For Interval Computations, If Absolute-Accuracy Optimization is NP-Hard, Then So Is Relative-Accuracy Optimization

Vladik Kreinovich
Department of Computer Science
University of Texas at El Paso
El Paso, TX 79968, USA
email vladik@cs.utep.edu

Abstract

One of the basic problems of interval computations is to compute a range of a given function $f(x_1, \ldots, x_n)$ over a given box

$$[\underline{x}_1, \overline{x}_1] \times \ldots \times [\underline{x}_n, \overline{x}_n]$$

(i.e., to compute the maximum and the minimum of the function on the box). For many classes of functions (e.g., for quadratic functions) this problem is NP-hard; it is even NP-hard if instead of computing the minimum and maximum *exactly*, we want to compute them with a given (absolute) *accuracy* $\varepsilon > 0$. In practical situations, it is more realistic to ask for a *relative* accuracy; are the corresponding problems still NP-hard? We show that under some reasonable conditions, NP-hardness of absolute-accuracy optimization implies that relative-accuracy optimization is NP-hard as well.

Data processing: a practical problem which leads to interval computations. In many real-life situations, we are interested in the value of some physical quantity y which is difficult (or even impossible) to measure directly. To estimate y, we measure directly measurable quantities $x_1, \ldots, x_n$ which have a known relationship with y, and then reconstruct y from the results $\widetilde{x}_1, \ldots, \widetilde{x}_n$ of these measurements by using this known relation: $\widetilde{y} = f(\widetilde{x}_1, \ldots, \widetilde{x}_n)$, where f is a known algorithm.

Measurements are never 100% accurate; as a result, the actual value x_i of each measured quantity may differ from the measured value $\widetilde{x}_i$. If we know the upper bound Δ_i for the measurement error $|\Delta x_i| = |\widetilde{x}_i - x_i|$, then after we get the measurement result $\widetilde{x}_i$, we can conclude that the actual value x_i of the measured quantity belongs to the *interval* $\mathbf{x}_i = [\widetilde{x}_i - \Delta_i, \widetilde{x}_i + \Delta_i]$. A natural question is: when $x_i \in \mathbf{x}_i$, what is the resulting interval $\mathbf{y} = f(\mathbf{x}_1, \ldots, \mathbf{x}_n) = \{f(x_1, \ldots, x_n) \,|\, x_i \in \mathbf{x}_i\}$ of possible values of y? The problem of estimating this range interval is called the problem of *interval computations* (see, e.g., [2, 3, 4, 5, 7]).

Interval computations as a particular case of global optimization. In optimization terms:

- the lower endpoint $\underline{y}$ of the range interval $\mathbf{y} = [\underline{y}, \overline{y}]$ is the infimum of the function $f(x_1, \ldots, x_n)$ on the box $\mathbf{x}_1 \times \ldots \times \mathbf{x}_n$, and
- the upper endpoint $\overline{y}$ of the range interval is the supremum of the function $f(x_1, \ldots, x_n)$ over the same box.

Interval computation is NP-hard for many reasonable classes of problems. It is known that the problem of interval computation is NP-hard for many reasonable classes of functions (see, e.g., [6]): e.g., it is NP-hard:

- for the class of all polynomials;
- for the class of all quadratic functions $f(x_1, \ldots, x_n) = a_0 + \sum a_i \cdot x_i + \sum a_{ij} \cdot x_i \cdot x_j$ (the proof of this particular result is due to Vavasis [10]);
- for several classes of quadratic functions with sparse matrices a_{ij};
- for all piecewise-linear functions, etc.

(For related problems, see, e.g., [9].)

For many interval computation problems, absolute-accuracy optimization is also NP-hard. Most of the above problems are NP-hard even if instead of looking for, say, the *precise* maximum $\overline{y}$ of a function f, we are looking for *absolute-accuracy* optimization, i.e., if we fix a real number $\varepsilon > 0$ and we are looking for values $\widetilde{y}$ which are (absolutely) ε-close to the (unknown) precise maximum, i.e., for which $|\widetilde{y} - \overline{y}| \le \varepsilon$.

Relative-accuracy optimization may be more realistic. From the practical viewpoint, the desire to have the bounds with an absolute accuracy may be too strong. If the actual maximum is large, it may be more reasonable to look for an estimate which is good within a certain *relative* accuracy, i.e., for which $|\widetilde{y} - \overline{y}| \le \varepsilon \cdot |\widetilde{y}|$ and $|\widetilde{y} - \overline{y}| \le \varepsilon \cdot |\overline{y}|$, where $\varepsilon > 0$ is a given relative accuracy (such as 1% or 0.01%).

For *small* values $\overline{y}$ thus defined relative accuracy is too strong: e.g., for $\overline{y} = 0$, the only way to compute this value with relative accuracy ε (in the sense of the above two inequalities) is to compute it precisely, because the only value $\widetilde{y}$ which satisfies these two inequalities is $\widetilde{y} = 0 = \overline{y}$. For small values $\overline{y}$, absolute accuracy makes more practical sense. So, a more realistic approach is to fix a value ε and to require that the computed value $\widetilde{y}$ and the actual value

$\overline{y}$ are either absolutely ε-close or strongly relatively ε-close (in the sense of the above two inequalities). In short, we arrive at the following definition (similar definitions were used in Ch. 11 of [6]):

Definition 1. *Let $\varepsilon > 0$.*

- *We say that two real numbers $\widetilde{a}$ and a are absolutely ε-close if $|\widetilde{a} - a| \le \varepsilon$.*
- *We say that two real numbers $\widetilde{a}$ and a are strongly relatively ε-close if $|\widetilde{a} - a| \le \varepsilon \cdot |\widetilde{a}|$ and $|\widetilde{a} - a| \le \varepsilon \cdot |a|$.*
- *We say that two real numbers $\widetilde{a}$ and a are relatively ε-close if they are either absolutely ε-close or strongly relatively ε-close.*

Is the corresponding optimization problem still NP-hard? The above notion of relative ε-closeness is weaker than the notion of absolute ε-closeness; so, in general, it may be possible that the absolute-accuracy optimization problem is NP-hard, while the weaker relative-accurate optimization problem is easier to solve. We will show, however, that for many important classes, the new problem is still NP-hard.

Definition 2. *We say that a class $\mathcal{F}$ of computable functions (from tuples of real numbers into real numbers) is shift-invariant if for every function $f \in \mathcal{F}$ and for every rational number c, the class $\mathcal{F}$ also contains the function $f + c$.*

Comment. All above classes are shift-invariant.

Definition 3. *We say that class $\mathcal{F}$ of functions is feasibly boundable if there exists a polynomial-time (feasible) algorithm which, given a function $f \in \mathcal{F}$ and a given box (with rational coordinates), produces a lower and an upper bound for the range of this function on this box.*

Definition 4. *We say that for a class of functions $\mathcal{F}$, absolute-accuracy optimization is NP-hard if for every $\varepsilon > 0$, the problem of computing a maximum M of a given function $f \in \mathcal{F}$ on a given box with an absolute accuracy ε (i.e., the problem of computing a value $\widetilde{M}$ which is absolutely ε-close to M) is NP-hard.*

Comment. All above classes are feasibly boundable, and for all above classes, absolute-accurate optimization is NP-hard.

Definition 5. *We say that for a class of functions $\mathcal{F}$, relative-accuracy optimization is NP-hard if for every $\varepsilon > 0$, the problem of computing a maximum M of a given function $f \in \mathcal{F}$ on a given box with a relative accuracy ε (i.e., the problem of computing a value $\widetilde{M}$ which is relatively ε-close to M in the sense of Definition 1) is NP-hard.*

Proposition. *If for some shift-invariant feasibly boundable class $\mathcal{F}$, absolute-accuracy optimization problem is NP-hard, then for this class, relative-accuracy optimization problem is also NP-hard.*

Comments.

- The condition that the class should be feasibly boundable is not really restrictive: indeed, if we are able to solve the relative-accuracy optimization problem in feasible time, then we get feasible bounds as well; therefore, if the class is not feasible boundable, then no feasible algorithm can solve the corresponding relative-accuracy optimization problem.

- As a corollary, we conclude that for quadratic polynomials (and for quadratic functions with a sparse matrix a_{ij}), the relative-accuracy optimization problem is NP-hard.

Proof. To prove the theorem, we will show that for shift-invariant feasibly boundable classes, the absolute-accuracy optimization problem can be polynomially Turing-reduced to the relative-accuracy optimization problem for this same class (i.e., that we can solve the absolute-accuracy optimization problem in polynomial time if we use the solver of the relative-invariant optimization problem as an oracle; for precise definitions, see, e.g., [1, 8]). Since the absolute-accuracy optimization problem is known to be NP-hard, we can thus conclude that the relative-accuracy optimization problem is NP-hard as well.

Let us show the desired reduction. Assume that we want to solve the absolute-accuracy optimization problem with an accuracy $\delta > 0$. This means that, given a function $f(x_1, \ldots, x_n)$ (from the class $\mathcal{F}$) and a box, we must find values which are absolutely δ-close to the minimum and to the maximum of the given function on a given box. Let us show how we can use the relative-accuracy oracle to solve this problem. Without losing generality, we will show how to compute the maximum with the accuracy δ (for minimum, the computation is similar).

First, since the function $f(x_1, \ldots, x_n)$ is feasibly boundable, we can apply the polynomial-time bounding algorithm and find a lower bound B_l and an upper bound B_u for the range of the function f on a given box. Then, for every $x = (x_1, \ldots, x_n)$ from the given box, the value $f(x)$ belongs to the interval $[B_l, B_u]$. In particular, the actual maximum M of the function is somewhere in this interval. Let us denote the width $B_u - B_l$ of this interval by Δ.

Let us fix a real number $\varepsilon > 0$ and an integer m (their values will be specified later). We can define, for every integer i from 0 to m, the values $c_i = B_l + i \cdot \Delta/m$. These values cover the entire interval $[B_l, B_u]$ so that every value from this interval (including the unknown actual maximum M) is at most (absolutely) $\Delta/(2m)$-close to one of these points. Since the class $\mathcal{F}$ is shift-invariant, for every integer i from 0 to m, the function $f_i = f - c_i$ also belongs to the class $\mathcal{F}$; the actual maximum of this function is $M_i = M - c_i$. Let us apply the oracle to the functions $f_0, \ldots, f_m$, and compute the ε-relative approximations $\widetilde{M}_i$ to these maxima.

For the value j for which $|M - c_j| \leq \Delta/(2m)$, we have $|M_j| = |M - c_j| \leq$

$\Delta/(2m)$. Therefore, since $\widetilde{M}_j$ should be relatively ε-close to M_j, we can conclude that:

- either $\left|\widetilde{M}_j - M_j\right| \le \varepsilon$, in which case

$$\left|\widetilde{M}_j\right| \le |M_j| + \left|\widetilde{M}_j - M_j\right| \le \frac{\Delta}{2m} + \varepsilon,$$

- or $\left|\widetilde{M}_j - M_j\right| \le \varepsilon \cdot |M_j|$, in which case

$$\left|\widetilde{M}_j\right| \le |M_j| + \left|\widetilde{M}_j - M_j\right| \le |M_j| + \varepsilon \cdot |M_j| = \frac{\Delta}{2m} + \varepsilon \cdot \frac{\Delta}{2m}.$$

In both cases,

$$\left|\widetilde{M}_j\right| \le \frac{\Delta}{2m} + \max\left(\varepsilon, \varepsilon \cdot \frac{\Delta}{2m}\right).$$

Since for every $a, b \ge 0$, we have $\max(a, b) \le a + b$, we can conclude that $|M_j| \le K$, where

$$K = \frac{\Delta}{2m} + \varepsilon + \varepsilon \cdot \frac{\Delta}{2m} = \varepsilon + \frac{\Delta}{2m} \cdot (1 + \varepsilon). \tag{1}$$

So, there always exists a k for which $\left|\widetilde{M}_k\right| \le K$. If we find such k, then, since M_k is relatively ε-close to $\widetilde{M}_k$, we can conclude that:

- either $\left|M_k - \widetilde{M}_k\right| \le \varepsilon$, in which case,

$$|M_k| \le \left|\widetilde{M}_k\right| + \left|M_k - \widetilde{M}_k\right| \le K + \varepsilon,$$

- or $\left|M_k - \widetilde{M}_k\right| \le \varepsilon \cdot \left|\widetilde{M}_k\right|$, in which case,

$$|M_k| \le \left|\widetilde{M}_k\right| + \left|M_k - \widetilde{M}_k\right| \le \left|\widetilde{M}_k\right| + \varepsilon \cdot \left|\widetilde{M}_k\right| \le K + \varepsilon \cdot K.$$

In both cases, $|M_k| \le K + \max(\varepsilon, K \cdot \varepsilon)$, and therefore (due to $\max(a, b) \le a + b$), $|M_k| \le B$, where

$$B = K + \varepsilon + \varepsilon \cdot K = \varepsilon + K \cdot (1 + \varepsilon). \tag{2}$$

Since $M_k = M - c_k$, we can thus conclude that $|M - c_k| \le B$, and therefore, the actual value of M belongs to the interval $[c_k - B, c_k + B]$ of width $\Delta' = 2B$.

Substituting the expression (2) for B into the formula for Δ', and then substituting the formula (1) for K, we can get the following expression for Δ':

$$\Delta' = 4\varepsilon + 2\varepsilon^2 + \frac{\Delta \cdot (1 + \varepsilon)^2}{m}. \tag{3}$$

If we choose m for which $(1+\varepsilon)^2/m \le 1/2$, e.g., if we choose $m = \lceil 2(1+\varepsilon)^2 \rceil$, then we can conclude that

$$\Delta' \le 4\varepsilon + 2\varepsilon^2 + \frac{\Delta}{2}. \tag{4}$$

Subtracting twice the ε-term from both sides of this inequality, we conclude that

$$\Delta' - (8\varepsilon + 4\varepsilon^2) \le \frac{\Delta - (8\varepsilon + 4\varepsilon^2)}{2}. \tag{5}$$

So, if originally, we had $\Delta > (8\varepsilon + 4\varepsilon^2)$, then the difference between Δ and the ε-term gets halved.

We can repeat the same construction, starting with the new maximum-containing interval $[c_k - B, c_k + B]$ of width Δ' (instead of the original interval $[B_l, B_u]$ of width Δ), etc., etc., until we reach (in polynomially many steps, i.e., in $\approx \log_2(\Delta/\varepsilon)$ steps) a maximum-containing interval for which the difference between its width Δ_{final} and the ε-expression $8\varepsilon + 4\varepsilon^2$ is $\le \varepsilon$. For this interval, $\Delta_{\text{final}} \le 9\varepsilon + 4\varepsilon^2$. So:

- either after the first step we already had a maximum-containing interval of width $\Delta \le 8\varepsilon + 4\varepsilon^2 < 9\varepsilon + 4\varepsilon^2$,

- or, in polynomial number of steps, we get a maximum-containing interval of width $\Delta_{\text{final}} \le 9\varepsilon + 4\varepsilon^2$.

In both cases, if we choose ε for which $9\varepsilon + 4\varepsilon^2 = \delta$ (such a value ε always exists and is easy to find), we will find an interval of width $\le \delta$ which contains the desired maximum M. Each endpoint of this interval is, therefore, the desired absolute δ-approximation to M. The reduction is proven, and so is the proposition.

Acknowledgments. This work was supported in part by NASA under cooperative agreement NCC5-209, by NSF grants No. DUE-9750858 and CDA-9522207, by United Space Alliance, grant No. NAS 9-20000 (PWO C0C67713A6), by the Future Aerospace Science and Technology Program (FAST) Center for Structural Integrity of Aerospace Systems, effort sponsored by the Air Force Office of Scientific Research, Air Force Materiel Command, USAF, under grant number F49620-95-1-0518, and by the National Security Agency under Grant No. MDA904-98-1-0564.

The author is thankful to all the participants of the Conference on Approximation and Complexity in Numerical Optimization: Continuous and Discrete Problems (Gainesville, Florida, February 28–March 2, 1999), especially to P. Pardalos and S. Vavasis, for valuable discussions.

References

[1] M. E. Garey and D. S. Johnson, *Computers and intractability: a guide to the theory of NP-completeness*, Freeman, San Francisco, 1979.

[2] R. Hammer, M. Hocks, U. Kulisch, D. Ratz, *Numerical toolbox for verified computing. I. Basic numerical problems*, Springer Verlag, Heidelberg, N.Y., 1993.

[3] E. R. Hansen, *Global optimization using interval analysis*, Marcel Dekker, N.Y., 1992.

[4] R. B. Kearfott, *Rigorous global search: continuous problems*, Kluwer, Dordrecht, 1996.

[5] R. B. Kearfott and V. Kreinovich (eds.), *Applications of Interval Computations*, Kluwer, Dordrecht, 1996.

[6] V. Kreinovich, A. Lakeyev, J. Rohn, and P. Kahl, *Computational complexity and feasibility of data processing and interval computations*, Kluwer, Dordrecht, 1997.

[7] R. Moore, *Methods and Applications of Interval Analysis*, SIAM, Philadelphia, 1979.

[8] C. H. Papadimitriou, *Computational Complexity*, Addison Wesley, San Diego, 1994.

[9] P. M. Pardalos, *Complexity in Numerical Optimization*, World Scientific, Singapore, 1993.

[10] S. A. Vavasis, *Nonlinear optimization: complexity issues*, Oxford University Press, N.Y., 1991.