence, that $n_j^{[k]} = -2^k \cdot x_j^{[0]}$ and $p_j^{[k]} = 2^k \cdot x_j^{[0]}$.

- Similarly, we can conclude that $x_{ij}^{[k]} = 2^k \cdot x_{ij}^{[0]}$, $n_{ij}^{[k]} = -2^k \cdot x_{ij}^{[0]}$, and $p_{ij}^{[k]} = 2^k \cdot x_{ij}^{[0]}$.

- From the equation $n_{ij}^{[0]} + [0, 1]x_j^{[0]} = 0$, and from our assumption that the coefficient is equal to $c_{ij} \in [0, 1]$, we conclude that $n_{ij}^{[0]} = -c_{ij}x_j^{[0]}$, and therefore, that $p_{ij}^{[0]} = -n_{ij}^{[0]} = c_{ij}x_j^{[0]}$.

Thus, the sums of the terms of the type $n_j^{[k]}$ and $x_{ij}^{[k]}$ reduce to zn_j and $zx_{ij} = (zc_{ij})x_j$, where $zc_{ij} \in z \cdot [0, 1] = [0, z]$. Thus, the values $x_0 = 1, x_1, \ldots, x_n$ form a solution of the system that we had before this step.

Since the system that we had before this step was regular, for each choice of coefficients in their intervals, the variables $x_0 (= 1), x_1, \ldots, x_n$ are uniquely determined; the new variables $x_j^{[k]}$, $n_j^{[k]}$, and $p_j^{[k]}$ are uniquely determined by the values

x_j. For fixed c_{ij}, the values of the variables $x_{ij}^{[k]}$, $n_{ij}^{[k]}$, and $p_{ij}^{[k]}$ are also uniquely determined by the values x_j. Thus, the transformed system is also regular.

The theorem is proven.　　　　　　　　　　　　　　　　　　　　　□

References

1. Garey, M. R. and Johnson, D. S.: *Computers and Intractability: A Guide to Theory of NP-Completeness*, Freeman, San Francisco, 1979.
2. Kreinovich, V., Lakeyev, A. V., and Noskov, S. I.: Optimal Solution of Interval Linear Systems Is Intractable (NP-Hard), *Interval Computations* 1 (1993), pp. 6–14.
3. Rohn, J.: Linear Interval Equations: Computing Enclosures with Bounded Relative Overestimation Is NP-Hard, in: Kearfott, R. B. and Kreinovich, V. (eds), *Applications of Interval Computations*, Kluwer, Boston, MA, 1996, pp. 81–89.
4. Rohn, J. and Kreinovich, V.: Computing Exact Componentwise Bounds on Solutions of Linear Systems with Interval Data Is NP-Hard, *SIAM Journal on Matrix Analysis and Applications (SIMAX)* 16 (1995), pp. 415–420.