Introduction to Interval Computation

Interval Programming 1

Milan Hladík [1] Michal Černý [2]

[1] Faculty of Mathematics and Physics,
Charles University in Prague, Czech Republic
http://kam.mff.cuni.cz/~hladik/

[2] Faculty of Computer Science and Statistics,
University of Economics, Prague, Czech Republic
http://nb.vse.cz/~cernym/

Workshop on Interval Programming
7th International Conference of Iranian Operation Research Society
Semnan, Iran, May 12–13, 2014

Outline

1. Motivation
2. Interval Computations
3. Interval Functions
4. Algorithmic Issues

Next Section

Introduction

Interval computation = solving problems with interval data.

Where interval data do appear

1. numerical analysis (handling rounding errors)
2. computer-assisted proofs
3. global optimization
4. modelling uncertainty
   (an alternative to fuzzy and stochastic programming)

Example (Rump, 1988)

Consider the expression

$$f = 333.75b^6 + a^2(11a^2b^2 - b^6 - 121b^4 - 2) + 5.5b^8 + \frac{a}{2b},$$

with

$$a = 77617, \quad b = 33096.$$

Calculations from 80s gave

$$\begin{array}{rl} \text{single precision} & f \approx 1.172603\ldots \\ \text{double precision} & f \approx 1.1726039400531\ldots \\ \text{extended precision} & f \approx 1.172603940053178\ldots \\ \text{the true value} & f = -0.827386\ldots \end{array}$$

Computer-Assisted Proofs

Kepler conjecture

What is the densest packing of balls? (Kepler, 1611)

That one how the oranges are stacked in a shop.

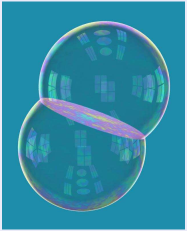
The conjecture was proved by T.C. Hales (2005).

Double bubble problem

What is the minimal surface of two given volumes?

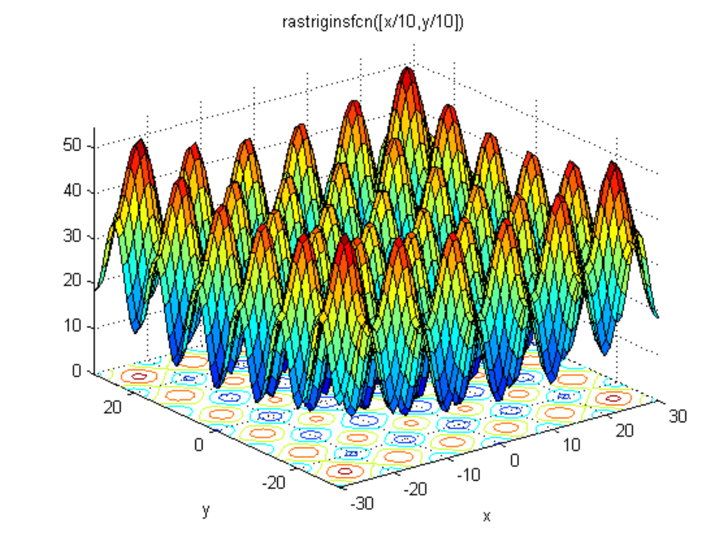
Two pieces of spheres meeting at an angle of 120°.

Hass and Schlafly (2000) proved the equally sized case.
Hutchings et al. (2002) proved the general case.



Global Optimization

Rastrigin's function $f(x) = 20 + x_1^2 + x_2^2 - 10(\cos(2\pi x_1) + \cos(2\pi x_2))$



Further Sources of Intervals

- Mass number of chemical elements (sue to several stable isotopes)
  - $[12.0096, 12.0116]$ for the carbon
- physical constants
  - $[9.78,\ 9.82]\, ms^{-2}$ for the gravitational acceleration
- mathematical constants
  - $\pi \in [3.14159265358979323 84, 3.14159265358979323 85]$.
- measurement errors
  - temperature measured $23°\mathrm{C} \pm 1°\mathrm{C}$
- discretization
  - time is split in days
  - temperature during the day in $[18, 29]°\mathrm{C}$ for Semnan in May
- missing data
  - What was the temperature in Semnan on May 12, 1999?
  - Very probably in $[10, 40]°\mathrm{C}$.
- processing a state space
  - find robot singularities, where it may breakdown
  - check joint angles $[0, 180]°$.

Next Section

Notation

An interval matrix

$$\mathbf{A} := [\underline{A}, \overline{A}] = \{A \in \mathbb{R}^{m \times n} \mid \underline{A} \leq A \leq \overline{A}\}.$$

The center and radius matrices

$$A^c := \frac{1}{2}(\overline{A} + \underline{A}), \quad A^\Delta := \frac{1}{2}(\overline{A} - \underline{A}).$$

The set of all $m \times n$ interval matrices: $\mathbb{IR}^{m \times n}$.

Main Problem

Let $f : \mathbb{R}^n \mapsto \mathbb{R}^m$ and $\mathbf{x} \in \mathbb{IR}^n$. Determine the image

$$f(\mathbf{x}) = \{f(x) : x \in \mathbf{x}\}.$$

Interval Arithmetic

Interval Arithmetic

$$\mathbf{a}+\mathbf{b} = [\underline{a}+\underline{b}, \overline{a}+\overline{b}],$$
$$\mathbf{a}-\mathbf{b} = [\underline{a}-\overline{b}, \overline{a}-\underline{b}],$$
$$\mathbf{a}\cdot\mathbf{b} = [\min(\underline{a}\underline{b}, \underline{a}\overline{b}, \overline{a}\underline{b}, \overline{a}\overline{b}), \max(\underline{a}\underline{b}, \underline{a}\overline{b}, \overline{a}\underline{b}, \overline{a}\overline{b})],$$
$$\mathbf{a}/\mathbf{b} = [\min(\underline{a}/\underline{b}, \underline{a}/\overline{b}, \overline{a}/\underline{b}, \overline{a}/\overline{b}), \max(\underline{a}/\underline{b}, \underline{a}/\overline{b}, \overline{a}/\underline{b}, \overline{a}/\overline{b})], \quad 0 \notin \mathbf{b}.$$

Theorem (Basic properties of interval arithmetic)

- *Interval addition and multiplication is commutative and associative.*
- *It is not distributive in general, but sub-distributive instead,*

$$\forall \mathbf{a}, \mathbf{b}, \mathbf{c} \in \mathbb{IR} : \mathbf{a}(\mathbf{b}+\mathbf{c}) \subseteq \mathbf{ab}+\mathbf{ac}.$$

Example ($\mathbf{a} = [1, 2]$, $\mathbf{b} = 1$, $\mathbf{c} = -1$)

$$\mathbf{a}(\mathbf{b}+\mathbf{c}) = [1, 2]\cdot(1-1) = [1, 2]\cdot 0 = 0,$$
$$\mathbf{ab}+\mathbf{ac} = [1, 2]\cdot 1 + [1, 2]\cdot(-1) = [1, 2] - [1, 2] = [-1, 1].$$

Exercises for YOU

Prove or Disprove

For $\mathbf{a}, \mathbf{b}, \mathbf{c} \in \mathbb{IR}$ and $d \in \mathbb{R}$:

1. $\mathbf{a} + \mathbf{b} = \mathbf{a} + \mathbf{c} \quad \Rightarrow \quad \mathbf{b} = \mathbf{c}$.
2. $(\mathbf{b} + \mathbf{c})d = \mathbf{b}d + \mathbf{c}d$
3. $(\underline{a} \geq 0 \text{ and } \mathbf{ab} = \mathbf{ac}) \quad \Rightarrow \quad \mathbf{b} = \mathbf{c}$.
4. $\mathbf{a} \subseteq \mathbf{b} \quad \Leftrightarrow \quad |a^c - b^c| + a^\Delta \leq b^\Delta$,
5. $\mathbf{a} \cap \mathbf{b} \neq \emptyset \quad \Leftrightarrow \quad |a^c - b^c| \leq a^\Delta + b^\Delta$,

Next Section

Images of Functions

Monotone Functions

If $f : \mathbf{x} \to \mathbb{R}$ is non-decreasing, then $f(\mathbf{x}) = [f(\underline{x}), f(\overline{x})]$.

Example

$\exp(\mathbf{x}) = [\exp(\underline{x}), \exp(\overline{x})]$, $\log(\mathbf{x}) = [\log(\underline{x}), \log(\overline{x})]$, ...

Some Basic Functions

Images $\mathbf{x}^2$, $\sin(\mathbf{x})$, ..., are easily calculated, too.

$$\mathbf{x}^2 = \begin{cases} [\min(\underline{x}^2, \overline{x}^2), \max(\underline{x}^2, \overline{x}^2)] & \text{if } 0 \notin \mathbf{x}, \\ \mathbf{x}^2 = [0, \max(\underline{x}^2, \overline{x}^2)] & \text{otherwise} \end{cases}$$

But...

...what to do for more complex functions?

Images of Functions

Notice

$f(\mathbf{x})$ need not be an interval (neither closed nor connected).

Interval Hull $\square f(\mathbf{x})$

Compute the interval hull instead

$$\square f(\mathbf{x}) = \bigcap_{\mathbf{v} \in \mathbb{IR}^n \,:\, f(\mathbf{x}) \subseteq \mathbf{v}} \mathbf{v}.$$

Bad News

Computing $\square f(\mathbf{x})$ is still very difficult.

Interval Enclosure

Compute as tight as possible $\mathbf{v} \in \mathbb{IR}^n : f(\mathbf{x}) \subseteq \mathbf{v}$.

Interval Functions

Definition (Inclusion Isotonicity)

$\mathbf{f} : \mathbb{IR}^n \mapsto \mathbb{IR}$ is *inclusion isotonic* if for every $\mathbf{x}, \mathbf{y} \in \mathbb{IR}^n$:

$$\mathbf{x} \subseteq \mathbf{y} \Rightarrow \mathbf{f}(\mathbf{x}) \subseteq \mathbf{f}(\mathbf{y}).$$

Definition (Interval Extension)

$\mathbf{f} : \mathbb{IR}^n \mapsto \mathbb{IR}$ is *an interval extension* of $f : \mathbb{R}^n \mapsto \mathbb{R}$ if for every $x \in \mathbb{R}^n$:

$$f(x) = \mathbf{f}(x).$$

Theorem (Fundamental Theorem of Interval Analysis)

If $\mathbf{f} : \mathbb{IR}^n \mapsto \mathbb{IR}$ *satisfies both properties, then*

$$f(\mathbf{x}) \subseteq \mathbf{f}(\mathbf{x}), \quad \forall \mathbf{x} \in \mathbb{IR}^n.$$

Proof.

For every $x \in \mathbf{x}$, one has by interval extension and inclusion isotonicity that $f(x) = \mathbf{f}(x) \subseteq \mathbf{f}(\mathbf{x})$, whence $f(\mathbf{x}) \subseteq \mathbf{f}(\mathbf{x})$. □

Natural Interval Extension

Definition (Natural Interval Extension)

Let $f : \mathbb{R}^n \mapsto \mathbb{R}$ be a function given by an arithmetic expression. The corresponding *natural interval extension* $\mathbf{f}$ of f is defined by that expression when replacing real arithmetic by the interval one.

Theorem

Natural interval extension of an arithmetic expression is both an interval extension and inclusion isotonic.

Proof.

It is easy to see that interval arithmetic is both an interval extension and inclusion isotonic. Next, proceed by mathematical induction. $\square$

Natural Interval Extension

Example

$$f(x) = x^2 - x, \quad x \in \mathbf{x} = [-1, 2].$$

Then

$$\mathbf{x}^2 - \mathbf{x} = [-1, 2]^2 - [-1, 2] = [-2, 5],$$

$$\mathbf{x}(\mathbf{x} - 1) = [-1, 2]([-1, 2] - 1) = [-4, 2],$$

$$\text{Best one?} (\mathbf{x} - \tfrac{1}{2})^2 - \tfrac{1}{4} = ([-1, 2] - \tfrac{1}{2})^2 - \tfrac{1}{4} = [-\tfrac{1}{4}, 2].$$

Theorem

Suppose that in an expression of $f : \mathbb{R}^n \mapsto \mathbb{R}$ each variable $x_1, \ldots, x_n$ appears at most once. The corresponding natural interval extension $\mathbf{f}(\mathbf{x})$ satisfies for every $\mathbf{x} \in \mathbb{IR}^n$: $f(\mathbf{x}) = \mathbf{f}(\mathbf{x})$.

Proof.

Inclusion "$\subseteq$" by the previous theorems.
Inclusion "$\supseteq$" by induction and exactness of interval arithmetic. □

Software

Matlab libraries

- *Intlab* (by S.M. Rump),
  interval arithmetic and elementary functions
  http://www.ti3.tu-harburg.de/~rump/intlab/
- *Versoft* (by J. Rohn),
  verification software written in Intlab
  http://uivtx.cs.cas.cz/~rohn/matlab/
- *Lime* (by M. Hladík, J. Horáček et al.),
  interval methods written in Intlab, under development

Other languages libraries

- *Int4Sci Toolbox* (by Coprin team, INRIA),
  A Scilab Interface for Interval Analysis
  http://www-sop.inria.fr/coprin/logiciels/Int4Sci/
- *C++ libraries*: C-XSC, PROFIL/BIAS, BOOST interval, FILIB++,...
- *many others*: for Fortran, Pascal, Lisp, Maple, Mathematica,...

References – books

G. Alefeld and J. Herzberger.
Introduction to Interval Computations.
Academic Press, New York, 1983.

L. Jaulin, M. Kieffer, O. Didrit, and É. Walter.
Applied Interval Analysis.
Springer, London, 2001.

R. E. Moore.
Interval Analysis.
Prentice-Hall, Englewood Cliffs, NJ, 1966.

R. E. Moore, R. B. Kearfott, and M. J. Cloud.
Introduction to Interval Analysis.
SIAM, Philadelphia, PA, 2009.

A. Neumaier.
Interval Methods for Systems of Equations.
Cambridge University Press, Cambridge, 1990.

Next Section

Algorithmic Issues

Motivation

- Interval Analysis is not only an exciting theory, but is should be **useful in practice**
- **Useful in practice** = efficient algorithms
- Therefore, theory of algorithms plays an important role

In IntAnal, we meet many notions from Recursion Theory and Complexity Theory. For example:

- non-recursivity (= algorithmic unsolvability)
- NP-completeness, coNP-completeness
- weak and strong polynomiality
- Turing model and real-number computation model

AlgoIss: Nonrecursivity

In mathematics, there are many problems which are nonrecursive **= not algorithmically solvable** at all. Three examples:

Diophantine equations (Matiyasevich's Theorem, 1970; Hilbert's Tenth Problem, 1900)

- **Input:** a polynomial $p(x_1, \ldots, x_9)$ with integer coefficients.
- **Task:** decide whether there exist $x_1^*, \ldots, x_9^* \in \mathbb{Z}$ such that $p(x_1^*, \ldots, x_9^*) = 0$.

Provability (Gödel's Theorem, 1931)

- **Input:** a claim (= closed formula in the set-theoretic language) φ.
- **Task:** decide whether φ is provable in Set Theory (say, ZFC).

Randomness of a coin toss

- **Input:** a finite 0-1 sequence γ and a number K.
- **Task:** decide whether Kolmogorov complexity of γ is greater than K.

AlgoIss: Nonrecursivity (contd.)

The core **nonrecursive problem of Interval Analysis:**

- **Input:** a function $f : \mathbb{R}^n \to \mathbb{R}$, intervals $\mathbf{x}_1, \ldots, \mathbf{x}_n$ and $\xi \in \mathbb{R}$.
- **Task:** decide whether $\xi \in f(\mathbf{x}_1, \ldots, \mathbf{x}_n)$.

Negative results

- In general, we cannot determine the range of a function over intervals.
- In general, we cannot determine the interval hull $\Box f(\mathbf{x}_1, \ldots, \mathbf{x}_n)$.
- In general, we cannot determine an approximation of $\Box f(\mathbf{x}_1, \ldots, \mathbf{x}_n)$ with a prescribed precision.

Positive research motivation

- Find special classes of functions for which determination or approximation of $\Box f(\mathbf{x}_1, \ldots, \mathbf{x}_n)$ is algorithmically solvable.
- Find special classes of functions for which $\Box f(\mathbf{x}_1, \ldots, \mathbf{x}_n)$ is **efficiently** computable, i.e. in polynomial computation time.

AlgoIss: Nonrecursivity (contd.)

Proof-idea that "$\xi \in^? \text{range}(f)$" is nonrecursive

- By Matiyasevich we know that given a polynomial $p(x_1, \ldots, x_9)$, it is nonrecursive to decide whether p has a integer-valued root.
- Let $p(x_1, \ldots, x_9)$ be given and consider the function

$$f(x_1, \ldots, x_n) = p(x_1, \ldots, x_9)^2 + \sum_{i=1}^{9} \sin^2(\pi x_i).$$

- Now $0 \in \text{range}(f)$ iff $p(x_1, \ldots, x_9)$ has an integer-valued root.

- The proof showed an example of a **reduction** of one problem to another. This is **the** proof-method for hardness-of-computation results.
- We reasoned as follows: *if* somebody designed an algorithm for the question "$\xi \in^? \text{range}(f)$", *then* she would have solved the question "does p have an integer-valued root?". But the latter is impossible.

AlgoIss: An example of recursive, but "inefficient" problem

Problem formulation

Let $p(x_1, \dots, x_n)$ be a polynomial over given intervals $\mathbf{x}_1, \dots, \mathbf{x}_n$. By continuity we have

$$\overline{\mathrm{range}(p)} = \overline{\Box p} = \max\{p(x_1, \dots, x_n) : \; x_1 \in \mathbf{x}_1, \dots, x_n \in \mathbf{x}_n\},$$
$$\underline{\mathrm{range}(p)} = \underline{\Box p} = \min\{p(x_1, \dots, x_n) : \; x_1 \in \mathbf{x}_1, \dots, x_n \in \mathbf{x}_n\}.$$

Is computation of $\overline{\Box p}$, $\underline{\Box p}$ recursive?

- Yes.
- Proof-idea: use Tarski's Quantifier Elimination Method (completeness of the theory of Real Closed Fields).
- But: computation time can be *double-exponential*.
- So the problem is recursive, but *inefficient* for practical purposes.

AlgoIss: polynomiality, NP, coNP, hardness, completeness

Polynomiality

- **Consensus.** **Efficient algorithm** = algorithm running in time $p(L)$, where p is a polynomial and L is the bit-size of input.
- **Example.** Interval arithmetic runs in polynomial time.
- **Example.** Linear programming runs is polynomial time (e.g. Ellipsoid Method, IPMs; but **not** the Simplex Method!).
- **Remark.** In numerical problems, the bit-size L involves also *lengths of encodings of binary representations of rational numbers*.
  - Recall that this is a serious issue in linear programming: all known poly-time algorithms for LP are weakly polynomial.
  - So keep in mind: whenever we prove a polynomial-time result in Interval Analysis, which uses LP as a subroutine (which is a frequent case), we have a *weakly polynomial* result.
  - Weak polynomiality of LP is one of *Smale's Millenium Problems* for 21st century.

AlgoIss: polynomiality, NP, coNP, hardness, completeness

NP, coNP

- NP = a class of YES/NO problems s.t. a YES answer has a short and efficiently verifiable witness.
- coNP = a class of YES/NO problems s.t. a NO answer has a short and efficiently verifiable witness.

Examples

- CNFSAT: Is a given boolean formula in conjunctive normal form satisfiable? (NP)
- TAUT: Is a given boolean formula tautology? (coNP)
- TSP: Given a graph G with weighted edges and a number K, does G have a Hamiltonian cycle with length $\leq K$? (NP)
- KNAPSACK: Does a given equation $a^T x = b$ with $a \geq 0$ have a 0-1 solution? (NP)
- ILP: Does a given inequality system $Ax \leq b$ have an integer solution? (NP, nontrivial proof)

Reductions

- **Informally:** When every instance of a problem $\mathfrak{A}$ can be written as a particular instance of a problem $\mathfrak{B}$, then we say that $\mathfrak{A}$ is *reducible* to $\mathfrak{B}$. We write

$$\mathfrak{A} \leq \mathfrak{B}.$$

- **Example:** CNFSAT $\leq$ ILP. To illustrate, the CNFSAT-instance

$$(x_1 \vee \neg x_2 \vee \neg x_3) \quad \& \quad (\neg x_1 \vee x_4 \vee \neg x_5) \quad \& \quad (\neg x_1 \vee \neg x_2)$$

can be written as the ILP-instance

$$\begin{aligned} x_1 + (1 - x_2) + (1 - x_3) &\geq 1, \\ (1 - x_1) + x_4 + (1 - x_5) &\geq 1, \\ (1 - x_1) + (1 - x_2) &\geq 1, \\ x_i \in \{0, 1\} \quad (\forall i). \end{aligned}$$

Completeness

- $\mathfrak{B}$ is **NP-hard** $\Leftrightarrow (\forall \mathfrak{A} \in \mathrm{NP})\ \mathfrak{A} \leq \mathfrak{B}$,
- $\mathfrak{B}$ is **coNP-hard** $\Leftrightarrow (\forall \mathfrak{A} \in \mathrm{coNP})\ \mathfrak{A} \leq \mathfrak{B}$,
- $\mathfrak{B}$ is **NP-complete** $\Leftrightarrow \mathfrak{B} \in \mathrm{NP}$ and is NP-hard,
- $\mathfrak{B}$ is **coNP-complete** $\Leftrightarrow \mathfrak{B} \in \mathrm{coNP}$ and is coNP-hard.

Importance

- For (co)NP-hard (complete) problems we know only 2^n-algorithms or worse.
- Showing that a problem is (co)-NP hard (complete) is *bad news*: only small instances can be computed.
- Showing that a problem is (co)-NP hard (complete) is *good news for research*: inspect subproblems (special cases) which are tractable; or deal with approximate algorithms.

Generic problems

- Some well-known NP-complete problems: CNFSAT, ILP, TSP.
- Basic coNP-complete problem: TAUT.
- Following Jiří Rohn (our teacher, colleague and a celebrated personality in IntAnal), the following generic NP-complete problem is often used: *given a matrix A, decide whether there is $x \in \mathbb{R}^n$ s.t.*

$$|Ax| \leq e, \quad \|x\|_1 \geq 1.$$

- 2^n-algorithm: $\forall s \in \{-1,1\}^n$ set $T_s = \text{diag}(s)$ and solve the LP

$$-e \leq Ax \leq e, \quad e^T T_s x \geq 1.$$

- Rohn's NP-completeness result shows that this is "the best" algorithm we can expect.
- The 2^n-algorithm inspects $\mathbb{R}^n$ orthant-by-orthant; we will meet this **orthant decomposition method** repeatedly. (REMEMBER THIS!)

NP-hardness

- We use "NP-hardness" also for other than YES/NO problems.
- Then we say that a problem $\mathfrak{A}$ is NP-hard if the following holds: *if $\mathfrak{A}$ is solvable in polynomial time, then CNFSAT is solvable in polynomial time (and thus $P = NP$).*
- **Example:** given a polynomial $p(x_1, \ldots, x_n)$,
  - computation of $\overline{p(\mathbf{x}_1, \ldots, \mathbf{x}_n)}$ is NP-hard,
  - computation of $\underline{p(\mathbf{x}_1, \ldots, \mathbf{x}_n)}$ is NP-hard.

AlgoIss: Examples of complexity of computation of $\square f$

To recall: the basic problem of Interval Analysis is: *given a function f and intervals $\mathbf{x}_1, \ldots, \mathbf{x}_n$, determine $\square f(\mathbf{x}_1, \ldots, \mathbf{x}_n)$*. Examples from statistics:

Example: sample mean $f \equiv \mu := \frac{1}{n}\sum_{i=1}^{n} \mathbf{x}_i$

- Both $\overline{\square\mu}$ and $\underline{\square\mu}$ can be computed in polynomial time by interval arithmetic.

Example: sample variance $f \equiv \sigma^2 := \frac{1}{n}\sum_{i=1}^{n}(\mathbf{x}_i - \mu)^2$

- $\underline{\square\sigma^2}$: polynomial time.
- $\overline{\square\sigma^2}$: NP-hard, computable in time 2^n.
- *inapproximability result*: approximate computation of $\overline{\square\sigma^2}$ with an arbitrary absolute error: NP-hard.

Example: variation coefficient $f \equiv t := \frac{\mu}{\sigma}$

- $\underline{t}$: NP-hard, computable in time 2^n.
- *inapproximability result*: approximate computation of $\underline{t}$ with an arbitrary absolute error: NP-hard.
- $\overline{t}$: computable in polynomial time.

Interval linear equations, part I.

Interval Programming 2

Milan Hladík [1] Michal Černý [2]

[1] Faculty of Mathematics and Physics,
Charles University in Prague, Czech Republic
http://kam.mff.cuni.cz/~hladik/

[2] Faculty of Computer Science and Statistics,
University of Economics, Prague, Czech Republic
http://nb.vse.cz/~cernym/

Workshop on Interval Programming
7th International Conference of Iranian Operation Research Society
Semnan, Iran, May 12–13, 2014

Outline

1. Interval Linear Equations – Solution Concept
2. Enclosure Methods
3. Application: Verification of Real Linear Equations
4. Algorithmic Issues

Next Section

Solution Set

Interval Linear Equations

Let $\mathbf{A} \in \mathbb{IR}^{m \times n}$ and $\mathbf{b} \in \mathbb{IR}^{m}$. The family of systems

$$Ax = b, \quad A \in \mathbf{A}, \ b \in \mathbf{b}.$$

is called interval linear equations and abbreviated as $\mathbf{A}x = \mathbf{b}$.

Solution set

The solution set is defined

$$\Sigma := \{x \in \mathbb{R}^n : \exists A \in \mathbf{A} \exists b \in \mathbf{b} : Ax = b\}.$$

Important Notice

We do not want to compute $\mathbf{x} \in \mathbb{IR}^n$ such that $\mathbf{Ax} = \mathbf{b}$.

Theorem (Oettli–Prager, 1964)

The solution set Σ is a non-convex polyhedral set described by

$$|A^c x - b^c| \leq A^\Delta |x| + b^\Delta.$$

Proof of the Oettli–Prager Theorem

Let $x \in \Sigma$, that is, $Ax = b$ for some $A \in \mathbf{A}$ and $b \in \mathbf{b}$. Now,

$$|A^c x - b^c| = |(A^c - A)x + (Ax - b) + (b - b^c)| = |(A^c - A)x + (b - b^c)|$$
$$\leq |A^c - A||x| + |b - b^c| \leq A^\Delta |x| + b^\Delta.$$

Conversely, let $x \in \mathbb{R}^n$ satisfy the inequalities. Define $y \in [-1, 1]^m$ as

$$y_i = \begin{cases} \frac{(A^c x - b^c)_i}{(A^\Delta |x| + b^\Delta)_i} & \text{if } (A^\Delta |x| + b^\Delta)_i > 0, \\ 1 & \text{otherwise.} \end{cases}$$

Now, we have $(A^c x - b^c)_i = y_i (A^\Delta |x| + b^\Delta)_i$, or,

$$A^c x - b^c = \operatorname{diag}(y)(A^\Delta |x| + b^\Delta).$$

Define $z := \operatorname{sgn}(x)$, then $|x| = \operatorname{diag}(z)x$ and we can write

$$A^c x - b^c = \operatorname{diag}(y) A^\Delta \operatorname{diag}(z) x + \operatorname{diag}(y) b^\Delta,$$

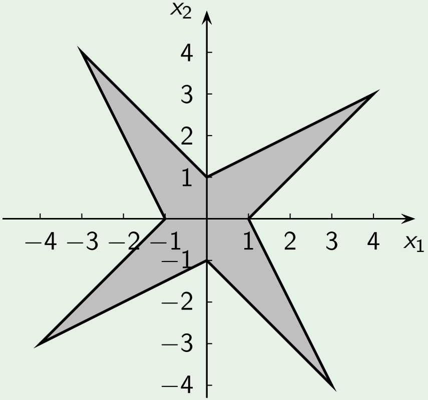
or

$$(A^c - \operatorname{diag}(y) A^\Delta \operatorname{diag}(z))x = b^c + \operatorname{diag}(y) b^\Delta. \quad \square$$

Example of the Solution Set

Example (Barth & Nuding, 1974))

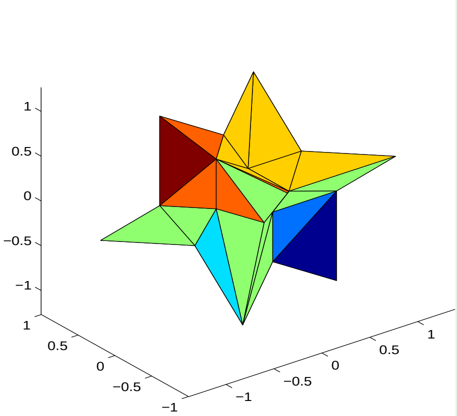
$$\begin{pmatrix} [2,4] & [-2,1] \\ [-1,2] & [2,4] \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} [-2,2] \\ [-2,2] \end{pmatrix}$$



Example of the Solution Set

Example

$$\begin{pmatrix} [3,5] & [1,3] & -[0,2] \\ -[0,2] & [3,5] & [0,2] \\ [0,2] & -[0,2] & [3,5] \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} [-1,1] \\ [-1,1] \\ [-1,1] \end{pmatrix}.$$



Topology of the Solution Set

Proposition

In each orthant, Σ is either empty or a convex polyhedral set.

Proof.

Restriction to the orthant given by $s \in \{\pm 1\}^n$:

$$|A^c x - b^c| \le A^\Delta |x| + b^\Delta, \ \operatorname{diag}(s)x \ge 0.$$

Since $|x| = \operatorname{diag}(s)x$, we have

$$|A^c x - b^c| \le A^\Delta \operatorname{diag}(s)x + b^\Delta, \ \operatorname{diag}(s)x \ge 0.$$

Using $|a| \le b \;\Leftrightarrow a \le b, \ -a \le b$, we get

$$(A^c - A^\Delta \operatorname{diag}(s))x \le \overline{b}, \ (-A^c - A^\Delta \operatorname{diag}(s))x \le -\underline{b}, \ \operatorname{diag}(s)x \ge 0. \quad \square$$

Corollary

The solutions of $\mathbf{A}x = \mathbf{b}$, $x \ge 0$ *is described by* $\underline{A}x \le \overline{b}, \ \overline{A}x \ge \underline{b}, \ x \ge 0.$

Interval Hull $\square\Sigma$

Goal

Seeing that Σ is complicated, compute $\square\Sigma$ instead.

First Idea

Go through all 2^n orthants of $\mathbb{R}^n$, determine interval hull of restricted sets (by solving $2n$ linear programs), and then put together.

Theorem

If $\mathbf{A}$ is regular (each $A \in \mathbf{A}$ is nonsingular), Σ is bounded and connected.

Theorem (Jansson, 1997)

When $\Sigma \neq \emptyset$, then exactly one of the following alternatives holds true:

1. *Σ is bounded and connected.*
2. *Each topologically connected component of Σ is unbounded.*

Second Idea – Jansson's Algorithm

Check the orthant with $(A^c)^{-1}b^c$ and then all the topologically connected.

Exercises for YOU

Prove or Disprove

1. $x \in \Sigma$ if and only if $0 \in \mathbf{A}x - \mathbf{b}$,
2. $x \in \Sigma$ if and only if $\mathbf{A}x \cap \mathbf{b} \neq \emptyset$.

Polynomial Cases

Two Basic Polynomial Cases

1. $A^c = I_n$,
2. $\mathbf{A}$ is inverse nonnegative, i.e., $A^{-1} \geq 0 \; \forall A \in \mathbf{A}$.

Theorem (Kuttler, 1971)

$\mathbf{A} \in \mathbb{IR}^{n\times n}$ is inverse nonnegative if and only if $\underline{A}^{-1} \geq 0$ and $\overline{A}^{-1} \geq 0$.

Theorem

Let $\mathbf{A} \in \mathbb{IR}^{n\times n}$ be inverse nonnegative. Then

1. $\square\Sigma = [\overline{A}^{-1}\underline{b}, \underline{A}^{-1}\overline{b}]$ *when* $\underline{b} \geq 0$,
2. $\square\Sigma = [\underline{A}^{-1}\underline{b}, \overline{A}^{-1}\overline{b}]$ *when* $\overline{b} \leq 0$,
3. $\square\Sigma = [\underline{A}^{-1}\underline{b}, \underline{A}^{-1}\overline{b}]$ *when* $0 \in \mathbf{b}$.

Proof.

1. Let $A \in \mathbf{A}$ and $b \in \mathbf{b}$. Since $\overline{b} \geq b \geq \underline{b} \geq 0$ and $\underline{A}^{-1} \geq A^{-1} \geq \overline{A}^{-1} \geq 0$, we get $\overline{A}^{-1}\underline{b} \leq A^{-1}b \leq \underline{A}^{-1}\overline{b}$. $\square$

Next Section

Preconditioning

Enclosure

Since Σ is hard to determine and deal with, we seek for enclosures

$$\mathbf{x} \in \mathbb{IR}^n \text{ such that } \Sigma \subseteq \mathbf{x}.$$

Many methods for enclosures exists, usually employ preconditioning.

Preconditioning (Hansen, 1965)

Let $C \in \mathbb{R}^{n \times n}$. The preconditioned system of equations:

$$(C\mathbf{A})x = C\mathbf{b}.$$

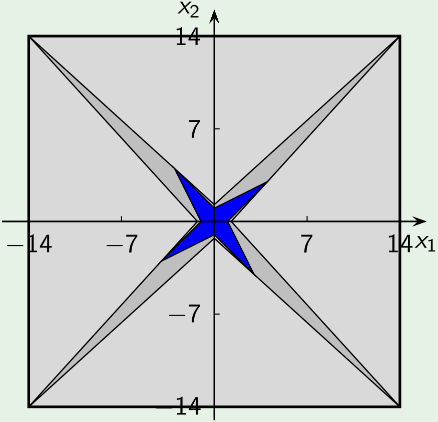
Remark

- the solution set of the preconditioned systems contains Σ
- usually, we use $C \approx (A^c)^{-1}$
- then we can compute the best enclosure (Hansen, 1992, Bliek, 1992, Rohn, 1993)

Preconditioning

Example (Barth & Nuding, 1974))

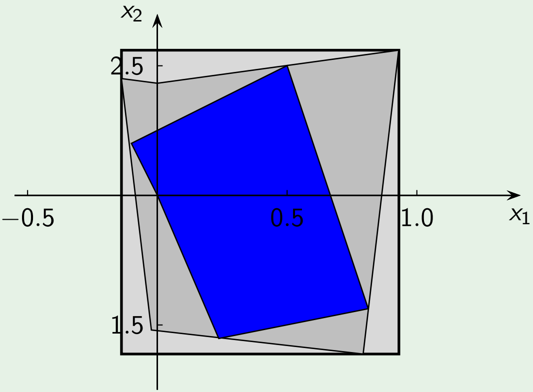
$$\begin{pmatrix} [2,4] & [-2,1] \\ [-1,2] & [2,4] \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} [-2,2] \\ [-2,2] \end{pmatrix}$$



Preconditioning

Example (typical case)

$$\begin{pmatrix} [6,7] & [2,3] \\ [1,2] & -[4,5] \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} [6,8] \\ -[7,9] \end{pmatrix}$$



Interval Gaussian Elimination

Interval Gaussian elimination = Gaussian elimination + interval arithmetic.

Example (Barth & Nuding, 1974))

$$\begin{pmatrix} [2,4] & [-2,1] \\ [-1,2] & [2,4] \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} [-2,2] \\ [-2,2] \end{pmatrix}$$

Then we proceed as follows

$$\begin{pmatrix} [2,4] & [-2,1] & [-2,2] \\ [-1,2] & [2,4] & [-2,2] \end{pmatrix} \sim \begin{pmatrix} [2,4] & [-2,1] & [-2,2] \\ 0 & [1,6] & [-4,4] \end{pmatrix}.$$

By back substitution, we compute

$$\begin{aligned} \mathbf{x}_2 &= [-4,4], \\ \mathbf{x}_1 &= \big([-2,2] - [-2,1] \cdot [-4,4]\big) \,/\, [2,4] = [-5,5]. \end{aligned}$$

Interval Jacobi and Gauss-Seidel Iterations

Idea

From the ith equation of $Ax = b$ we get

$$x_i = \frac{1}{a_{ii}} \left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j - \sum_{j=i+1}^{n} a_{ij} x_j \right).$$

If $\mathbf{x}^0 \supseteq \Sigma$ is an initial enclosure, then

$$x_i \in \frac{1}{\mathbf{a}_{ii}} \left(\mathbf{b}_i - \sum_{j \neq i} \mathbf{a}_{ij} \mathbf{x}_j^0 \right), \quad \forall x \in \Sigma.$$

Thus, we can tighten the enclosure by iterations

Interval Jacobi / Gauss–Seidel Iterations ($k = 1, 2, \dots$)

1: **for** $i = 1, \dots, n$ **do**

2: $\quad \mathbf{x}_i^k := \frac{1}{\mathbf{a}_{ii}} \left(\mathbf{b}_i - \sum_{j \neq i} \mathbf{a}_{ij} \mathbf{x}_j^{k-1} \right) \cap \mathbf{x}_i^{k-1};$

3: **end for**

Krawczyk Iterations

Krawczyk operator

Krawczyk operator $K : \mathbb{IR}^n \to \mathbb{IR}^n$ reads

$$K(\mathbf{x}) := C\mathbf{b} + (I_n - C\mathbf{A})\mathbf{x}$$

Proposition

If $x \in \mathbf{x} \cap \Sigma$, then $x \in K(\mathbf{x})$.

Proof.

Let $x \in \mathbf{x} \cap \Sigma$, so $Ax = b$ for some $A \in \mathbf{A}$ and $b \in \mathbf{b}$. Thus $CAx = Cb$, whence $x = Cb + (I_n - CA)x \in C\mathbf{b} + (I_n - C\mathbf{A})\mathbf{x} = K(\mathbf{x})$. □

Krawczyk Iterations

Let $\mathbf{x}^0 \supseteq \Sigma$ is an initial enclosure, and iterate $(k = 1, 2, \dots)$:

1: $\mathbf{x}^k := K(\mathbf{x}^{k-1}) \cap \mathbf{x}^{k-1}$;

ε-inflation

Theorem

Let $\mathbf{x} \in \mathbb{IR}^n$ *and* $C \in \mathbb{R}^{n\times n}$. *If*

$$K(\mathbf{x}) = C\mathbf{b} + (I - C\mathbf{A})\mathbf{x} \subseteq \operatorname{int}\mathbf{x},$$

then C *is nonsingular,* $\mathbf{A}$ *is regular, and* $\Sigma \subseteq \mathbf{x}$.

Proof.

Existence of a solution based on Brouwer's fixed-point theorem.
Nonsingularity and uniqueness based on the Perron–Frobenius theory. □

Remark

- A reverse iteration method to the Krawczyk method.
- It starts with a small box around $(A^c)^{-1}b^c$, and then interatively inflates the box.
- Implemented in Intlab v. 6.

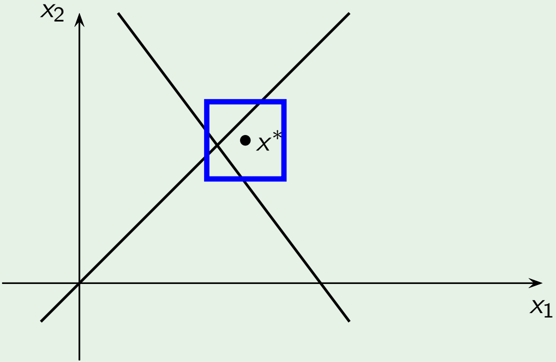
Next Section

Verification of Real Linear Equations

Problem formulation

Given a real system $Ax = b$ and x^* approximate solution, find $\mathbf{y} \in \mathbb{IR}^n$ such that $A^{-1}b \in x^* + \mathbf{y}$.

Example



Verification of Real Linear Equations

Theorem

Let $\mathbf{y} \in \mathbb{IR}^n$ *and* $C \in \mathbb{R}^{n\times n}$*. If*

$$C(b - Ax^*) + (I - CA)\mathbf{y} \subseteq int\,\mathbf{y},$$

then C *and* A *are nonsingular, and* $A^{-1}b \in x^* + \mathbf{y}$.

Proof.

Substitute $x := y + x^*$, and apply the ε-inflation method for the system

$$Ay = b - Ax^*.$$

□

ε-inflation method (Caprani and Madsen, 1978, Rump, 1980)

Repeat inflating $\mathbf{y} := [0.9, 1.1]\mathbf{x} + 10^{-20}[-1, 1]$ and updating

$$\mathbf{x} := C(b - Ax^*) + (I - CA)\mathbf{y}$$

until $\mathbf{x} \subseteq \operatorname{int}\mathbf{y}$.

Then, $\Sigma \subseteq x^* + \mathbf{x}$.

Verification of Real Linear Equations

Example

Let A be the Hilbert matrix of size 10 (i.e., $a_{ij} = \frac{1}{i+j-1}$), and $b := Ae$.

Then $Ax = b$ has the solution $x = e = (1, \ldots, 1)^T$.

Approximate solution by Matlab:	Enclosing interval by ε-inflation method (2 iterations):
0.999999999235452	[0.99999973843401, 1.00000026238575]
1.000000065575364	[0.99999843048508, 1.00000149895660]
0.999998607887449	[0.99997745481481, 1.00002404324710]
1.000012638750021	[0.99978166603900, 1.00020478046370]
0.999939734980300	[0.99902374408278, 1.00104070076742]
1.000165704992114	[0.99714060702796, 1.00268292103727]
0.999727989024899	[0.99559932282378, 1.00468935360003]
1.000263042205847	[0.99546972629357, 1.00425202249136]
0.999861803020249	[0.99776781605377, 1.00237789028988]
1.000030414871015	[0.99947719419921, 1.00049082925529]

References

G. Alefeld and J. Herzberger.
Introduction to Interval Computations.
Academic Press, New York, 1983.

M. Fiedler, J. Nedoma, J. Ramík, J. Rohn, and K. Zimmermann.
Linear optimization problems with inexact data.
Springer, New York, 2006.

R. E. Moore, R. B. Kearfott, and M. J. Cloud.
Introduction to interval analysis.
SIAM, Philadelphia, PA, 2009.

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

J. Rohn.
A handbook of results on interval linear problems.
Tech. Rep. 1163, Acad. of Sci. of the Czech Republic, Prague, 2012.
`http://uivtx.cs.cas.cz/~rohn/publist/!aahandbook.pdf`

S. M. Rump.
Verification methods: Rigorous results using floating-point arithmetic.
Acta Numer., 19:287–449, 2010.

Next Section

Algorithmic Issues: Solvability of $\mathbf{A}x = \mathbf{b}$

To recall:

- System $\mathbf{A}x = \mathbf{b}$ is *solvable* iff $(\exists A \in \mathbf{A})(\exists b \in \mathbf{b})(\exists x \in \mathbb{R}^n)\ Ax = b$.
- *Solution set* is defined by

$$\Sigma(\mathbf{A}, \mathbf{b}) = \bigcup_{A \in \mathbf{A},\ b \in \mathbf{b}} \{x \in \mathbb{R}^n : \ Ax = b\}.$$

Theorem

Checking solvability is an NP-complete problem.

Outline: we must prove (1) NP-hardness and (2) presence in NP.

Proof

Step 1. Proof of NP-hardness. We will show that Rohn's generic problem of solvability of $-e \le Ax \le e,\ \|x\|_1 \ge 1$ is reducible to checking solvability of a particular system $\mathbf{A}x = \mathbf{b}$.
(Informally: if somebody manages to design an efficient method for checking solvability, then she also managed to solve the Rohn's generic problem; but it is impossible unless P = NP.)

AlgoIss: Solvability of $\mathbf{A}x = \mathbf{b}$ (contd.)

Proof of NP-completeness of $\Sigma(\mathbf{A}, \mathbf{b}) \neq^? \emptyset$. Step 1 continued

Claim: Rohn's system $-e \le Ax \le e, \|x\|_1 \ge 1$ is solvable iff

$$[A, A]x = [-e, e], \quad [-e^T, e^T]x = [1, 1] \tag{1}$$

is solvable. Thus, if we have an efficient method for (1), then we have an efficient method for Rohn's system, which is NP-complete. This proves NP-hardness.

Proof of claim

- If x solves Rohn's system, then $x' := \frac{x}{\|x\|_1}$ solves (1). [*Proof.* $|Ax'| = \frac{1}{\|x\|_1}|Ax| \le |Ax| \le e$; thus x' solves $Ax' = [-e, e]$. In addition, $\|x'\|_1 = 1$; thus $\mathrm{sgn}(x')^T x' = 1$ and $\mathrm{sgn}(x') \in [-e, e]$.]
- If x solves $Ax = b, c^T x = 1$ with $b \in [-e, e]$ and $c \in [-e, e]$, then $|Ax| = |b| \le e$ and $\|x\|_1 = e^T|x| \ge |c|^T|x| \ge c^T x = 1$. QED

Proof of NP-completeness of $\Sigma(\mathbf{A}, \mathbf{b}) \neq^? \emptyset$. Step 2

Step 2. Proof that the problem is in NP.

- Intuitively, any pair (A_0, b_0) s.t.

$$A_0 \in \mathbf{A}, \quad b_0 \in \mathbf{b}, \quad \{x : A_0 x = b_0\} \neq \emptyset \tag{2}$$

  could serve as an NP-witness. (Observe that the conditions (2) can be verified in polynomial time.)
- However, there is a technical problem: *the NP-witness must have polynomial size*. In other words: we must prove that there exists a polynomial p s.t. $\text{bitsize}(A_0, b_0) \leq p(\text{bitsize}(\underline{A}, \overline{A}, \underline{b}, \overline{b}))$.
- We proceed otherwise.

Algolss: Solvability of $\mathbf{A}x = \mathbf{b}$ (contd.)

Proof of NP-completeness of $\Sigma(\mathbf{A}, \mathbf{b}) \neq^? \emptyset$. Step 2 contd.

- We use Oettli-Prager: we know that

$$\begin{aligned}&\Sigma(\mathbf{A}, \mathbf{b}) \cap \mathbb{R}^n_s \\ &= \{x \in \mathbb{R}^n : \underbrace{-A^\Delta T_s x - b^\Delta \leq A^c x - b^c \leq A^\Delta T_s x + b^\Delta,\ T_s x \geq 0}_{(\star)}\},\end{aligned}$$

  where $s \in \{-1, 1\}^n$, $T_s = \mathrm{diag}(s)$ and $\mathbb{R}^n_s = \{x \in \mathbb{R}^n : T_s x \geq 0\}$.
- Given s, nonemptiness of the polyhedron $(\star)$ can be checked in polynomial time by LP.
- Clearly, bitsize of s is bounded by bitsize$(\underline{A}, \overline{A}, \underline{b}, \overline{b})$.
- Thus, s s.t. $\Sigma \cap \mathbb{R}^n_s \neq \emptyset$ is a valid NP-witness for the fact $\Sigma \neq \emptyset$. $\square$

Algolss: Boundedness of the solution set

To recall: the solution set is defined as

$$\Sigma = \bigcup_{A \in \mathbf{A},\ b \in \mathbf{b}} \{x \in \mathbb{R}^n : \ Ax = b\}.$$

Theorem

Deciding whether Σ is bounded is a coNP-complete problem.

Proof idea of coNP-hardness.

Consider the system $\mathbf{A}x = 0$. Then $\Sigma = \Sigma(\mathbf{A}, [0, 0])$ is unbounded iff

$$\text{there is } A \in \mathbf{A} \text{ which is singular.} \tag{3}$$

Later we will show that deciding (3) is an NP-complete problem. So checking unboundedness of Σ is NP-hard, and checking boundedness is coNP-hard. □

Algolss: Computation of $\square\Sigma$

Some consequences

- Every *exact* enclosure method (i.e. every method computing $\underline{\square\Sigma}$ and $\overline{\square\Sigma}$ exactly) must be implicitly able to detect (at least) the following "extreme" cases:
  - $\Sigma = \emptyset$,
  - Σ is unbounded.
- Thus, any enclosure method *must* be able to solve two NP-complete problems. Thus $\square\Sigma$ is NP-hard.
- So we cannot expect that the 2^n-method, based on orthant decomposition by Oettli-Prager, could be significantly improved.

Further results

- The basic results on hardness-of-computation of $\square\Sigma$ can be pushed further: it holds that even *approximate* computation of $\square\Sigma$ with a given absolute error or relative error is NP-hard.
- So, in theory, even "not too redundant" enclosures are hard to compute.

Interval linear equations, part II.

Interval Programming 3

Milan Hladík[1] Michal Černý[2]

[1] Faculty of Mathematics and Physics,
Charles University in Prague, Czech Republic
http://kam.mff.cuni.cz/~hladik/

[2] Faculty of Computer Science and Statistics,
University of Economics, Prague, Czech Republic
http://nb.vse.cz/~cernym/

Workshop on Interval Programming
7th International Conference of Iranian Operation Research Society
Semnan, Iran, May 12–13, 2014

Outline

Next Section

1. Regularity of Interval Matrices

2. Parametric Interval Systems

3. AE Solution Set

4. Algorithmic Issues

Regularity

Definition (Regularity)

$\mathbf{A} \in \mathbb{IR}^{n \times n}$ is regular if each $A \in \mathbf{A}$ is nonsingular.

Theorem

Checking regularity of an interval matrix is co-NP-hard.

Forty necessary and sufficient conditions for regularity of $\mathbf{A}$ by Rohn (2010):

1. The system $|A^c x| \leq A^\Delta |x|$ has the only solution $x = 0$.
2. $\det(A^c - \operatorname{diag}(y) A^\Delta \operatorname{diag}(z))$ is constantly either positive or negative for each $y, z \in \{\pm 1\}^n$.
3. For each $y \in \{\pm 1\}^n$, the system $A^c x - \operatorname{diag}(y) A^\Delta |x| = y$ has a solution.
4. ...

Regularity – Sufficient / Necessary Conditions

Theorem (Beeck, 1975)

If $\rho(|(A^c)^{-1}|A^\Delta) < 1$, then $\mathbf{A}$ *is regular.*

Proof.

Precondition $\mathbf{A}$ by the midpoint inverse: $\mathbf{M} := (A^c)^{-1}\mathbf{A}$. Now,

$$M^c = I_n, \quad M^\Delta = |(A^c)^{-1}|A^\Delta,$$

and for each $M \in \mathbf{M}$ we have

$$|M - M^c| = |M - I_n| \leq M^\Delta.$$

From the theory of eigenvalues of nonnegative matrices it follows

$$\rho(M - I_n) \leq \rho(M^\Delta) < 1,$$

so M has no zero eigenvalue and is nonsingular. □

Necessary Condition

If $0 \in \mathbf{A}x$ for some $0 \neq x \in \mathbb{R}^n$, then $\mathbf{A}$ is not regular. (Try $x := (A^c)^{-1}_{*i}$)

Exercises for YOU

The following conditions are necessary for the regularity of $\mathbf{A}$. Decide which of them are sufficient as well:

1. all matrices A of the form $a_{ij} \in \{\underline{a}_{ij}, \overline{a}_{ij}\}$ are nonsingular,
2. all matrices A of the form $a_{ij} \in \{\underline{a}_{ij}, \overline{a}_{ij}\}$, and A^c are nonsingular.

Next Section

Parametric Interval Systems

Parametric Interval Systems

$$A(p)x = b(p),$$

where the entries of $A(p)$ and $b(p)$ depend on parameters $p_1 \in \mathbf{p}_1, \ldots, p_K \in \mathbf{p}_K$.

Definition (Solution Set)

$$\Sigma_{\mathrm{p}} = \{x \in \mathbb{R}^n \colon A(p)x = b(p) \text{ for some } p \in \mathbf{p}\}.$$

Relaxation

Compute (enclosures of) the ranges $\mathbf{A} := A(\mathbf{p})$ and $\mathbf{b} := b(\mathbf{p})$ and solve

$$\mathbf{A}x = \mathbf{b}.$$

May overestimate a lot!

Special Case: Parametric Linear Interval Systems

Parametric Linear Interval Systems

$$A(p)x = b(p),$$

where

$$A(p) = \sum_{k=1}^{K} A_k p_k, \quad b(p) = \sum_{k=1}^{K} b_k p_k$$

and $p \in \mathbf{p}$ for some given interval vector $\mathbf{p} \in \mathbb{IR}^K$, matrices $A_1, \ldots, A_K \in \mathbb{R}^{n \times n}$ and vectors $b_1, \ldots, b_n \in \mathbb{R}^n$.

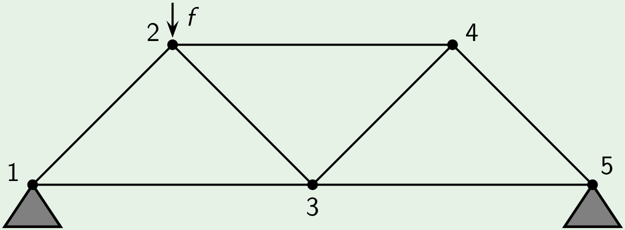
Remark

It covers many structured matrices: symmetric, skew-symmetric, Toeplitz or Hankel.

Parametric Linear Interval Systems – Example

Example (Displacements of a truss structure (Skalna, 2006))

The 7-bar truss structure subject to downward force.
The stiffnesses s_{ij} of bars are uncertain.
The displacements d of the nodes, are solutions of the system $Kd = f$, where f is the vector of forces.



Parametric Linear Interval Systems – Example

Example (Displacements of a truss structure (Skalna, 2006))

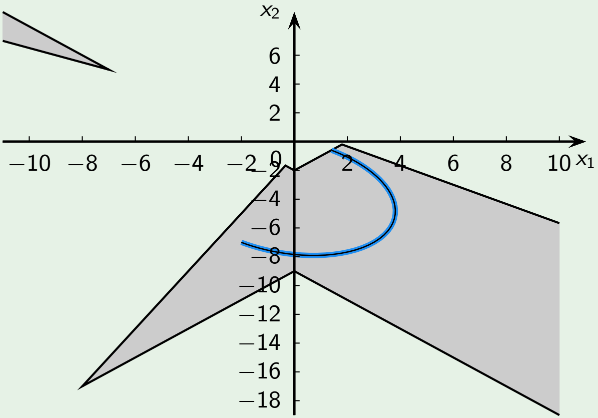
The 7-bar truss structure subject to downward force.
The stiffnesses s_{ij} of bars are uncertain.
The displacements d of the nodes, are solutions of the system $Kd = f$, where f is the vector of forces.

$$K = \begin{pmatrix}
\frac{s_{12}}{2} + s_{13} & -\frac{s_{12}}{2} & -\frac{s_{12}}{2} & -s_{13} & 0 & 0 & 0 \\
-\frac{s_{21}}{2} & \frac{s_{21}+s_{23}}{2} + s_{24} & \frac{s_{21}-s_{23}}{2} & -\frac{s_{23}}{2} & \frac{s_{23}}{2} & -s_{24} & 0 \\
-\frac{s_{21}}{2} & \frac{s_{21}-s_{23}}{2} & \frac{s_{21}+s_{23}}{2} & \frac{s_{23}}{2} & -\frac{s_{23}}{2} & 0 & 0 \\
-s_{31} & -\frac{s_{32}}{2} & \frac{s_{32}}{2} & s_{31} + \frac{s_{32}+s_{34}}{2} + s_{35} & \frac{s_{34}-s_{32}}{2} & -\frac{s_{34}}{2} & -\frac{s_{34}}{2} \\
0 & \frac{s_{32}}{2} & -\frac{s_{32}}{2} & \frac{s_{34}-s_{32}}{2} & \frac{s_{34}+s_{32}}{2} & -\frac{s_{34}}{2} & -\frac{s_{34}}{2} \\
0 & -s_{42} & 0 & -\frac{s_{43}}{2} & -\frac{s_{43}}{2} & s_{42} + \frac{s_{43}+s_{45}}{2} & 0 \\
0 & 0 & 0 & -\frac{s_{43}}{2} & -\frac{s_{43}}{2} & 0 & \frac{s_{43}+s_{45}}{2}
\end{pmatrix}$$

Parametric Linear Interval Systems – Example

Example

$$\begin{pmatrix} 1-2p & 1 \\ 2 & 4p-1 \end{pmatrix} x = \begin{pmatrix} 7p-9 \\ 3-2p \end{pmatrix}, \quad p \in \mathbf{p} = [0,1].$$



Parametric Linear Interval Systems – Solution Set

Theorem

If $x \in \Sigma_{\mathrm{p}}$, then it solves

$$|A(p^c)x - b(p^c)| \leq \sum_{k=1}^{K} p_k^{\Delta} |A^k x - b^k|.$$

Proof.

$$|A(p^c)x - b(p^c)| = \left| \sum_{k=1}^{K} p_k^c (A^k x - b^k) \right| = \left| \sum_{k=1}^{K} p_k^c (A^k x - b^k) - \sum_{k=1}^{K} p_k (A^k x - b^k) \right|$$

$$= \left| \sum_{k=1}^{K} (p_k^c - p_k)(A^k x - b^k) \right| \leq \sum_{k=1}^{K} |p_k^c - p_k| |A^k x - b^k| \leq \sum_{k=1}^{K} p_k^{\Delta} |A^k x - b^k|. \quad \square$$

- Popova (2009) showed that it is the complete characterization of Σ_{p} as long as no interval parameter appears in more than one equation.
- Checking $x \in \Sigma_{\mathrm{p}}$ for a given $x \in \mathbb{R}^n$ is a polynomial problem via linear programming.

Parametric Linear Interval Systems – Enclosures

Relaxation and Preconditioning – First Idea

Evaluate $\mathbf{A} := A(\mathbf{p})$, $\mathbf{b} := b(\mathbf{p})$, choose $C \in \mathbb{R}^{n\times n}$ and solve

$$(C\mathbf{A})x = C\mathbf{b}.$$

Relaxation and Preconditioning – Second Idea

Solve $\mathbf{A}'x = \mathbf{b}'$, where

$$\mathbf{A}' := \sum_{k=1}^{K} (CA^k)\mathbf{p}_k, \quad \mathbf{b}' := \sum_{k=1}^{K} (Cb^k)\mathbf{p}_k.$$

Second Idea is Provably Better

Due to sub-distributivity law,

$$\mathbf{A}' := \sum_{k=1}^{K} (CA^k)\mathbf{p}_k \subseteq C\left(\sum_{k=1}^{K} A^k\mathbf{p}_k\right) = (C\mathbf{A}).$$

Special Case: Symmetric Systems

The Symmetric Solution Set of $\mathbf{A}x = \mathbf{b}$

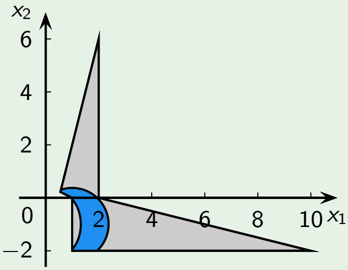
$$\{x \in \mathbb{R}^n : Ax = b \text{ for some symmetric } A \in \mathbf{A} \text{ and } b \in \mathbf{b}\}.$$

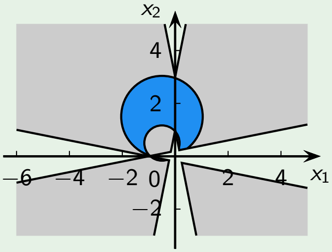
Described by $\frac{1}{2}(4^n - 3^n - 2 \cdot 2^n + 3) + n$ nonlinear inequalities (H., 2008).

Example

$$\mathbf{A} = \begin{pmatrix} [1,2] & [0,a] \\ [0,a] & -1 \end{pmatrix}, \ \mathbf{b} = \begin{pmatrix} 2 \\ 2 \end{pmatrix}.$$

$$\mathbf{A} = \begin{pmatrix} -1 & [-5,5] \\ [-5,5] & 1 \end{pmatrix}, \ \mathbf{b} = \begin{pmatrix} 1 \\ [1,3] \end{pmatrix}.$$





Application: Least Square Solutions

Least Square Solution

Let $A \in \mathbb{IR}^{m \times n}$, $b \in \mathbb{IR}^m$ and $m > n$. The least square solution of

$$Ax = b,$$

is defined as the optimal solution of

$$\min_{x \in \mathbb{R}^n} \|Ax - b\|_2,$$

or, alternatively as the solution to

$$A^T Ax = A^T b.$$

Interval Least Square Solution Set

Let $\mathbf{A} \in \mathbb{IR}^{m \times n}$ and $\mathbf{b} \in \mathbb{IR}^m$ and $m > n$. The LSQ solution set is defined

$$\Sigma_{LSQ} := \{x \in \mathbb{R}^n : \exists A \in \mathbf{A}\, \exists b \in \mathbf{b} : A^T Ax = A^T b\}.$$

Proposition

Σ_{LSQ} is contained in the solution set to $\mathbf{A}^T \mathbf{A} x = \mathbf{A}^T \mathbf{b}$.

Application: Least Square Solutions

Proposition

Σ_{LSQ} is contained in the solution set to

$$\begin{pmatrix} 0 & \mathbf{A}^T \\ \mathbf{A} & I_m \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 0 \\ \mathbf{b} \end{pmatrix}. \tag{1}$$

Proof.

Let $A \in \mathbf{A}$, $b \in \mathbf{b}$. If x, y solve

$$A^T y = 0,\ \ Ax + y = b,$$

then

$$0 = A^T(b - Ax) = A^T b - A^T Ax,$$

and vice versa. □

Proposition

Relaxing the dependencies, the solution set to $\mathbf{A}^T\mathbf{A}x = \mathbf{A}^T\mathbf{b}$ is contained in the solution set to (1).

Next Section

Tolerable Solutions

Motivation

So far, existentially quantified interval systems

$$\Sigma := \{x \in \mathbb{R}^n : \exists A \in \mathbf{A}\, \exists b \in \mathbf{b} : Ax = b\}.$$

Now, incorporate universal quantification as well!

Definition (Tolerable Solutions)

A vector $x \in \mathbb{R}^n$ is a tolerable solution to $\mathbf{A}x = \mathbf{b}$ if for each $A \in \mathbf{A}$ there is $b \in \mathbf{b}$ such that $Ax = b$.

In other words,

$$\forall A \in \mathbf{A}\, \exists b \in \mathbf{b} : Ax = b.$$

Equivalent Characterizations

- $\mathbf{A}x \subseteq \mathbf{b}$,
- $|A^c x - b^c| \leq -A^\Delta |x| + b^\Delta$.

Tolerable Solutions

Theorem (Rohn, 1986)

A vector $x \in \mathbb{R}^n$ is a tolerable solution if and only if $x = x_1 - x_2$, where

$$\overline{A}x_1 - \underline{A}x_2 \leq \overline{b},\ \ \underline{A}x_1 - \overline{A}x_2 \geq \underline{b},\ \ x_1, x_2 \geq 0.$$

Proof.

"$\Leftarrow$" Let $A \in \mathbf{A}$. Then

$$Ax = Ax_1 - Ax_2 \leq \overline{A}x_1 - \underline{A}x_2 \leq \overline{b},$$
$$Ax = Ax_1 - Ax_2 \geq \underline{A}x_1 - \overline{A}x_2 \geq \underline{b}$$

Thus, $Ax \in \mathbf{b}$ and $Ax = b$ for some $b \in \mathbf{b}$.
"$\Rightarrow$" Let $x \in \mathbb{R}^n$ be a tolerable solution. Define $x_1 := \max\{x, 0\}$ and $x_2 := \max\{-x, 0\}$ the positive and negative part of x, respectively. Then $x = x_1 - x_2$, $|x| = x_1 + x_2$, and $|A^c x - b^c| \leq -A^\Delta |x| + b^\Delta$ draws

$$A^c(x_1 - x_2) - b^c \leq -A^\Delta(x_1 + x_2) + b^\Delta,$$
$$-A^c(x_1 - x_2) + b^c \leq -A^\Delta(x_1 + x_2) + b^\Delta.$$

$\square$

Tolerable Solutions – Application

Example (Leontief's Input–Output Model of Economics)

- economy with n sectors (e.g., agriculture, industry, transportation, etc.),
- sector i produces a single commodity of amount x_i,
- production of each unit of the jth commodity will require a_{ij} (amount) of the ith commodity
- d_i the final demand in sector i.

Now the model draws

$$x_i = a_{i1}x_1 + \cdots + a_{in}x_n + d_i.$$

or, in a matrix form

$$x = Ax + d.$$

The solution $x = (I_n - A)^{-1}d = \sum_{k=0}^{\infty} A^k d$ is nonnegative if $\rho(A) < 1$.

Question: Exists x such that for any $A \in \mathbf{A}$ there is $d \in \mathbf{d}$: $(I_n - A)x = d$?

AE Solutions

Quantified system $\mathbf{A}x = \mathbf{b}$

- each interval parameter $\mathbf{a}_{ij}$ and $\mathbf{b}_i$ is quantified by $\forall$ or $\exists$
- the universally quantified parameters are denoted by $\mathbf{A}^\forall$, $\mathbf{b}^\forall$,
- the existentially quantified parameters are denoted by $\mathbf{A}^\exists$, $\mathbf{b}^\exists$
- the system reads $(\mathbf{A}^\forall + \mathbf{A}^\exists)x = \mathbf{b}^\forall + \mathbf{b}^\exists$

Definition (AE Solution Set)

$$\Sigma_{AE} := \Big\{x \in \mathbb{R}^n : \\ \forall A^\forall \in \mathbf{A}^\forall \, \forall b^\forall \in \mathbf{b}^\forall \, \exists A^\exists \in \mathbf{A}^\exists \, \exists b^\exists \in \mathbf{b}^\exists : (A^\forall + A^\exists)x = b^\forall + b^\exists\Big\}.$$

AE Solutions

Theorem (Shary, 1995)

$$\Sigma_{AE} = \left\{x \in \mathbb{R}^n : \mathbf{A}^{\forall} x - \mathbf{b}^{\forall} \subseteq \mathbf{b}^{\exists} - \mathbf{A}^{\exists} x\right\}. \qquad (2)$$

Proof.

$$\begin{aligned}\Sigma_{AE} &= \left\{x \in \mathbb{R}^n : \forall A^{\forall} \in \mathbf{A}^{\forall}\, \forall b^{\forall} \in \mathbf{b}^{\forall}\, \exists A^{\exists} \in \mathbf{A}^{\exists}\, \exists b^{\exists} \in \mathbf{b}^{\exists} : A^{\forall} x - b^{\forall} = b^{\exists} - A^{\exists} x\right\} \\ &= \left\{x \in \mathbb{R}^n : \forall A^{\forall} \in \mathbf{A}^{\forall}\, \forall b^{\forall} \in \mathbf{b}^{\forall} : A^{\forall} x - b^{\forall} \in \mathbf{b}^{\exists} - \mathbf{A}^{\exists} x\right\} \\ &= \left\{x \in \mathbb{R}^n : \mathbf{A}^{\forall} x - \mathbf{b}^{\forall} \subseteq \mathbf{b}^{\exists} - \mathbf{A}^{\exists} x\right\}.\end{aligned}$$

□

Theorem (Rohn, 1996)

$$\Sigma_{AE} = \left\{x \in \mathbb{R}^n : |A^c x - b^c| \leq \left((\mathbf{A}^{\exists})^{\Delta} - (\mathbf{A}^{\forall})^{\Delta}\right)|x| + (\mathbf{b}^{\exists})^{\Delta} - (\mathbf{b}^{\forall})^{\Delta}\right\}.$$

Proof.

Using (2) and the fact $\mathbf{p} \subseteq \mathbf{q} \;\Leftrightarrow\; |p^c - q^c| \leq p^{\Delta} - q^{\Delta}$, we get

$$\begin{aligned}\left|(\mathbf{A}^{\forall} x - \mathbf{b}^{\forall})^c - (\mathbf{b}^{\exists} - \mathbf{A}^{\exists} x)^c\right| &\leq \left(\mathbf{A}^{\exists} x - \mathbf{b}^{\exists}\right)^{\Delta} - \left(\mathbf{b}^{\forall} - \mathbf{A}^{\forall} x\right)^{\Delta} \\ &= (\mathbf{A}^{\exists})^{\Delta}|x| + \mathbf{b}^{\exists\Delta} - (\mathbf{A}^{\forall})^{\Delta} x| - \mathbf{b}^{\forall\Delta}.\end{aligned}$$

□

Strong solution of $\mathbf{A}x = \mathbf{b}$

Characterize when $x \in \mathbb{R}^n$ solves $Ax = b$ for every $A \in \mathbf{A}$ and $b \in \mathbf{b}$.

Software

webComputing (by E. Popova)

- interactive free visualization at
  http://cose.math.bas.bg/webComputing/
- parametric solution set
- AE solution set
- 3D standard solution set

Parametric interval systems

- Mathematica package (Popova, 2004)
- C++ library C-XCS implementation (Popova and Krämer, 2007; Zimmer, Krämer and Popova, 2012)

References

M. Fiedler, J. Nedoma, J. Ramík, J. Rohn, and K. Zimmermann.
Linear optimization problems with inexact data.
Springer, New York, 2006.

M. Hladík.
Enclosures for the solution set of parametric interval linear systems.
Int. J. Appl. Math. Comput. Sci., 22(3):561–574, 2012.

E. D. Popova.
Explicit description of AE solution sets for parametric linear systems.
SIAM J. Matrix Anal. Appl., 33(4):1172–1189, 2012.

J. Rohn.
Forty necessary and sufficient conditions for regularity of interval matrices: A survey.
Electron. J. Linear Algebra, 18:500–512, 2009.

S. M. Rump.
Verification methods: Rigorous results using floating-point arithmetic.
Acta Numer., 19:287–449, 2010.

Next Section

Algorithmic Issues: Various solution concepts

Various solution concepts of $\mathbf{A}x = \mathbf{b}$

- **Traditional solution concept:** $\exists x_0 \exists A \exists b$-**concept**
  - Solvability $\Leftrightarrow$ $(\exists x_0 \in \mathbb{R}^n)(\exists A \in \mathbf{A})(\exists b \in \mathbf{b})\ Ax_0 = b$
  - We proved: checking solvability is NP-complete
  - But we know: checking *nonnegative* solvability — polynomial time
- **Strong solvability:** $\forall A \forall b \exists x_0$-**concept**
  - Strong solvability $\Leftrightarrow$ $(\forall A \in \mathbf{A})(\forall b \in \mathbf{b})(\exists x_0 \in \mathbb{R}^n)\ Ax_0 = b$
  - Complexity: coNP-complete
  - Remains coNP-complete even if we restrict to $x_0 \geq 0$
- **Strong solution:** $\exists x_0 \forall A \forall b$-**concept**
  - $x_0 \in \mathbb{R}^n$ is a *strong solution* if $(\forall A \in \mathbf{A})(\forall b \in \mathbf{b})\ Ax_0 = b$
  - Existence of a strong solution $\Leftrightarrow$ $(\exists x_0 \in \mathbb{R}^n)(\forall A \in \mathbf{A})(\forall b \in \mathbf{b})\ Ax_0 = b$
  - Complexity of testing existence: polynomial time
  - **Remark.** Strong solutions exist very rarely; for example, a necessary condition for existence is $b^\Delta = 0$ (Exercise)
  - **Caution.** In case of linear *inequalities*, the situation is different: a system $\mathbf{A}x \leq \mathbf{b}$ is strongly solvable iff it has a strong solution. But nothing similar holds for equations...

AlgoIss: Various solution concepts (contd.)

Various solution concepts of $\mathbf{A}x = \mathbf{b}$

- **Tolerable solution:** $\exists x_0 \forall A \exists b$**-concept**
  - Existence of a tolerable solution $\Leftrightarrow$ $(\exists x_0 \in \mathbb{R}^n)(\forall A \in \mathbf{A})(\exists b \in \mathbf{b})\ Ax_0 = b$.
  - Complexity: polynomial time
- **Control solution:** $\exists x_0 \forall b \exists A$**-concept**
  - Existence of a control solution $\Leftrightarrow$ $(\exists x_0 \in \mathbb{R}^n)(\forall b \in \mathbf{b})(\exists A \in \mathbf{A})\ Ax_0 = b$
  - Complexity: NP-complete
- **AE-solution:** $\exists x_0 \forall A^\forall \forall b^\forall \exists A^\exists \exists b^\exists$**-concept**
  - Existence of AE-solution $\Leftrightarrow$ $(\exists x_0 \in \mathbb{R}^n)(\forall A^\forall \in \mathbf{A}^\forall)(\forall b^\forall \in \mathbf{b}^\forall)(\exists A^\exists \in \mathbf{A}^\exists)(\exists b^\exists \in \mathbf{b}^\exists)\ (A^\forall + A^\exists)x_0 = b^\forall + b^\exists$
  - Complexity: NP-complete
  - To recall: 2^n-algorithm — orthant decomposition by Rohn's Theorem

AlgoIss: Various solution concepts

A natural generalization

- One can imagine a natural generalization to any level of quantifier complexity, e.g.
  - Σ_k-solution: $\exists\forall\exists\cdots$ with $k-1$ quantifier alternations,
  - Π_k-solution: $\forall\exists\forall\cdots$ with $k-1$ quantifier alternations.
- Study of formulae with Σ_k- and Π_k-prefixes is popular in logic (recall e.g. the Arithmetical Hierarchy) as well as in Complexity Theory (recall e.g. the Polynomial Time Hierarchy).
- About Σ_k- and Π_k-solutions we can say only that checking existence is recursive: can be decided (in double-exponential time) via Tarski's Quantifier Elimination Method
- But possibly more could be said and more efficient methods might exist...
- If logic and complexity theory "like" building hierarchies based on quantifier complexity, why couldn't we try something similar in interval analysis?

AlgoIss: Regularity

Let E denote the all-one matrix.

Proposition

The following statements are equivalent:

(a) *Rohn's system* $|Ax| \leq e, \|x\|_1 \geq 1$ *is solvable.*

(b) *Interval system* $[A - E, A + E]x = 0$, $[-e^T, e^T]x = 1$ *has a solution.*

(c) $[A - E, A + E]$ *is singular* ($=$ *contains a singular matrix*).

Corollary

(a) *Checking regularity of an interval matrix is a coNP-complete problem.*

(b) *Checking existence of a solution of an interval system* $\mathbf{A}x = \mathbf{b}$ *is an NP-complete problem.* (*This is another proof of a previously proved statement.*)

(c) *Checking existence of a* **control** *solution of an interval system* $\mathbf{A}x = \mathbf{b}$ *is an NP-complete problem.*

AlgoIss: Regularity (contd.)

Proof of Proposition

Step 1. Singularity of $[A-E, A+E]$ $\Leftrightarrow$ solvability of $[A-E, A+E]x = 0$, $[-e^T, e^T]x = 1$.

- $A' \in [A-E, A+E]$ is singular $\Leftrightarrow A'x = 0, \|x\|_1 = 1$ is solvable $\Leftrightarrow$ $A'x = 0, \mathrm{sgn}(x)^T x = 1$ is solvable. Now $A' \in [A-E, A+E]$, $\mathrm{sgn}(x)^T \in [-e^T, e^T]$.
- $A'x = 0, c^T x = 1$ is solvable for $A' \in [A-E, A+E], c^T \in [-e^T, e^T]$ $\Rightarrow$ there is a solution $x \neq 0 \Rightarrow A'$ is singular.

Step 2. Solvability of $|Ax| \leq e, \|x\|_1 \geq 1$ $\Leftrightarrow$ solvability of $[A-E, A+E]x = 0$, $[-e^T, e^T]x = 1$.

- x solves $|Ax| \leq e, \|x\|_1 \geq 1$ iff $x' := \frac{x}{\|x\|_1}$ solves

$$\left| \begin{pmatrix} A \\ 0^T \end{pmatrix} x - \begin{pmatrix} 0 \\ 1 \end{pmatrix} \right| \leq \begin{pmatrix} E \\ e^T \end{pmatrix} |x|.$$

The last inequality is Oettli-Prager expression for the solution set of $[A-E, A+E]x = 0$, $[-e^T, e^T]x = 1$. $\square$

Exercise

Prove in detail that checking regularity of a given interval matrix is indeed in coNP.

Interval linear inequalities

Interval Programming 4

Milan Hladík [1] Michal Černý [2]

[1] Faculty of Mathematics and Physics,
Charles University in Prague, Czech Republic
http://kam.mff.cuni.cz/~hladik/

[2] Faculty of Computer Science and Statistics,
University of Economics, Prague, Czech Republic
http://nb.vse.cz/~cernym/

Workshop on Interval Programming
7th International Conference of Iranian Operation Research Society
Semnan, Iran, May 12–13, 2014

Outline

1. Software Presentation
2. Interval Linear Inequalities – Solution Set
3. Algorithmic Issues

Next Section

1. Software Presentation

2. Interval Linear Inequalities – Solution Set

3. Algorithmic Issues

Software Presentation

- **webComputing** (E. Popova)
  visualization of solution sets
  `http://cose.math.bas.bg/webComputing/`
- **Intlab** (S. M. Rump)
  interval library for Matlab
  `http://www.ti3.tu-harburg.de/rump/intlab/`

Next Section

Solution Set

Interval Linear Inequalities

Let $\mathbf{A} \in \mathbb{IR}^{m \times n}$ and $\mathbf{b} \in \mathbb{IR}^{m}$. The family of systems

$$Ax \leq b, \quad A \in \mathbf{A}, \ b \in \mathbf{b}.$$

is called interval linear inequalities and abbreviated as $\mathbf{A}x \leq \mathbf{b}$.

Solution set

The solution set is defined

$$\Sigma := \{x \in \mathbb{R}^n : \exists A \in \mathbf{A} \exists b \in \mathbf{b} : Ax \leq b\}.$$

Theorem (Gerlach, 1981)

A vector $x \in \mathbb{R}^n$ is a solution of $\mathbf{A}x \leq \mathbf{b}$ if and only if

$$A^c x \leq A^{\Delta}|x| + \overline{b}.$$

Corollary

An $x \in \mathbb{R}^n$ is a solution of $\mathbf{A}x \leq \mathbf{b}$, $x \geq 0$ if and only if $\underline{A}x \leq \overline{b}$, $x \geq 0$.

Proof of Gerlach's Theorem

Theorem (Gerlach, 1981)

A vector $x \in \mathbb{R}^n$ *is a solution of* $\mathbf{A}x \leq \mathbf{b}$ *if and only if*

$$A^c x \leq A^\Delta |x| + \overline{b}. \tag{1}$$

Proof.

If x is a solution of $\mathbf{A}x \leq \mathbf{b}$, then $Ax \leq b$ for some $A \in \mathbf{A}$ and $b \in \mathbf{b}$, and one has

$$A^c x \leq A^c x + b - Ax = (A^c - A)x + b \leq |(A^c - A)||x| + b \leq A^\Delta |x| + \overline{b}.$$

Conversely, let x satisfy (1). Set $z := \operatorname{sgn}(x)$, so $|x| = \operatorname{diag}(z)x$. Thus (1) takes the form of

$$A^c x \leq A^\Delta \operatorname{diag}(z)x + \overline{b},$$

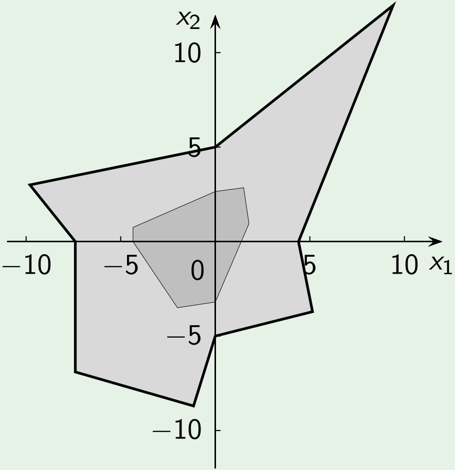
or

$$(A^c - A^\Delta \operatorname{diag}(z))x \leq \overline{b}.$$

Hence x fulfills $Ax \leq b$ for $b := \overline{b}$ and $A := A^c - A^\Delta \operatorname{diag}(z)$. □

Example of the Solution Set

Example (An interval polyhedron)



$$\begin{pmatrix} -[2,5] & -[7,11] \\ [1,13] & -[4,6] \\ [5,8] & [-2,1] \\ -[1,4] & [5,9] \\ -[5,6] & -[0,4] \end{pmatrix} x \leq \begin{pmatrix} [61,63] \\ [19,20] \\ [15,22] \\ [24,25] \\ [26,37] \end{pmatrix}$$

- union of all feasible sets in light gray,
- intersection of all feasible sets in dark gray,

Strong Solution

Strong Solution

A vector $x \in \mathbb{R}^n$ is a strong solution to $\mathbf{A}x \leq \mathbf{b}$ if $Ax \leq b$ for every $A \in \mathbf{A}$ and $b \in \mathbf{b}$.

Theorem (Rohn & Kreslová, 1994)

A vector $x \in \mathbb{R}^n$ is a strong solution iff there are $x^1, x^2 \in \mathbb{R}^n$ such that

$$x = x^1 - x^2,\ \overline{A}x^1 - \underline{A}x^2 \leq \underline{b},\ x^1 \geq 0,\ x^2 \geq 0. \tag{2}$$

Theorem (Machost, 1970)

A vector $x \in \mathbb{R}^n$ is a strong solution $\mathbf{A}x \leq \mathbf{b},\ x \geq 0$ iff it solves

$$\overline{A}x \leq \underline{b},\ x \geq 0.$$

Proof.

One direction is trivial.

Conversely, if x^* solves $\overline{A}x \leq \underline{b},\ x \geq 0$, then for each $A \in \mathbf{A}$ and $b \in \mathbf{b}$,

$$Ax^* \leq \overline{A}x^* \leq \underline{b} \leq b.$$

□

Strong Solution

Theorem (Rohn & Kreslová, 1994)

An interval system $\mathbf{A}x \leq \mathbf{b}$ *(*$x \geq 0$*) has a strong solution iff* $Ax \leq b$ *is feasible for each* $A \in \mathbf{A}$ *and* $b \in \mathbf{b}$*.*

Proof.

One direction obvious, the latter not obvious. $\square$

Remark

The statement is surprising. Analogy for interval equations does not hold, for example

$$x + y = [1, 2], \quad x - y = [2, 3]$$

is feasible for each realization, but there is no common solution.

Exercises for YOU

What are topological properties of the solution set to $\mathbf{A}x \leq \mathbf{b}$?

1. Can Σ be disconnected?
2. Can Σ have both bounded and unbounded connectivity components?
3. Can Σ have several bounded connectivity components?

Summary of Solution Set Descriptions

solution type	description
solution of $\mathbf{A}x = \mathbf{b}$	$\lvert A^c x - b^c \rvert \le A^\Delta \lvert x \rvert + b^\Delta$
strong solution of $\mathbf{A}x = \mathbf{b}$	$A^c x - b^c = A^\Delta \lvert x \rvert = b^\Delta = 0$
tolerance solution of $\mathbf{A}x = \mathbf{b}$	$\lvert A^c x - b^c \rvert \le -A^\Delta \lvert x \rvert + b^\Delta$
solution of $\mathbf{A}x \le \mathbf{b}$	$A^c x - b^c \le A^\Delta \lvert x \rvert + b^\Delta$
strong solution of $\mathbf{A}x \le \mathbf{b}$	$A^c x - b^c \le -A^\Delta \lvert x \rvert - b^\Delta$

References

M. Fiedler, J. Nedoma, J. Ramík, J. Rohn, and K. Zimmermann.
Linear optimization problems with inexact data.
Springer, New York, 2006.

W. Gerlach.
Zur Lösung linearer Ungleichungssysteme bei Störung der rechten Seite und der Koeffizientenmatrix.
Math. Operationsforsch. Stat., Ser. Optimization, 12:41–43, 1981.

M. Hladík.
Weak and strong solvability of interval linear systems of equations and inequalities.
Linear Algebra Appl., 438(11):4156–4165, 2013.

B. Machost.
Numerische Behandlung des Simplexverfahrens mit intervallanalytischen Methoden.
Technical Report 30, Berichte der Gesellschaft für Mathematik und Datenverarbeitung, 54 pages, Bonn, 1970.

J. Rohn and J. Kreslová.
Linear interval inequalities.
Linear Multilinear Algebra, 38(1-2):79–82, 1994.

Next Section

Algorithmic Issues: Inequalities

Polynomial-time cases

- **Nonnegative solvability.** By Gerlach: the system $\mathbf{A}x \leq \mathbf{b}, x \geq 0$ is solvable $\Leftrightarrow$ the system $\underline{A}x \leq \overline{b}, x \geq 0$ is solvable (LP).
- **Strong nonnegative solvability (and existence of a strong nonnegative solution).** Check $\overline{A}x \leq \underline{b}, x \geq 0$ (LP).
- **Strong solvability (and existence of a strong solution).** The system $\mathbf{A}x \leq \mathbf{b}$ is strongly solvable $\Leftrightarrow$ it has a strong solution x^0 $\Leftrightarrow$ $(\exists x^1, x^2 \geq 0)$ s.t. $x^0 = x^1 - x^2$ and $\overline{A}x^1 - \underline{A}x^2 \leq \underline{b}$ (LP).

Theorem

Checking solvability of $\mathbf{A}x \leq \mathbf{b}$ *is NP-complete.*

AlgoIss: NP-completeness of solvability

Proof.

Rohn's system $|Ax| \le e,\ \|x\|_1 \ge 1$ can be rewritten as

$$\begin{pmatrix} A \\ -A \\ 0^T \end{pmatrix} x - \begin{pmatrix} 0 \\ 0 \\ e^T \end{pmatrix} |x| \le \begin{pmatrix} e \\ e \\ -1 \end{pmatrix}$$

and this is Gerlach's inequality for

$$\begin{pmatrix} [A, A] \\ [-A, -A] \\ [-e^T, e^T] \end{pmatrix} x \le \begin{pmatrix} e \\ e \\ -1 \end{pmatrix}.$$

$\square$

Remark. Observe that there is no "dependency problem" even if A occurs in both the first and the second inequality.

Remark. Observe that the problem is NP-complete even if $\mathbf{b}$ is crisp and $\mathbf{A}$ has intervals in one row only.

AlgoIss: Comparison

	$\mathbf{A}x = \mathbf{b}$	$\mathbf{A}x \leq \mathbf{b}$
solvability $x \in \mathbb{R}^n$	NP-complete	NP-complete
solvability $x \geq 0$	poly-time	poly-time
strong solvability $x \in \mathbb{R}^n$	coNP-complete	poly-time
strong solvability $x \geq 0$	coNP-complete	poly-time

To recall: *strong solvability* means

$$(\forall A \in \mathbf{A})(\forall b \in \mathbf{b})(\exists x \in \mathbb{R}^n)\ Ax = b \ \ (Ax \leq b).$$

Interval linear programming

Interval Programming 5

Milan Hladík [1] Michal Černý [2]

[1] Faculty of Mathematics and Physics,
Charles University in Prague, Czech Republic
http://kam.mff.cuni.cz/~hladik/

[2] Faculty of Computer Science and Statistics,
University of Economics, Prague, Czech Republic
http://nb.vse.cz/~cernym/

Workshop on Interval Programming
7th International Conference of Iranian Operation Research Society
Semnan, Iran, May 12–13, 2014

Outline

Next Section

Introduction

Linear programming – three basic forms

$$
\begin{aligned}
f(A,b,c) &\equiv \min c^T x \ \text{ subject to } \ Ax = b,\ x \geq 0,\\
f(A,b,c) &\equiv \min c^T x \ \text{ subject to } \ Ax \leq b,\\
f(A,b,c) &\equiv \min c^T x \ \text{ subject to } \ Ax \leq b,\ x \geq 0.
\end{aligned}
$$

Interval linear programming

Family of linear programs with $A \in \mathbf{A}$, $b \in \mathbf{b}$, $c \in \mathbf{c}$, in short

$$
f(\mathbf{A},\mathbf{b},\mathbf{c}) \equiv \min \mathbf{c}^T x \ \text{ subject to } \ \mathbf{A}x \overset{(\leq)}{=} \mathbf{b},\ (x \geq 0).
$$

The three forms are not transformable between each other!

Goals

- determine the optimal value range;
- determine a tight enclosure to the optimal solution set.

Complexity of Basic Problems

	$\mathbf{A}x = \mathbf{b},\ x \geq 0$	$\mathbf{A}x \leq \mathbf{b}$	$\mathbf{A}x \leq \mathbf{b},\ x \geq 0$
strong feasibility	co-NP-hard	polynomial	polynomial
weak feasibility	polynomial	NP-hard	polynomial
strong unboundedness	co-NP-hard	polynomial	polynomial
weak unboundedness	suff. / necessary conditions only	suff. / necessary conditions only	polynomial
strong optimality	co-NP-hard	co-NP-hard	polynomial
weak optimality	suff. / necessary conditions only	suff. / necessary conditions only	suff. / necessary conditions only
optimal value range	$\underline{f}$ polynomial $\overline{f}$ NP-hard	$\underline{f}$ NP-hard $\overline{f}$ polynomial	polynomial

Next Section

Optimal Value Range

Definition

$$\underline{f} := \min f(A, b, c) \;\; \text{subject to} \;\; A \in \mathbf{A}, \;\; b \in \mathbf{b}, \;\; c \in \mathbf{c},$$
$$\overline{f} := \max f(A, b, c) \;\; \text{subject to} \;\; A \in \mathbf{A}, \;\; b \in \mathbf{b}, \;\; c \in \mathbf{c}.$$

Theorem (Vajda, 1961)

We have for type ($\mathbf{A}x \leq \mathbf{b}, \;\; x \geq 0$)

$$\underline{f} = \min \underline{c}^T x \;\; \textit{subject to} \;\; \underline{A}x \leq \overline{b}, \;\; x \geq 0,$$
$$\overline{f} = \min \overline{c}^T x \;\; \textit{subject to} \;\; \overline{A}x \leq \underline{b}, \;\; x \geq 0.$$

Theorem (Rohn, 2006)

We have for type ($\mathbf{A}x = \mathbf{b}, \;\; x \geq 0$)

$$\underline{f} = \min \underline{c}^T x \;\; \textit{subject to} \;\; \underline{A}x \leq \overline{b}, \;\; \overline{A}x \geq \underline{b}, \;\; x \geq 0,$$
$$\overline{f} = \max_{p \in \{\pm 1\}^m} f(A^c - \operatorname{diag}(p) A^\Delta, b^c + \operatorname{diag}(p) b^\Delta, \overline{c}).$$

Optimal Value Range

Algorithm (Optimal value range $[\underline{f}, \overline{f}]$)

1. Compute
$$\underline{f} := \inf\ (c^c)^T x - (c^\Delta)^T |x| \ \text{ subject to } \ x \in \mathcal{M},$$
where $\mathcal{M}$ is the primal solution set.
2. If $\underline{f} = \infty$, then set $\overline{f} := \infty$ and stop.
3. Compute
$$\overline{\varphi} := \sup\ (b^c)^T y + (b^\Delta)^T |y| \ \text{ subject to } \ y \in \mathcal{N},$$
where $\mathcal{N}$ is the dual solution set.
4. If $\overline{\varphi} = \infty$, then set $\overline{f} := \infty$ and stop.
5. If the primal problem is strongly feasible, then set $\overline{f} := \overline{\varphi}$; otherwise set $\overline{f} := \infty$.

Next Section

Optimal Solution Set

The optimal solution set

Denote by $\mathcal{S}(A, b, c)$ the set of optimal solutions to

$$\min c^T x \ \text{ subject to } \ Ax = b, \ x \geq 0,$$

Then the optimal solution set is defined

$$\mathcal{S} := \bigcup_{A \in \mathbf{A},\, b \in \mathbf{b},\, c \in \mathbf{c}} \mathcal{S}(A, b, c).$$

Goal

Find a tight enclosure to $\mathcal{S}$.

Characterization

By duality theory, we have that $x \in \mathcal{S}$ if and only if there is some $y \in \mathbb{R}^m$, $A \in \mathbf{A}$, $b \in \mathbf{b}$, and $c \in \mathbf{c}$ such that

$$Ax = b, \ x \geq 0, \ A^T y \leq c, \ c^T x = b^T y,$$

where $A \in \mathbf{A}$, $b \in \mathbf{b}$, $c \in \mathbf{c}$.

Next Section

Basis Stability

Definition

The interval linear programming problem

$$\min \mathbf{c}^T x \quad \text{subject to} \quad \mathbf{A}x = \mathbf{b},\ x \geq 0,$$

is B-stable if B is an optimal basis for each realization.

Theorem

B-stability implies that the optimal value bounds are

$$\underline{f} = \min\ \underline{c}_B^T x \quad \textit{subject to} \quad \underline{A}_B x_B \leq \overline{b},\ -\overline{A}_B x_B \leq -\underline{b},\ x_B \geq 0,$$

$$\overline{f} = \max\ \overline{c}_B^T x \quad \textit{subject to} \quad \underline{A}_B x_B \leq \overline{b},\ -\overline{A}_B x_B \leq -\underline{b},\ x_B \geq 0.$$

Under the unique B-stability, the set of all optimal solutions reads

$$\underline{A}_B x_B \leq \overline{b},\ -\overline{A}_B x_B \leq -\underline{b},\ x_B \geq 0,\ x_N = 0.$$

(Otherwise each realization has at least one optimal solution in this set.)

Basis Stability

Non-interval case

Basis B is optimal iff

C1. A_B is non-singular;

C2. $A_B^{-1}b \geq 0$;

C3. $c_N^T - c_B^T A_B^{-1} A_N \geq 0^T$.

Interval case

The problem is B-stable iff C1–C3 holds for each $A \in \mathbf{A}$, $b \in \mathbf{b}$, $c \in \mathbf{c}$.

Condition C1

- C1 says that $\mathbf{A}_B$ is regular;
- co-NP-hard problem;
- Beeck's sufficient condition: $\rho\left(|((A^c)_B)^{-1}|(A^\Delta)_B\right) < 1$.

Basis Stability

Non-interval case

Basis B is optimal iff

C1. A_B is non-singular;

C2. $A_B^{-1}b \geq 0$;

C3. $c_N^T - c_B^T A_B^{-1} A_N \geq 0^T$.

Interval case

The problem is B-stable iff C1–C3 holds for each $A \in \mathbf{A}$, $b \in \mathbf{b}$, $c \in \mathbf{c}$.

Condition C2

- C2 says that the solution set to $\mathbf{A}_B x_B = \mathbf{b}$ lies in $\mathbb{R}^n_+$;
- sufficient condition: check of some enclosure to $\mathbf{A}_B x_B = \mathbf{b}$.

Basis Stability

Non-interval case

Basis B is optimal iff

C1. A_B is non-singular;

C2. $A_B^{-1}b \geq 0$;

C3. $c_N^T - c_B^T A_B^{-1} A_N \geq 0^T$.

Interval case

The problem is B-stable iff C1–C3 holds for each $A \in \mathbf{A}$, $b \in \mathbf{b}$, $c \in \mathbf{c}$.

Condition C3

- C2 says that $\mathbf{A}_N^T y \leq \mathbf{c}_N,\ \mathbf{A}_B^T y = \mathbf{c}_B$ is strongly feasible;
- co-NP-hard problem;
- <u>sufficient</u> condition:
  $(\mathbf{A}_N^T)\mathbf{y} \leq \underline{c}_N$, where $\mathbf{y}$ is an enclosure to $\mathbf{A}_B^T y = \mathbf{c}_B$.

Basis Stability

Theorem

Condition C3 holds true if and only if for each $q \in \{\pm 1\}^m$ the polyhedral set described by

$$\begin{aligned} ((A^c)_B^T - (A^\Delta)_B^T \operatorname{diag}(q))y &\leq \overline{c}_B, \\ -((A^c)_B^T + (A^\Delta)_B^T \operatorname{diag}(q))y &\leq -\underline{c}_B, \\ \operatorname{diag}(q)y &\geq 0 \end{aligned}$$

lies inside the polyhedral set

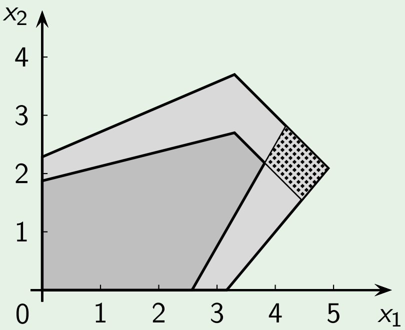
$$((A^c)_N^T + (A^\Delta)_N^T \operatorname{diag}(q))y \leq \underline{c}_N, \ \operatorname{diag}(q)y \geq 0.$$

Example

Example

Consider an interval linear program

$$\max\ ([5,6],[1,2])^T x \ \text{ s.t. } \begin{pmatrix} -[2,3] & [7,8] \\ [6,7] & -[4,5] \\ 1 & 1 \end{pmatrix} x \le \begin{pmatrix} [15,16] \\ [18,19] \\ [6,7] \end{pmatrix}, \ x \ge 0.$$



- union of all feasible sets in light gray,
- intersection of all feasible sets in dark gray,
- set of optimal solutions in dotted area

Basis Stability – Interval Right-Hand Side

Interval case

Basis B is optimal iff

C1. A_B is non-singular;

C2. $A_B^{-1}b \geq 0$ for each $b \in \mathbf{b}$.

C3. $c_N^T - c_B^T A_B^{-1} A_N \geq 0^T$.

Condition C1

- C1 and C3 are trivial
- C2 is simplified to

  $$\underline{A_B^{-1}\mathbf{b}} \geq 0,$$

  which is easily verified by interval arithmetic
- overall complexity: polynomial

Basis Stability – Interval Objective Function

Interval case

Basis B is optimal iff

C1. A_B is non-singular;

C2. $A_B^{-1} b \geq 0$;

C3. $c_N^T - c_B^T A_B^{-1} A_N \geq 0^T$ for each $c \in \mathbf{c}$

Condition C1

- C1 and C2 are trivial
- C3 is simplified to

$$A_N^T y \leq \mathbf{c}_N, \; A_B^T y = \mathbf{c}_B$$

or,

$$\overline{(A_N^T A_B^{-T}) \mathbf{c}_B} \leq \underline{c}_N.$$

- overall complexity: polynomial

Next Section

Applications

Real-Life Applications

- Transportation problems with uncertain demands, suppliers, and/or costs.
- Networks flows with uncertain capacities.
- Diet problems with uncertain amounts of nutrients in foods.
- Portfolio selection with uncertain rewards.
- Matrix games with uncertain payoffs.

Technical Applications

- Tool for global optimization.
- Measure of sensitivity of linear programs.

Applications – Diet Problem

Example (Stigler's Nutrition Model)

`http://www.gams.com/modlib/libhtml/diet.htm`.

- $n = 20$ different types of food,
- $m = 9$ nutritional demands,
- a_{ij} is the the amount of nutrient j contained in one unit of food i,
- b_i is the required minimal amount of nutrient j,
- c_j is the price per unit of food j,
- minimize the overall cost

The model reads

$$\min c^T x \quad \text{subject to} \quad Ax \geq b, \; x \geq 0.$$

The entries a_{ij} are not stable!

Applications – Diet Problem

Example (Stigler's Nutrition Model (cont.))

Nutritive value of foods (per dollar spent)

	calorie (1000)	protein (g)	calcium (g)	iron (mg)	vitamin-a (1000iu)	vitamin-b1 (mg)	vitamin-b2 (mg)	niacin (mg)	vitamin-c (mg)
wheat	44.7	1411	2.0	365		55.4	33.3	441	
cornmeal	36	897	1.7	99	30.9	17.4	7.9	106	
cannedmilk	8.4	422	15.1	9	26	3	23.5	11	60
margarine	20.6	17	.6	6	55.8	.2			
cheese	7.4	448	16.4	19	28.1	.8	10.3	4	
peanut-b	15.7	661	1	48		9.6	8.1	471	
lard	41.7				.2		.5	5	
liver	2.2	333	.2	139	169.2	6.4	50.8	316	525
porkroast	4.4	249	.3	37		18.2	3.6	79	
salmon	5.8	705	6.8	45	3.5	1	4.9	209	
greenbeans	2.4	138	3.7	80	69	4.3	5.8	37	862
cabbage	2.6	125	4	36	7.2	9	4.5	26	5369
onions	5.8	166	3.8	59	16.6	4.7	5.9	21	1184
potatoes	14.3	336	1.8	118	6.7	29.4	7.1	198	2522
spinach	1.1	106		138	918.4	5.7	13.8	33	2755
sweet-pot	9.6	138	2.7	54	290.7	8.4	5.4	83	1912
peaches	8.5	87	1.7	173	86.8	1.2	4.3	55	57
prunes	12.8	99	2.5	154	85.7	3.9	4.3	65	257
limabeans	17.4	1055	3.7	459	5.1	26.9	38.2	93	
navybeans	26.9	1691	11.4	792		38.4	24.6	217	

Applications – Diet Problem

Example (Stigler's Nutrition Model (cont.))

If the entries a_{ij} are known with 10% accuracy, then

- the problem is not basis stable
- the minimal cost ranges in $[0.09878, 0.12074]$,
- the interval enclosure of the solution set is

$$[0, 0.0734],\ [0, 0.0438],\ [0, 0.0576],\ [0, 0.0283],\ [0, 0.0535],\ [0, 0.0315],\ [0, 0.0339],$$
$$[0, 0.0300],\ [0, 0.0246],\ [0, 0.0337],\ [0, 0.0358],\ [0, 0.0387],\ [0, 0.0396],\ [0, 0.0429],$$
$$[0, 0.0370],\ [0, 0.0443],\ [0, 0.0290],\ [0, 0.0330],\ [0, 0.0472],\ [0, 0.1057].$$

If the entries a_{ij} are known with 1% accuracy, then

- the problem is basis stable
- the minimal cost ranges in $[0.10758, 0.10976]$,
- the interval hull of the solution set is

$$x_1 = [0.0282, 0.0309],\ x_8 = [0.0007, 0.0031],\ x_{12} = [0.0110, 0.0114],$$
$$x_{15} = [0.0047, 0.0053],\ x_{20} = [0.0600, 0.0621].$$

Research Directions

Research Directions

- Special cases of linear programs.
- Generalizations to nonlinear, multiobjective and other programs.
- Considering simple dependencies (H., Č., 2014).
- Approximation of NP-hard optimal value bounds (H., 2014)
- Other concepts of optimality; similarly to AE-solutions.
  (W. Li, J. Luo et al., 2013, 2014)

Open problems

Open Problems

- A sufficient and necessary condition for weak unboundedness, strong boundedness and weak optimality.
- A method for determining the image of the optimal value function.
- A sufficient and necessary condition for duality gap to be zero for each realization.
- A method to test if a basis B is optimal for some realization.
- Tight enclosure to the optimal solution set.

References

M. Fiedler, J. Nedoma, J. Ramík, J. Rohn, and K. Zimmermann.
Linear optimization problems with inexact data.
Springer, New York, 2006.

M. Hladík.
Interval linear programming: A survey.
In Z. A. Mann, editor, *Linear Programming – New Frontiers in Theory and Applications*, chapter 2, pages 85–120. Nova Science Publishers, 2012.

M. Hladík.
Weak and strong solvability of interval linear systems of equations and inequalities.
Linear Algebra Appl., 438(11):4156–4165, 2013.

M. Hladík.
How to determine basis stability in interval linear programming.
Optim. Lett., 8(1):375–389, 2014.

W. Li, J. Luo and C. Deng.
Necessary and sufficient conditions of some strong optimal solutions to the interval linear programming.
Linear Algebra Appl., 439(10):3241–3255, 2013.

Next Section

AlgoIss: Optimal Value Range for the form $\mathbf{A}x = \mathbf{b}, x \geq 0$

To recall: By correctness of the Optimal Value Range algorithm, we have the following form of IntLP-duality:

Lemma

If $\overline{f} = \sup_{(A,b,c)\in(\mathbf{A},\mathbf{b},\mathbf{c})} \inf\{c^T x : A^T x = b, x \geq 0\}$ *is finite, then*

$$\overline{f} = \overline{\varphi} \ := \sup\{(b^c)^T y + (b^\Delta)^T |y| : y \in \mathcal{N}\},$$

where $\mathcal{N} = \bigcup_{(A,b,c)\in(\mathbf{A},\mathbf{b},\mathbf{c})}\{y : Ay \leq c\} = \{y : A^c y - A^\Delta |y| \leq \overline{c}\}$ *is the dual solution set.*

An interesting special case with crisp $A^T = (A^T, -A^T), c = (e^T, e^T)$ and interval $\mathbf{b} = [-e, e]$:

Corollary

If $\overline{f} = \sup_{b\in[-e,e]} \inf\{e^T x^1 + e^T x^2 : A^T(x^1 - x^2) = b, x^1 \geq 0, x^2 \geq 0\}$ *is finite, then*

$$\overline{f} = \max\{e^T |y| : -e \leq Ay \leq e\} \ (= \max\{\|y\|_1 : -e \leq Ay \leq e\}).$$

Algolss: Optimal Value Range for the form $\mathbf{A}x = \mathbf{b}, x \geq 0$

We have almost proved:

Theorem

Computation of $\overline{f}$ is NP-hard for the form $\mathbf{A}x = \mathbf{b}, x \geq 0$.

Proof.

- The following form of Rohn's generic problem is NP-complete: *given a regular matrix A, decide whether the system $-e \leq Ax \leq e$, $\|x\|_1 \geq 1$ is solvable*. [Singular matrices can be easily excluded: if A is singular, then $Ax = 0, \|x\|_1 \geq 1$ has a solution, and so does $-e \leq Ax \leq e, \|x\|_1 \geq 1$.]
- Let a regular matrix A be given and consider the problem of computing $\overline{f}$ for

$$\min e^T x^1 + e^T x^2 : \; A^T(x^1 - x^2) = [-e, e], \; x^1 \geq 0, \; x^2 \geq 0.$$

The dual feasible set $\mathcal{N} = \{y : -e \leq Ay \leq e\}$ is nonempty ($0 \in \mathcal{N}$) and bounded (since A is regular); moreover, $|b| \leq e$ is also bounded. Thus the dual problem is feasible and bounded for every $b \in [-e, e]$, and so is the primal problem by LP-duality. Thus $\overline{f}$ is finite and

$$\overline{f} = \max\{\|y\|_1 : -e \leq Ay \leq e\}.$$

Now $\overline{f} \geq 1$ iff $-e \leq Ax \leq e, \|x\|_1 \geq 1$ is solvable. $\square$

AlgoIss: Comments

Comments

- **To recall:** $\underline{f}$ is computable in polynomial time by the LP
$$\min \underline{c}^T x \text{ s.t. } \underline{A}x \le \overline{b}, \overline{A}x \ge \underline{b}, x \ge 0.$$
- The NP-hardness result shows that the 2^n-algorithm based on orthant decomposition
$$\overline{f} = \max_{s \in \{\pm 1\}} \min\{\overline{c}^T x : (A^c - T_s A^\Delta)x = \ b^c + T_s b^\Delta,\ x \ge 0\}$$
with $T_s = \text{diag}(s)$ is the "best possible".

Exercise

- Prove analogous results for the forms
  - $\mathbf{A}x \le \mathbf{b}$: $\overline{f}$ poly-time, $\underline{f}$ NP-hard;
  - $\mathbf{A}x \le \mathbf{b},\ x \ge 0$: both $\overline{f}$, $\underline{f}$ poly-time.
- Note that duality plays role here: the forms $Ax \le b$ and $Ax = b, x \ge 0$ are dual to each other and complexity results are "complementary". The form $Ax \le b, x \ge 0$ is "self-dual".

Linear regression

- Consider the linear regression model

$$y = X\beta + \varepsilon,$$

where columns of X are *regressors* and y is a *dependent variable*. Often we use minimum norm estimators

  - $\widehat{\beta} = \mathrm{argmin}_\beta \|y - X\beta\|_2 = (X^T X)^{-1} X^T y$ (least squares),
  - $\widehat{\beta} = \mathrm{argmin}_\beta \|y - X\beta\|_1$ (least absolute deviations),
  - $\widehat{\beta} = \mathrm{argmin}_\beta \|y - X\beta\|_\infty$ (Chebyshev approximation).

- The $\|\cdot\|_1$ and $\|\cdot\|_\infty$ problems can be written as linear programs:

$$\min_{r,\beta} e^T r \text{ s.t. } X\beta - y \le r, \quad -X\beta + y \le r, \quad r \ge 0.$$

$$\min_{t,\beta} t \text{ s.t. } X\beta - y \le te, \quad -X\beta + y \le te, \quad t \ge 0.$$

- We will consider the latter problem with interval data $(\mathbf{X}, \mathbf{y})$:

$$\min_{t,\beta} t \text{ s.t. } \mathbf{X}\beta - \mathbf{y} \le te, \quad -\mathbf{X}\beta + \mathbf{y} \le te, \quad t \ge 0.$$

AlgoIss: Basis stability (contd.)

We are given interval data $(\mathbf{X}, \mathbf{y})$ and we are to solve

$$\min_{t,\beta} t \text{ s.t. } \mathbf{X}\beta - \mathbf{y} \leq te, \quad -\mathbf{X}\beta + \mathbf{y} \leq te, \quad t \geq 0. \tag{1}$$

Illustration

Basis stability = robustness of classification:

- Let Class 1 be defined by $C_1 = \{i : y_i \geq X_{i,:}\widehat{\beta}\}$.
- Let Class 2 be defined by $C_2 = \{i : y_i \leq X_{i,:}\widehat{\beta}\}$.
- Basis stability: the same classification (i.e. $C_1 = C_2$) for every $(X, y) \in (\mathbf{X}, \mathbf{y})$.

Theorem

Testing basis stability of the interval LP (1) is a coNP-complete problem.

Remark. The IntLP (1) is a fake IntLP since it suffers from dependencies...

AlgoIss: Basis stability (contd.)

Proof

We will show that testing regularity of a given interval matrix $\mathbf{A}$ is reducible to testing basis stability of

$$\min_{t,\beta} t \text{ s.t. } \mathbf{X}\beta - \mathbf{y} \leq te, \quad -\mathbf{X}\beta + \mathbf{y} \leq te, \quad t \geq 0. \tag{2}$$

Let $\mathbf{A}$ be given and consider (2) with $(\mathbf{X}, \mathbf{y}) = (\mathbf{A}, [-e, e])$.

- **Step 1.** Regularity $\Rightarrow$ Basis stability. Let $\mathbf{X} = \mathbf{A}$ be regular. For every $X \in \mathbf{X}$, $\beta = X^{-1}y$, $t = 0$ is the optimal solution. Thus, all $2n + 1$ inequalities of the system

$$X\beta - y \leq te, \quad -X\beta + y \leq te, \quad t \geq 0$$

  hold as equations. Thus the basis $\{1, \ldots, n, 2n + 1\}$ is optimal.

- **Step 2.** Singularity $\Rightarrow$ Basis instability. Let $X_0 \in \mathbf{X} = \mathbf{A}$ be singular. We will show two different choices of $y \in [-e, e]$ leading to two different optimal bases.

AlgoIss: Basis stability (contd.)

Proof (contd.)

To recall: we work with $\min_{t,\beta} t$ s.t. $\mathbf{X}\beta - \mathbf{y} \leq te, \quad -\mathbf{X}\beta + \mathbf{y} \leq te, \quad t \geq 0.$

We want to prove Step 2: *Singularity* $\Rightarrow$ *Basis instability.* Let $X_0 \in \mathbf{X} = \mathbf{A}$ be singular. We will show two different choices of $y \in [-e, e]$ leading to two different optimal bases.

- **Choice 1:** Let $y_0 \in [-e, e]$ s.t. y_0 is linearly independent of columns of X_0. (By singularity of X_0, such a choice is possible.) Any optimal solution of

$$X_0\beta - y_0 \leq te, \quad -X_0\beta + y_0 \leq te, \quad t \geq 0$$

  must have $t > 0$ (since $t = 0$ implies $X_0\beta = y_0$ and y_0 is dependent on columns of X_0). Thus an optimal basis does not contain the inequality $t \geq 0$ (= index $2n + 1$) since always $t > 0$.
- **Choice 2:** Let $y_0 = 0$. Then $\beta = 0, t = 0$ is an optimal solution. Thus every optimum solution has $t = 0$ and we must choose an optimal basis containing $t = 0$ (= index $2n + 1$). $\square$

Eigenvalues and positive definiteness of interval matrices

Interval Programming 6

Milan Hladík [1] Michal Černý [2]

[1] Faculty of Mathematics and Physics,
Charles University in Prague, Czech Republic
`http://kam.mff.cuni.cz/~hladik/`

[2] Faculty of Computer Science and Statistics,
University of Economics, Prague, Czech Republic
`http://nb.vse.cz/~cernym/`

Workshop on Interval Programming
7th International Conference of Iranian Operation Research Society
Semnan, Iran, May 12–13, 2014

Outline

Next Section

Eigenvalues of Symmetric Interval Matrices

A Symmetric Interval Matrix

$$\mathbf{A}^S := \{A \in \mathbf{A} : A = A^T\}.$$

Without loss of generality assume that $\underline{A} = \underline{A}^T$, $\overline{A} = \overline{A}^T$, and $\mathbf{A}^S \neq \emptyset$.

Eigenvalues of a Symmetric Interval Matrix

Eigenvalues of a symmetric $A \in \mathbb{R}^{n\times n}$: $\lambda_1(A) \geq \cdots \geq \lambda_n(A)$.
Eigenvalue sets of $\mathbf{A}^S$:

$$\boldsymbol{\lambda}_i(\mathbf{A}^S) := \left\{\lambda_i(A) : A \in \mathbf{A}^S\right\}, \quad i = 1, \ldots, n.$$

Theorem

Checking whether $0 \in \boldsymbol{\lambda}_i(\mathbf{A}^S)$ *for some* $i = 1, \ldots, n$ *is NP-hard.*

Proof.

$\mathbf{A}$ is singular iff $\mathbf{M}^S := \begin{pmatrix} 0 & \mathbf{A} \\ \mathbf{A}^T & 0 \end{pmatrix}^S$ is singular (has a zero eigenvalue). $\square$

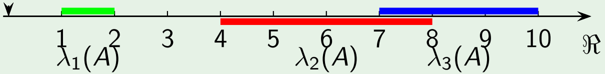
Eigenvalues – An Example

Example

Let

$$A \in \mathbf{A} = \begin{pmatrix} [1,2] & 0 & 0 \\ 0 & [7,8] & 0 \\ 0 & 0 & [4,10] \end{pmatrix}$$

What are the eigenvalue sets?
We have $\boldsymbol{\lambda}_1(\mathbf{A}^S) = [7,10]$, $\boldsymbol{\lambda}_2(\mathbf{A}^S) = [4,8]$ and $\boldsymbol{\lambda}_3(\mathbf{A}^S) = [1,2]$.



Eigenvalue sets are compact intervals. They may intersect or equal.

Eigenvalues – Some Exact Bounds

Theorem (Hertz, 1992)

We have

$$\overline{\lambda}_1(\mathbf{A}^S) = \max_{z \in \{\pm 1\}^n} \lambda_1(A^c + \operatorname{diag}(z) A^{\Delta} \operatorname{diag}(z)),$$

$$\underline{\lambda}_n(\mathbf{A}^S) = \min_{z \in \{\pm 1\}^n} \lambda_n(A^c - \operatorname{diag}(z) A^{\Delta} \operatorname{diag}(z)).$$

Proof.

"Upper bound." By contradiction suppose that there is $A \in \mathbf{A}^S$ such that

$$\lambda_1(A) > \max_{z \in \{\pm 1\}^n} \lambda_1(A_z), \quad \left[\text{where } A_z \equiv A^c + \operatorname{diag}(z) A^{\Delta} \operatorname{diag}(z)\right]$$

Thus $Ax = \lambda_1(A)x$ for some x with $\|x\|_2 = 1$.
Put $z^* := \operatorname{sgn}(x)$, and by the Rayleigh–Ritz Theorem we have

$$\begin{aligned} \lambda_1(A) &= x^T A x \leq x^T A_{z^*} x \\ &\leq \max_{y : \|y\|_2 = 1} y^T A_{z^*} y = \lambda_1(A_{z^*}). \end{aligned}$$

□

Eigenvalues – Some Other Exact Bounds

Theorem

$\underline{\lambda}_1(\mathbf{A}^S)$ and $\overline{\lambda}_n(\mathbf{A}^S)$ are polynomially computable by semidefinite programming.

Proof.

We have

$$\overline{\lambda}_n(\mathbf{A}^S) = \max \alpha \;\text{ subject to }\; A - \alpha I_n \text{ is positive semidefinite, } A \in \mathbf{A}^S.$$

Consider a block diagonal matrix $M(A, \alpha)$ with blocks

$$A - \alpha I_n,\; a_{ij} - \underline{a}_{ij},\; \overline{a}_{ij} - a_{ij},\; i \leq j.$$

Then the optimization problem reads

$$\overline{\lambda}_n(\mathbf{A}^S) = \max \alpha \;\text{ subject to }\; M(A, \alpha) \text{ is positive semidefinite.}$$

□

Eigenvalues – Enclosures

Theorem

We have

$$\boldsymbol{\lambda}_i(\mathbf{A}^S) \subseteq [\lambda_i(A^c) - \rho(A^\Delta), \lambda_i(A^c) + \rho(A^\Delta)], \quad i = 1, \dots, n.$$

Proof.

Recall for any $A, B \in \mathbb{R}^{n \times n}$,

$$|A| \le B \quad \Rightarrow \quad \rho(A) \le \rho(|A|) \le \rho(B),$$

and for A, B symmetric (Weyl's Theorem)

$$\lambda_i(A) + \lambda_n(B) \le \lambda_i(A + B) \le \lambda_i(A) + \lambda_1(B), \quad i = 1, \dots, n.$$

Let $A \in \mathbf{A}^S$, so $|A - A^c| \le A^\Delta$. Then

$$\begin{aligned} \lambda_i(A) = \lambda_i(A^c + (A - A^c)) &\le \lambda_i(A^c) + \lambda_1(A - A^c) \\ &\le \lambda_i(A^c) + \rho(|A - A^c|) \le \lambda_i(A^c) + \rho(A^\Delta). \end{aligned}$$

Similarly for the lower bound. □

Eigenvalues – Easy Cases

Theorem

1. *If A^c is essentially non-negative, i.e., $A^c_{ij} \geq 0 \ \forall i \neq j$, then*
$$\overline{\lambda}_1(\mathbf{A}^S) = \lambda_1(\overline{A}).$$
2. *If A^Δ is diagonal, then*
$$\overline{\lambda}_1(\mathbf{A}^S) = \lambda_1(\overline{A}), \quad \underline{\lambda}_n(\mathbf{A}^S) = \lambda_n(\underline{A}).$$

Proof.

1. For the sake of simplicity suppose $A^c \geq 0$. Then $\forall A \in \mathbf{A}^S$ we have $|A| \leq \overline{A}$, whence
$$\lambda_1(A) = \rho(A) \leq \rho(\overline{A}) = \lambda_1(\overline{A}).$$
2. By Hertz's theorem,
$$\begin{aligned}\overline{\lambda}_1(\mathbf{A}^S) &= \max_{z \in \{\pm 1\}^n} \lambda_1(A^c + \operatorname{diag}(z) A^\Delta \operatorname{diag}(z)), \\ &= \lambda_1(A^c + A^\Delta) = \lambda_1(\overline{A}).\end{aligned}$$

□

Next Section

Positive Semidefiniteness

$\mathbf{A}^S$ is *positive semidefinite* if every $A \in \mathbf{A}^S$ is positive semidefinite.

Theorem

The following are equivalent

1. $\mathbf{A}^S$ *is positive semidefinite,*
2. $A_z \equiv A^c - \operatorname{diag}(z) A^\Delta \operatorname{diag}(z)$ *is positive semidefinite* $\forall z \in \{\pm 1\}^n$,
3. $x^T A^c x - |x|^T A^\Delta |x| \geq 0$ *for each* $x \in \mathbb{R}^n$.

Proof.

"(1) $\Rightarrow$ (2)" Obvious from $A_z \in \mathbf{A}^S$.

"(2) $\Rightarrow$ (3)" Let $x \in \mathbb{R}^n$ and put $z := \operatorname{sgn}(x)$. Now,

$$x^T A^c x - |x|^T A^\Delta |x| = x^T A^c x - x^T \operatorname{diag}(z) A^\Delta \operatorname{diag}(z) x = x^T A_z x \geq 0.$$

"(3) $\Rightarrow$ (1)" Let $A \in \mathbf{A}^S$ and $x \in \mathbb{R}^n$. Now,

$$\begin{aligned} x^T A x = x^T A^c x + x^T (A - A^c) x &\geq x^T A^c x - |x^T (A - A^c) x| \\ &\geq x^T A^c x - |x|^T A^\Delta |x| \geq 0. \end{aligned}$$

□

Positive Definiteness

$\mathbf{A}^S$ is *positive definite* if every $A \in \mathbf{A}^S$ is positive definite.

Theorem

The following are equivalent

1. $\mathbf{A}^S$ *is positive definite,*
2. $A_z \equiv A^c - \operatorname{diag}(z)A^\Delta \operatorname{diag}(z)$ *is positive definite for each* $z \in \{\pm 1\}^n$*,*
3. $x^T A^c x - |x|^T A^\Delta |x| > 0$ *for each* $0 \neq x \in \mathbb{R}^n$*,*
4. A^c *is positive definite and* $\mathbf{A}$ *is regular.*

Proof.

"(1) $\Leftrightarrow$ (2) $\Leftrightarrow$ (3)" analogously.
"(1) $\Rightarrow$ (4)" If there are $A \in \mathbf{A}$ and $x \neq 0$ such that $Ax = 0$, then

$$0 = x^T A x = x^T \tfrac{1}{2}(A + A^T)x,$$

and so $\frac{1}{2}(A + A^T) \in \mathbf{A}^S$ is not positive definite.
"(4) $\Rightarrow$ (1)" Positive definiteness of A^c implies $\lambda_i(A^c) > 0\ \forall i$, and regularity of $\mathbf{A}$ implies $\boldsymbol{\lambda}_i(\mathbf{A}^S) > 0\ \forall i$. $\square$

Complexity

Theorem (Nemirovskii, 1993)

Checking positive semidefiniteness of $\mathbf{A}^S$ is co-NP-hard.

Theorem (Rohn, 1994)

Checking positive definiteness of $\mathbf{A}^S$ is co-NP-hard.

Theorem (Jaulin and Henrion, 2005)

Checking whether there is a positive semidefinite matrix in $\mathbf{A}^S$ is a polynomial time problem.

Proof.

There is a positive semidefinite matrix in $\mathbf{A}^S$ iff $\overline{\lambda}_n(\mathbf{A}^S) \geq 0$.
So we can check it by semidefinite programming. $\square$

Sufficient Conditions

Theorem

1. $\mathbf{A}^S$ *is positive semidefinite if* $\lambda_n(A^c) \geq \rho(A^\Delta)$.
2. $\mathbf{A}^S$ *is positive definite if* $\lambda_n(A^c) > \rho(A^\Delta)$.
3. $\mathbf{A}^S$ *is positive definite if* A^c *is positive definite and* $\rho(|(A^c)^{-1}|A^\Delta) < 1$.

Proof.

1. $\mathbf{A}^S$ is positive semidefinite iff $\underline{\lambda}_n(\mathbf{A}^S) \geq 0$.
Now, employ the smallest eigenvalue set enclosure
$$\boldsymbol{\lambda}_n(\mathbf{A}^S) \subseteq [\lambda_n(A^c) - \rho(A^\Delta), \lambda_n(A^c) + \rho(A^\Delta)].$$
2. Analogous.
3. Use Beeck's sufficient condition for regularity of $\mathbf{A}$. $\square$

Next Section

Application: Convexity Testing

Theorem

A function $f : \mathbb{R}^n \mapsto \mathbb{R}$ is convex on $\mathbf{x} \in \mathbb{IR}^n$ iff its Hessian $\nabla^2 f(x)$ is positive semidefinite $\forall x \in int\,\mathbf{x}$.

Corollary

A function $f : \mathbb{R}^n \mapsto \mathbb{R}$ is convex on $\mathbf{x} \in \mathbb{IR}^n$ if $\nabla^2 f(\mathbf{x})$ is positive semidefinite.

Application: Convexity Testing

Example

Let

$$f(x, y, z) = x^3 + 2x^2y - xyz + 3yz^2 + 8y^2,$$

where $x \in \mathbf{x} = [2, 3]$, $y \in \mathbf{y} = [1, 2]$ and $z \in \mathbf{z} = [0, 1]$. The Hessian of f reads

$$\nabla^2 f(x, y, z) = \begin{pmatrix} 6x + 4y & 4x - z & -y \\ 4x - z & 16 & -x + 6z \\ -y & -x + 6z & 6y \end{pmatrix}$$

Evaluation the Hessian matrix by interval arithmetic results in

$$\nabla^2 f(\mathbf{x}, \mathbf{y}, \mathbf{z}) \subseteq \begin{pmatrix} [16, 26] & [7, 12] & -[1, 2] \\ [7, 12] & 16 & [-3, 4] \\ -[1, 2] & [-3, 4] & [6, 12] \end{pmatrix}$$

Now, both sufficient conditions for positive definiteness succeed. Thus, we can conclude that f si convex on the interval domain.

References

M. Hladík, D. Daney, and E. Tsigaridas.
Bounds on real eigenvalues and singular values of interval matrices.
SIAM J. Matrix Anal. Appl., 31(4):2116–2129, 2010.

M. Hladík, D. Daney, and E. P. Tsigaridas.
Characterizing and approximating eigenvalue sets of symmetric interval matrices.
Comput. Math. Appl., 62(8):3152–3163, 2011.

L. Jaulin and D. Henrion.
Contracting optimally an interval matrix without loosing any positive semi-definite matrix is a tractable problem.
Reliab. Comput., 11(1):1–17, 2005.

J. Rohn.
Positive definiteness and stability of interval matrices.
SIAM J. Matrix Anal. Appl., 15(1):175–184, 1994.

J. Rohn.
A handbook of results on interval linear problems.
Tech. Rep. 1163, Acad. of Sci. of the Czech Republic, Prague, 2012.
`http://uivtx.cs.cas.cz/~rohn/publist/!aahandbook.pdf`

Handling constraints rigorously

Interval Programming 7

Milan Hladík [1] Michal Černý [2]

[1] Faculty of Mathematics and Physics,
Charles University in Prague, Czech Republic
http://kam.mff.cuni.cz/~hladik/

[2] Faculty of Computer Science and Statistics,
University of Economics, Prague, Czech Republic
http://nb.vse.cz/~cernym/

Workshop on Interval Programming
7th International Conference of Iranian Operation Research Society
Semnan, Iran, May 12–13, 2014

Outline

1. Nonlinear Equations
2. Interval Newton method (square system)
3. Krawczyk method (square case)
4. More general constraints

Next Section

1. Nonlinear Equations
2. Interval Newton method (square system)
3. Krawczyk method (square case)
4. More general constraints

Nonlinear Equations

Problem Statement

Find all solutions to

$$f_j(x_1, \ldots, x_n) = 0, \quad j = 1, \ldots, j^*$$

inside the box $\mathbf{x}^0 \in \mathbb{IR}^n$.

Theorem (Zhu, 2005)

For a polynomial $p(x_1, \ldots, x_n)$, there is no algorithm solving

$$p(x_1, \ldots, x_n)^2 + \sum_{i=1}^{n} \sin^2(\pi x_i) = 0.$$

Proof.

From Matiyasevich's theorem solving the 10th Hilbert problem. □

Remark

Using the arithmetical operations only, the problem is decidable by Tarski's theorem (1951).

Next Section

Interval Newton method

Classical Newton method

...is an iterative method

$$x^{k+1} := x^k - \nabla f(x^k)^{-1} f(x^k), \quad k = 0, \ldots$$

Cons

- Can miss some solutions
- Not verified (Are we really close to the true solution?)

Interval Newton method – Stupid Intervalization

$$\mathbf{x}^{k+1} := \mathbf{x}^k - \nabla f(\mathbf{x}^k)^{-1} f(\mathbf{x}^k), \quad k = 0, \ldots$$

Interval Newton method – Good Intervalization

$$\begin{aligned} N(x^k, \mathbf{x}^k) &:= x^k - \nabla f(\mathbf{x}^k)^{-1} f(x^k), \\ \mathbf{x}^{k+1} &:= \mathbf{x}^k \cap N(\mathbf{x}^k), \qquad k = 0, \ldots \end{aligned}$$

Interval Newton method

Theorem (Moore, 1966)

If $x, x^0 \in \mathbf{x}$ and $f(x) = 0$, then $x \in N(x^0, \mathbf{x})$.

Proof.

By the Mean value theorem,

$$f_i(x) - f_i(x^0) = \nabla f_i(c_i)^T (x - x^0), \quad \forall i = 1, \dots, n.$$

If x is a root, we have

$$-f_i(x^0) = \nabla f_i(c_i)^T (x - x^0).$$

Define $A \in \mathbb{R}^{n \times n}$ such that its ith row is equal to $\nabla f_i(c_i)^T$. Hence

$$-f(x^0) = A(x - x^0),$$

from which

$$x = x^0 - A^{-1} f(x^0) \in x^0 - \nabla f(\mathbf{x})^{-1} f(x^0).$$

Notice, that this does not mean that there is $c \in \mathbf{x}$ such that

$$-f(x^0) = \nabla f(c)(x - x^0).$$

□

Interval Newton method

Theorem (Nickel, 1971)

If $\emptyset \neq N(x^0, \mathbf{x}) \subseteq \mathbf{x}$, then there is a unique root in $\mathbf{x}$ and $\nabla f(\mathbf{x})$ is regular.

Proof.

"Regularity." Easy.

"Existence." By Brouwer's fixed-point theorem.
[Any continuous mapping of a compact convex set into itself has a fixed point.]

"Uniqueness." If there are two roots $y_1 \neq y_2$ in $\mathbf{x}$, then by the Mean value theorem,

$$f(y_1) - f(y_2) = A(y_1 - y_2)$$

for some $A \in \nabla f(\mathbf{x})$;. Since $f(y_1) = f(y_2) = 0$, we get

$$A(y_1 - y_2) = 0$$

and by the nonsingularity of A, the roots are identical. □

Interval Newton method

Practical Implementation

Instead of

$$N(x^k, \mathbf{x}^k) := x^k - \nabla f(\mathbf{x}^k)^{-1} f(x^k)$$

let $N(x^k, \mathbf{x}^k)$ be an enclosure of the solution set (with respect to x) of

$$\nabla f(\mathbf{x})(x - x^0) = -f(x^0).$$

Extended Interval Arithmetic

So far

$$\frac{[12, 15]}{[-2, 3]} = (-\infty, \infty).$$

Now,

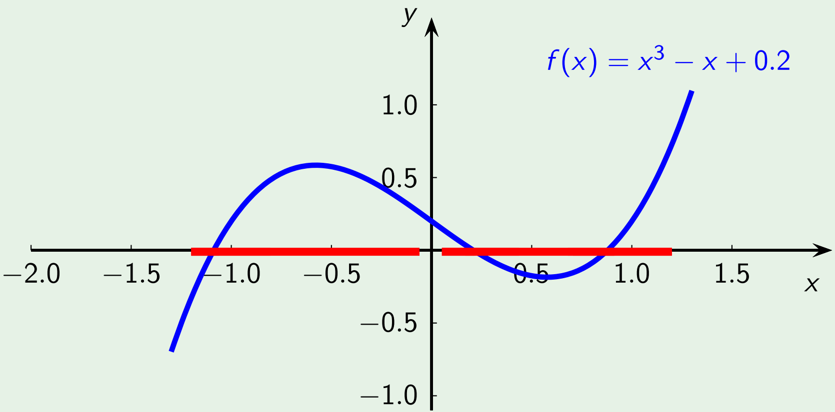
$$\mathbf{a}/\mathbf{b} := \{a/b \colon a \in \mathbf{a}, 0 \neq b \in \mathbf{b}\}.$$

So,

$$\frac{[12, 15]}{[-2, 3]} = (-\infty, -6] \cup [4, \infty).$$

Interval Newton method

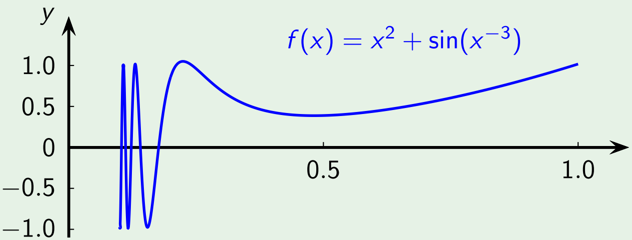
Example



In six iterations precision 10^{-11} (quadratic convergence).

Interval Newton method

Example (Moore, 1993)



All 318 roots of in the interval $[0.1, 1]$ found with accuracy 10^{-10}.
The left most root is contained in $[0.10003280626, 0.10003280628]$.

Summary

- $N(x^0, \mathbf{x})$ contains all solutions in $\mathbf{x}$
- If $\mathbf{x} \cap N(x^0, \mathbf{x}) = \emptyset$, then there is no root in $\mathbf{x}$
- If $\emptyset \neq N(x^0, \mathbf{x}) \subseteq \mathbf{x}$, then there is a unique root in $\mathbf{x}$

Next Section

Krawczyk method

Krawczyk operator

Let $x^0 \in \mathbf{x}$ and $C \in \mathbb{R}^{n\times n}$, usually $C \approx \nabla f(x^0)^{-1}$. Then

$$K(\mathbf{x}) := x^0 - Cf(x^0) + (I_n - C\nabla f(\mathbf{x}))(\mathbf{x} - x^0).$$

Theorem

Any root of $f(x)$ in $\mathbf{x}$ is included in $K(\mathbf{x})$.

Proof.

If x^1 is a root of $f(x)$, then it is a fixed point of

$$g(x) := x - Cf(x).$$

By the mean value theorem,

$$g(x^1) \in g(x^0) + \nabla g(\mathbf{x})(x^1 - x^0),$$

whence

$$\begin{aligned} x^1 \in g(\mathbf{x}) &\subseteq g(x^0) + \nabla g(\mathbf{x})(\mathbf{x} - x^0) \\ &= x^0 - Cf(x^0) + (I_n - C\nabla f(\mathbf{x}))(\mathbf{x} - x^0). \end{aligned}$$

$\square$

Krawczyk method

Theorem

If $K(\mathbf{x}) \subseteq \mathbf{x}$, then there is a root in $\mathbf{x}$.

Proof.

Recall

$$g(x) := x - Cf(x).$$

By the proof of the previous Theorem, $K(\mathbf{x}) \subseteq \mathbf{x}$ implies

$$g(\mathbf{x}) \subseteq \mathbf{x}.$$

Thus, there is a fixed point $x^0 \in \mathbf{x}$ of $g(x)$,

$$g(x^0) = x^0 - Cf(x^0) = x^0,$$

so x^0 is a root of $f(x)$. □

Krawczyk method

Theorem (Kahan, 1968)

If $K(\mathbf{x}) \subseteq \operatorname{int}\mathbf{x}$*, then there is a unique root in* $\mathbf{x}$ *and* $\nabla f(\mathbf{x})$ *is regular.*

Recall Theorem from Lecture 2

Let $\mathbf{x} \in \mathbb{IR}^n$ and $C \in \mathbb{R}^{n\times n}$. If

$$K(\mathbf{x}) = C\mathbf{b} + (I_n - C\mathbf{A})\mathbf{x} \subseteq \operatorname{int}\mathbf{x},$$

then C is nonsingular, $\mathbf{A}$ is regular, and $\Sigma \subseteq \mathbf{x}$.

Proof.

The inclusion $K(\mathbf{x}) \subseteq \operatorname{int}\mathbf{x}$ reads

$$-Cf(x^0) + (I_n - C\nabla f(\mathbf{x}))(\mathbf{x} - x^0) \subseteq \operatorname{int}(\mathbf{x} - x^0)$$

Apply the above Theorem for

$$\mathbf{b} := -f(x^0), \quad \mathbf{A} := \nabla f(\mathbf{x}), \quad \mathbf{x} := \mathbf{x} - x^0$$

We have that $\nabla f(\mathbf{x})$ is regular, which implies uniqueness. □

Exercise

Let $f(x, c) : \mathbb{R}^n \times \mathbb{R} \mapsto \mathbb{R}^n$ be a function depending on parameter c. Let $\mathbf{c} \in \mathbb{IR}$ and $\mathbf{x} \in \mathbb{IR}^n$. Give a condition under which there is a simple zero in $f(x, c)$ in $\mathbf{x}$ for each $c \in \mathbf{c}$.

Verification

Problem formulation

Given an approximate solution x^*, find $\mathbf{y} \in \mathbb{IR}^n$ such that there is a solution in $x^* + \mathbf{y}$.

ε-inflation method (Rump, 1983)

Put $\mathbf{y} := -Cf(x^0)$.

Repeat inflating $\mathbf{z} := [0.9, 1.1]\mathbf{y} + 10^{-20}[-1, 1]$ and updating

$$\mathbf{y} := -Cf(x^0) + (I_n - C\nabla f(\mathbf{x}))\mathbf{z}$$

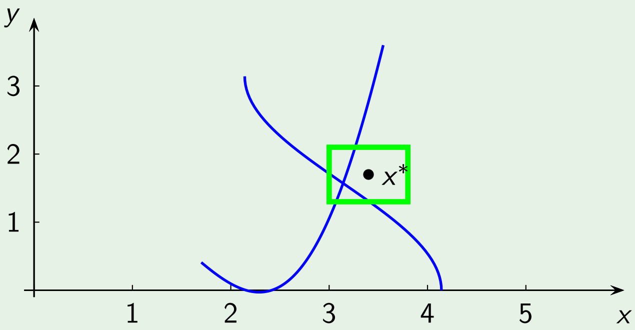
until $\mathbf{y} \subseteq \operatorname{int} \mathbf{z}$.

Then, there is a unique solution in $x^* + \mathbf{y}$.

Verification

Example

$$\pi^2(y - \pi/2) + 4x^2 \sin(x) = 0, \quad x - \pi - \cos(y) = 0.$$



- Approximate solution $x^* = (3.1415, 1.5708)^T$.
- Enclosing with accuracy 10^{-5} fails, but accuracy 10^{-4} succeeds.

Next Section

More general constraints

Constraints

- equations $h_i(x) = 0$, $i = 1, \ldots, I$
- inequalities $g_j(x) \leq 0$, $j = 1, \ldots, J$
- may be others, but not considered here
  ($\neq$, quantifications, logical operators, lexicographic orderings, . . .)

Problem

Denote by Σ the set of solutions in an initial box $\mathbf{x}^0 \in \mathbb{IR}^n$?

Problem: How to describe Σ?

Subpavings

Split $\mathbf{x}$ into a union of three sets of boxes such that

- the first set has boxes provably containing no solution
- the second set has boxes that provably consist of only solutions
- the third set has boxes which may or may not contain a solution

Subpaving Example

Example

$$x^2 + y^2 \leq 16,$$
$$x^2 + y^2 \geq 9$$

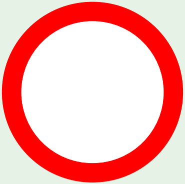


Figure: Exact solution set

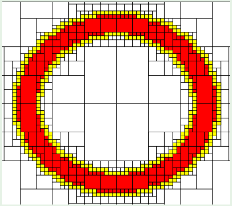


Figure: Subpaving approximation

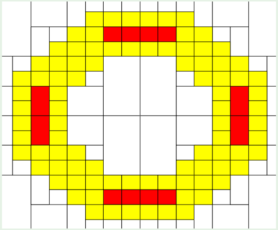
Subpaving Algorithm

Branch & Bound Scheme

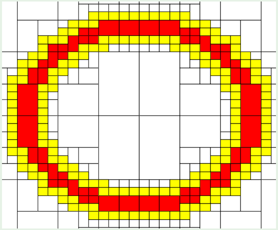
$\mathcal{L} := \{\mathbf{x}^0\}$, [set of boxes to process]
$\mathcal{S} := \emptyset$, [set of boxes with solutions only]
$\mathcal{N} := \emptyset$, [set of boxes with no solutions]
$\mathcal{B} := \emptyset$, [set of the undecidable boxes]
while $\mathcal{L} \neq \emptyset$ **do**
    choose $\mathbf{x} \in \mathcal{L}$ and remove $\mathbf{x}$ from $\mathcal{L}$
    if $\mathbf{x} \subseteq \Sigma$ **then**
        $\mathcal{S} := \mathcal{S} \cup \mathbf{x}$
    else if $\mathbf{x} \cap \Sigma = \emptyset$ **then**
        $\mathcal{N} := \mathcal{N} \cup \mathbf{x}$
    else if $x_i^{\Delta} < \varepsilon \; \forall i$ **then**
        $\mathcal{B} := \mathcal{B} \cup \mathbf{x}$
    else
        split $\mathbf{x}$ into sub-boxes and put them into $\mathcal{L}$
    end if
end while

Subpaving Example

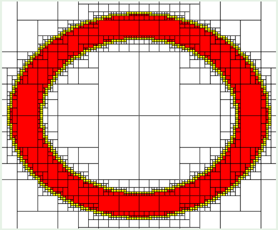
Example (thanks to Elif Garajová)



$\varepsilon = 1.0$
time: 0.952 s



$\varepsilon = 0.5$
time: 2.224 s



$\varepsilon = 0.125$
time: 9.966 s

Algorithm More in Detail

Test $\mathbf{x} \subseteq \Sigma$

- no equations and $\overline{g}_j(\mathbf{x}) \leq 0 \ \forall j$

Test $\mathbf{x} \cap \Sigma = \emptyset$

- $0 \notin h_i(\mathbf{x})$ for some i
- $\underline{g}_j(\mathbf{x}) > 0$ for some j

Also very important

- Which box to choose (data structure fo $\mathcal{L}$)?
- How to divide the box? (which coordinate, which place, how many sub-boxex)

Improvement

Contraction of $\mathbf{x}$ such that no solution is missed (and do not use $\mathcal{B}$).

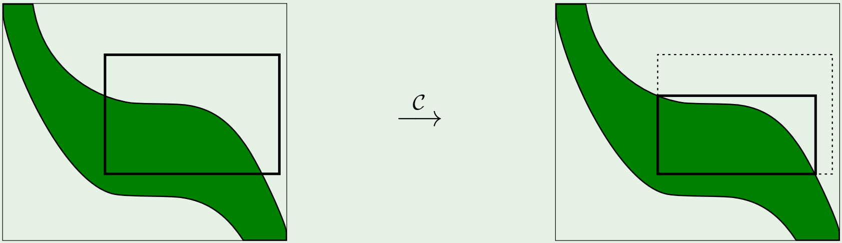
Contractors

Definition

Contractor A function $\mathcal{C} : \mathbb{IR}^n \to \mathbb{IR}^n$ is called a contractor if $\forall \mathbf{x} \in \mathbb{IR}^n$ we have

- $\mathcal{C}(\mathbf{x}) \subseteq \mathbf{x}$
- $\mathcal{C}(\mathbf{x}) \cap \Sigma = \mathbf{x} \cap \Sigma$

Example



A Simple Contractor – Constraint Propagation

Example

Consider the constraint

$$x + yz = 7, \quad x \in [0, 3],\ y \in [3, 5],\ z \in [2, 4].$$

- Express x

$$x = 7 - yz \in 7 - [3, 5][2, 4] = [-13, 1].$$

  Thus, the domain for x is $[0, 3] \cap [-13, 1] = [0, 1]$.

- Express y

$$y = (7 - x)/z \in (7 - [0, 1])/[2, 4] = [1.5,\ 3.5].$$

  Thus, the domain for y is $[3, 5] \cap [1.5,\ 3.5] = [3,\ 3.5]$.

- Express z

$$z = (7 - x)/y \in (7 - [0, 1])/[3,\ 3.5] = [\tfrac{12}{7}, \tfrac{7}{3}].$$

  Thus, the domain for z is $[2, 4] \cap [\frac{12}{7}, \frac{7}{3}] = [2, \frac{7}{3}]$.

No further propagation needed as each variable appears just once.

A Simple Contractor – Constraint Propagation

Example

Consider the constraint

$$e^x - xyz = 10, \quad x \in \mathbf{x} = [4,5],\ y \in \mathbf{y} = [3,4],\ z \in \mathbf{z} = [2,3].$$

Contractions of domains:

iteration	$\mathbf{x}$	$\mathbf{y}$	$\mathbf{z}$
1	[4, 4.2485]	[3.4991, 4]	[2.6243, 3]
2	[4, 4.1106]	[3.6165, 4]	[2.7124, 3]
3	[4, 4.0831]	[3.6409, 4]	[2.7306, 3]
4	[4, 4.0775]	[3.6458, 4]	[2.7344, 3]
5	[4, 4.0764]	[3.6469, 4]	[2.7351, 3]
$\vdots$			
∞	[4, 4.0761]	[3.6471, 4]	[2.7353, 3]

Multiple appearance of x causes infinite convergence.

2B-consistency

Definition (2B-consistency)

A set of constraints $c_k(x)$, $k = 1, \dots, K$, on a box $\mathbf{x}^0 \in \mathbb{IR}^n$ is 2B-consistent if for each $k \in \{1, \dots, K\}$ and each $i \in \{1, \dots, n\}$ there are some $x, x' \in \mathbf{x}^0$ such that $x_i = \underline{x}_i^0$, $x_i' = \overline{x}_i^0$, and conditions $c_k(x)$ and $c_k(x')$ are valid.

Remark

- Constraint propagation tries to approach 2B-consistency.
- Drawback: 2B-consistency looks at constraints separately.

Free Constraint Solving Software

- *Alias* (by Jean-Pierre Merlet, COPRIN team),
  A C++ library for system solving, with Maple interface,
  `http://www-sop.inria.fr/coprin/logiciels/ALIAS/ALIAS-C++/ALIAS-C++.html`
- *Quimper* (by Gill Chabert and Luc Jaulin),
  written in an interval C++ library IBEX,
  a language for interval modelling and handling constraints,
  `http://www.emn.fr/z-info/ibex`
- *RealPaver* (by L. Granvilliers and F. Benhamou),
  a C++ package for modeling and solving nonlinear and nonconvex constraint satisfaction problems,
  `http://pagesperso.lina.univ-nantes.fr/info/perso/permanents/granvil/realpaver`
- *RSolver* (by Stefan Ratschan),
  solver for quantified constraints over the real numbers,
  implemented in the programming language OCaml,
  `http://rsolver.sourceforge.net/`

References

G. Alefeld and J. Herzberger.
Introduction to Interval Computations.
Academic Press, New York, 1983.

F. Benhamou and L. Granvilliers.
Continuous and interval constraints.
In *Handbook of Constraint Programming*, chap. 16, 571–603. Elsevier, 2006.

G. Chabert and L. Jaulin.
Contractor programming.
Artif. Intell., 173(11):1079-1100, 2009.

F. Goualard and C. Jermann.
A reinforcement learning approach to interval constraint propagation.
Constraints, 13(1):206–226, 2008.

L. Jaulin, M. Kieffer, O. Didrit, and É. Walter.
Applied Interval Analysis.
Springer, London, 2001.

S. M. Rump.
Verification methods: Rigorous results using floating-point arithmetic.
Acta Numer., 19:287–449, 2010.

Global Optimization by Interval Techniques

Interval Programming 8

Milan Hladík [1] Michal Černý [2]

[1] Faculty of Mathematics and Physics,
Charles University in Prague, Czech Republic
http://kam.mff.cuni.cz/~hladik/

[2] Faculty of Computer Science and Statistics,
University of Economics, Prague, Czech Republic
http://nb.vse.cz/~cernym/

Workshop on Interval Programming
7th International Conference of Iranian Operation Research Society
Semnan, Iran, May 12–13, 2014

Outline

Next Section

Formulation and Complexity

Global optimization problem

Compute global (not just local!) optima to

$$\min f(x) \quad \text{subject to} \quad g(x) \leq 0, \; h(x) = 0, \; x \in \mathbf{x}^0,$$

where $\mathbf{x}^0 \in \mathbb{IR}^n$ is an initial box.

Theorem (Zhu, 2005)

There is no algorithm solving global optimization problems using operations $+, \times, \sin$.

Proof.

From Matiyasevich's theorem solving the 10th Hilbert problem. □

Remark

Using the arithmetical operations only, the problem is decidable by Tarski's theorem (1951).

Interval Approach to Global Optimization

Branch & Bound Scheme

$\mathcal{L} := \{\mathbf{x}^0\}$, [set of boxes]
$c^* := \infty$, [upper bound on the minimal value]
while $\mathcal{L} \neq \emptyset$ **do**
    choose $\mathbf{x} \in \mathcal{L}$ and remove $\mathbf{x}$ from $\mathcal{L}$,
    contract $\mathbf{x}$,
    find a feasible point $x \in \mathbf{x}$ and update c^*,
    if $\max_i x_i^{\Delta} > \varepsilon$ **then**
        split $\mathbf{x}$ into sub-boxes and put them into $\mathcal{L}$,
    else
        give $\mathbf{x}$ to the output boxes,
    end if
end while

It is a rigorous method to enclose all global minima in a set of boxes.

Box Selection

Which box to choose?

- the oldest one
- the one with the largest edge, i.e., for which $\max_i x_i^\Delta$ is maximal
- the one with minimal $\underline{f}(\mathbf{x})$.

Division Directions

How to divide the box?

1. Take the widest edge of $\mathbf{x}$, that is
$$k := \arg \max_{i=1,\dots,n} x_i^{\Delta}.$$
2. (Walster, 1992) Choose a coordinate in which f varies possibly mostly
$$k := \arg \max_{i=1,\dots,n} f'_{x_i}(\mathbf{x})^{\Delta} x_i^{\Delta}.$$
3. (Ratz, 1992) It is similar to the previous one, but uses
$$k := \arg \max_{i=1,\dots,n} (f'_{x_i}(\mathbf{x})\mathbf{x}_i)^{\Delta}.$$

Remarks

- by Ratschek & Rokne (2009) there is no best strategy for splitting
- combine several of them
- the splitting strategy influences the overall performance

Contracting and Pruning

Aim

Shrink $\mathbf{x}$ to a smaller box (or completely remove) such that no global minimum is removed.

Simple Techniques

- if $0 \notin h_i(\mathbf{x})$ for some i, then remove $\mathbf{x}$
- if $0 < g_j(\mathbf{x})$ for some j, then remove $\mathbf{x}$
- if $0 < f'_{x_i}(\mathbf{x})$ for some i, then fix $\mathbf{x}_i := \underline{x}_i$
- if $0 > f'_{x_i}(\mathbf{x})$ for some i, then fix $\mathbf{x}_i := \overline{x}_i$

Optimality Conditions

- employ the Fritz–John (or the Karush–Kuhn–Tucker) conditions

$$u_0 \nabla f(x) + u^T \nabla h(x) + v^T \nabla g(x) = 0, \quad v \geq 0,$$
$$h(x) = 0, \quad g(x) \leq 0, \quad v_\ell g_\ell(x) = 0 \ \forall \ell, \quad \|(u_0, u, v)\| = 1.$$

- solve by the Interval Newton method

Contracting and Pruning

Inside the Feasible Region

Suppose there are no equality constraints and $g_j(\mathbf{x}) < 0 \ \forall j$.

- (monotonicity test) if $0 \notin f'_{x_i}(\mathbf{x})$ for some i, then remove $\mathbf{x}$
- apply the Interval Newton method to the additional constraint $\nabla f(x) = 0$
- (nonconvexity test) if the interval Hessian $\nabla^2 f(\mathbf{x})$ contains no positive semidefinite matrix, then remove $\mathbf{x}$

Next Section

Feasibility Test

Aim

Find a feasible point x^*, and update $c^* := \min(c^*, f(x^*))$.

Why?

- Remove boxes with $\underline{f}(\mathbf{x}) > c^*$.
- We can include $f(x) \leq c^*$ to the constraints.

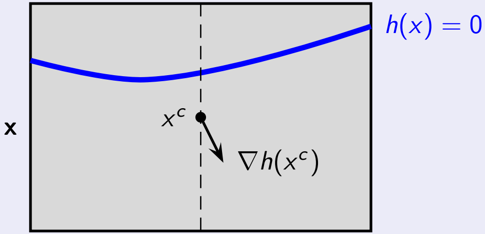
No equations

If no equality constraints, take, e.g., $x^* := x^c$ provided $g(x^c) \leq 0$.

Feasibility Test

For equations

- if k equality constraints, fix $n-k$ variables $x_i := x_i^c$ and solve system of equations by the interval Newton method
- if $k=1$, fix the variables corresponding to the smallest absolute values in $\nabla h(x^c)$



- If $k>1$, transform the matrix $\nabla h(x^c)$ to REF by using a complete pivoting, and fix components corresponding to the right most columns

Lower Bounds

Aim

Given a box $\mathbf{x} \in \mathbb{IR}^n$, determine a lower bound to $\underline{f}(\mathbf{x})$.

Why?

- if $\underline{f}(\mathbf{x}) > c^*$, we can remove $\mathbf{x}$
- minimum over all boxes gives a lower bound on the optimal value

Methods

- interval arithmetic
- mean value form
- slope form
- Lipschitz constant approach
- αBB algorithm
- ...

Next Section

αBB algorithm

Special cases: Bilinear terms

For every $y \in \mathbf{y} \in \mathbb{IR}$ and $z \in \mathbf{z} \in \mathbb{IR}$ we have

$$yz \geq \max\{\underline{y}z + \underline{z}y - \underline{y}\underline{z},\ \overline{y}z + \overline{z}y - \overline{y}\overline{z}\}.$$

General case: Convex underestimators for $f(x)$

Construct a function $g : \mathbb{R}^n \mapsto \mathbb{R}$ satisfying:

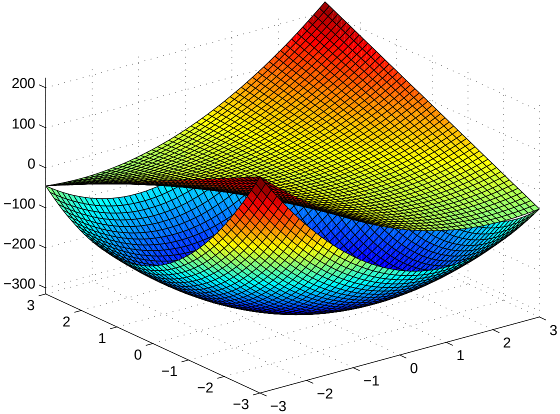
- $f(x) \geq g(x)$ for every $x \in \mathbf{x}$,
- $g(x)$ is convex on $x \in \mathbf{x}$.

αBB algorithm (Androulakis, Maranas & Floudas, 1995)

Consider an underestimator $g(x) \leq f(x)$ in the form

$$g(x) := f(x) - \sum_{i=1}^{n} \alpha_i(\overline{x}_i - x_i)(x_i - \underline{x}_i), \quad \text{where } \alpha_i \geq 0\ \forall i.$$

Illustration



Function $f(x)$ and its convex underestimator $g(x)$.

Computation of α

Idea

The Hessian of $g(x)$ reads

$$\nabla^2 g(x) = \nabla^2 f(x) + 2\,\mathrm{diag}(\alpha).$$

Choose α large enough to ensure positive semidefinitness of the Hessian of

$$g(x) := f(x) - \sum_{i=1}^{n} \alpha_i(\overline{x}_i - x_i)(x_i - \underline{x}_i).$$

Interval Hessian matrix

Let $\mathbf{H}$ be an interval matrix enclosing the image of $\nabla^2 f(x)$ over $x \in \mathbf{x}$:

$$\frac{\partial^2}{\partial x_i \partial x_j} f(x) \in \mathbf{h}_{ij} = [\underline{h}_{ij}, \overline{h}_{ij}], \quad \forall x \in \mathbf{x}.$$

Remarks

- Checking positive semidefiniteness of each $H \in \mathbf{H}$ is co-NP-hard.
- Various enclosures for eigenvalues of $H \in \mathbf{H}$.
- Scaled Gerschgorin method enables to express α_i-s.

Computation of α

Scaled Gerschgorin method for α

$$\alpha_i := \max\left\{0, -\tfrac{1}{2}\left(\underline{h}_{ii} - \textstyle\sum_{j\neq i} |\mathbf{h}_{ij}| d_j/d_i\right)\right\}, \quad i = 1, \ldots, n,$$

where $|\mathbf{h}_{ij}| = \max\left\{|\underline{h}_{ij}|, |\overline{h}_{ij}|\right\}$.

- To reflect the range of the variable domains, use $d := \overline{x} - \underline{x}$.

Theorem (H., 2014)

The choice $d := \overline{x} - \underline{x}$ is optimal (i.e., it minimizes the maximum separation distance between $f(x)$ and $g(x)$) if

$$\underline{h}_{ii} d_i - \textstyle\sum_{j\neq i} |\mathbf{h}_{ij}| d_j \leq 0, \quad \forall i = 1, \ldots, n.$$

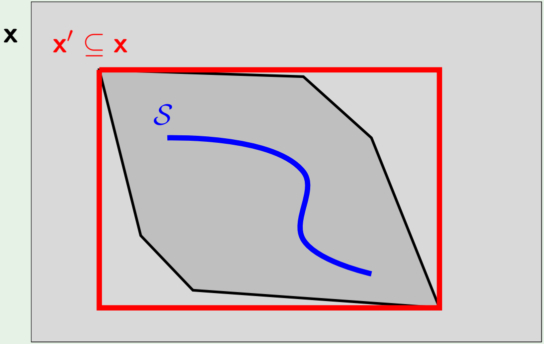
Next Section

Linearization

Interval linear programming approach

- linearize constraints,
- compute new bounds and iterate.

Example



Mean value form

Theorem

Let $f : \mathbb{R}^n \mapsto \mathbb{R}$, $\mathbf{x} \in \mathbb{IR}^n$ *and* $a \in \mathbf{x}$. *Then*

$$f(\mathbf{x}) \subseteq f(a) + \nabla f(\mathbf{x})^T(\mathbf{x} - a),$$

Proof.

By the mean value theorem, for any $x \in \mathbf{x}$ there is $c \in \mathbf{x}$ such that

$$f(x) = f(a) + \nabla f(c)^T(x - a) \in f(a) + \nabla f(\mathbf{x})^T(\mathbf{x} - a). \qquad \square$$

Improvements

- successive mean value form

$$\begin{aligned} f(\mathbf{x}) \subseteq f(a) &+ f'_{x_1}(\mathbf{x}_1, a_2, \ldots, a_n)(\mathbf{x}_1 - a_1) \\ &+ f'_{x_2}(\mathbf{x}_1, \mathbf{x}_2, a_3 \ldots, a_n)(\mathbf{x}_2 - a_2) + \ldots \\ &+ f'_{x_n}(\mathbf{x}_1, \ldots, \mathbf{x}_{n-1}, \mathbf{x}_n)(\mathbf{x}_n - a_n). \end{aligned}$$

- replace derivatives by slopes

Slopes

Slope form enclosure

$$f(\mathbf{x}) \subseteq f(a) + S(\mathbf{x}, a)(\mathbf{x} - a),$$

where $a \in \mathbf{x}$ and

$$S(x, a) := \begin{cases} \frac{f(x)-f(a)}{x-a} & \text{if } x \neq a, \\ f'(x) & \text{otherwise.} \end{cases}$$

Remarks

- Slopes can be replaced by derivatives, but slopes are tighter.
- Slopes can be computed in a similar way as derivatives.

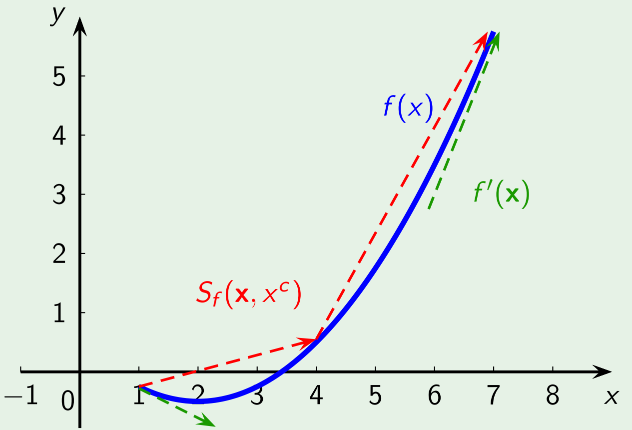
function	its slope $S(x, a)$
x	1
$f(x) \pm g(x)$	$S_f(x, a) \pm S_g(x, a)$
$f(x) \cdot g(x)$	$S_f(x, a)g(a) + f(x)S_g(x, a)$
$e^{f(x)}$	$e^{f(x)} S_f(x, a)$

Slopes

Example

$$f(x) = \tfrac{1}{4}x^2 - x + \tfrac{1}{2}, \qquad \mathbf{x} = [1, 7].$$

$$f'(\mathbf{x}) = [-\tfrac{1}{2}, \tfrac{5}{2}], \qquad S_f(\mathbf{x}, x^c) = [\tfrac{1}{4}, \tfrac{7}{4}].$$



Linearization

Interval linearization

Let $x^0 \in \mathbf{x}$. Suppose that a for some interval matrices $\mathbf{A}$ and $\mathbf{B}$ we have

$$h(x) \subseteq \mathbf{A}(x - x^0) + h(x^0), \quad \forall x \in \mathbf{x}$$
$$g(x) \subseteq \mathbf{B}(x - x^0) + g(x^0), \quad \forall x \in \mathbf{x},$$

e.g., by the mean value form, slopes, ...

Interval linear programming formulation

Now, the set $\mathcal{S}$ is enclosed by

$$\mathbf{A}(x - x^0) + h(x^0) = 0,$$
$$\mathbf{B}(x - x^0) + g(x^0) \leq 0.$$

What remains to do

- Solve the interval linear program
- Choose $x^0 \in \mathbf{x}$

Case $x^0 := \underline{x}$

Let $x^0 := \underline{x}$. Since $x - \underline{x}$ is non-negative, the solution set to

$$\mathbf{A}(x - x^0) + h(x^0) = 0,$$
$$\mathbf{B}(x - x^0) + g(x^0) \leq 0,$$

is described by

$$\underline{A}x \leq \underline{A}\,\underline{x} - h(\underline{x}), \quad \overline{A}x \geq \overline{A}\underline{x} - h(\underline{x}),$$
$$\underline{B}x \leq \underline{B}\,\underline{x} - g(\underline{x}).$$

- Similarly if x^0 is any other vertex of $\mathbf{x}$

Linearization

General case

Let $x^0 \in \mathbf{x}$. The solution set to

$$\mathbf{A}(x - x^0) + h(x^0) = 0,$$
$$\mathbf{B}(x - x^0) + g(x^0) \leq 0,$$

is described by

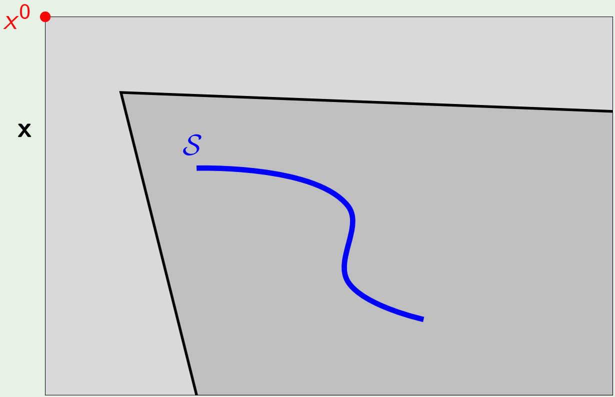
$$|A^c(x - x^0) + h(x^0)| \leq A^\Delta |x - x^0|,$$
$$B^c(x - x^0) + g(x^0) \ \leq B^\Delta |x - x^0|.$$

- Non-linear description due to the absolute values.
- How to get rid of them?
- Estimate from above by a linear function: $|x - x^0| \leq \alpha(x - x^0) + \beta$. (Easy to find the best upper linear estimation.)

Linearization

Example

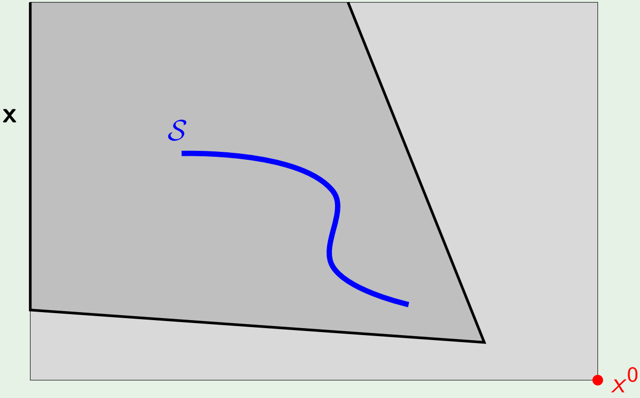
Typical situation when choosing x^0 to be vertex:



Linearization

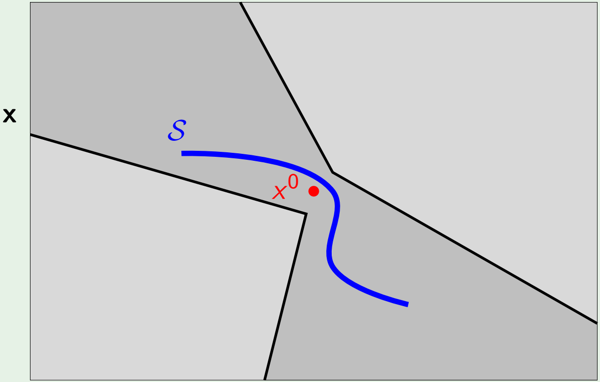
Example

Typical situation when choosing x^0 to be the opposite vertex:



Example

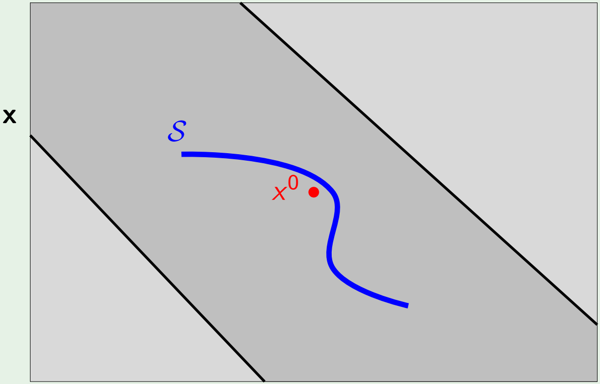
Typical situation when choosing $x^0 = x^c$:



Linearization

Example

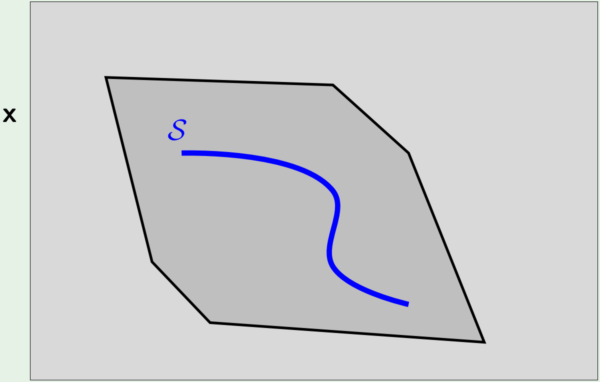
Typical situation when choosing $x^0 = x^c$ (after linearization):



Linearization

Example

Typical situation when choosing all of them:



Next Section

Examples

Example (The COPRIN examples, 2007, precision $\sim 10^{-6}$)

- tf12 (origin: COCONUT, solutions: 1, computation time: 60 s)

$$\begin{aligned} \min\ & x_1 + \tfrac{1}{2}x_2 + \tfrac{1}{3}x_3 \\ \text{s.t.}\ & -x_1 - \tfrac{i}{m}x_2 - (\tfrac{i}{m})^2 x_3 + \tan(\tfrac{i}{m}) \le 0, \quad i = 1, \ldots, m\ (m = 101). \end{aligned}$$

- o32 (origin: COCONUT, solutions: 1, computation time: 2.04 s)

$$\min\ 37.293239x_1 + 0.8356891x_5x_1 + 5.3578547x_3^2 - 40792.141$$

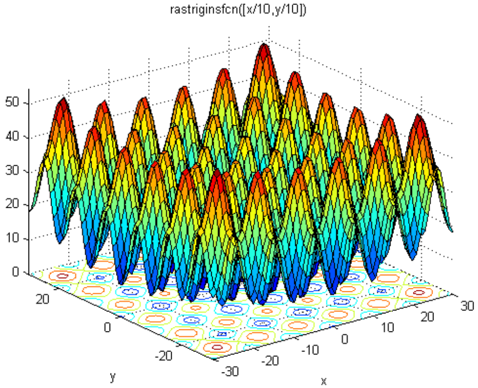
$$\begin{aligned} \text{s.t.}\quad -0.0022053x_3x_5 + 0.0056858x_2x_5 + 0.0006262x_1x_4 - 6.665593 &\le 0, \\ -0.0022053x_3x_5 - 0.0056858x_2x_5 - 0.0006262x_1x_4 - 85.334407 &\le 0, \\ 0.0071317x_2x_5 + 0.0021813x_3^2 + 0.0029955x_1x_2 - 29.48751 &\le 0, \\ -0.0071317x_2x_5 - 0.0021813x_3^2 - 0.0029955x_1x_2 + 9.48751 &\le 0, \\ 0.0047026x_3x_5 + 0.0019085x_3x_4 + 0.0012547x_1x_3 - 15.699039 &\le 0, \\ -0.0047026x_3x_5 - 0.0019085x_3x_4 - 0.0012547x_1x_3 + 10.699039 &\le 0. \end{aligned}$$

- Rastrigin (origin: Myatt (2004), solutions: 1 (approx.), time: 2.07 s)

$$\min\ 10n + \textstyle\sum_{j=1}^{n}(x_j - 1)^2 - 10\cos(2\pi(x_j - 1)),$$

where $n = 10$, $x_j \in [-5.12, 5.12]$.

Examples



One of the Rastrigin functions.

Software

Rigorous global optimization software

- *GlobSol* (by R. Baker Kearfott), written in Fortran 95, open-source exist conversions from AMPL and GAMS representations, `http://interval.louisiana.edu/`
- *COCONUT Environment*, open-source C++ classes `http://www.mat.univie.ac.at/~coconut/coconut-environment/`
- *GLOBAL* (by Tibor Csendes), for Matlab / Intlab, free for academic purposes `http://www.inf.u-szeged.hu/~csendes/linkek_en.html`
- *PROFIL / BIAS* (by O. Knüppel et al.), free C++ class `http://www.ti3.tu-harburg.de/Software/PROFILEnglisch.html`

See also

- *C.A. Floudas* (`http://titan.princeton.edu/tools/`)
- *A. Neumaier* (`http://www.mat.univie.ac.at/~neum/glopt.html`)

References

C. A. Floudas.
Deterministic Global Optimization. Theory, Methods and Applications.
Kluwer, Dordrecht, 2000.

C. A. Floudas and P. M. Pardalos, editors.
Encyclopedia of Optimization. 2nd ed.
Springer, New York, 2009.

E. R. Hansen and G. W. Walster.
Global Optimization Using Interval Analysis.
Marcel Dekker, New York, second edition, 2004.

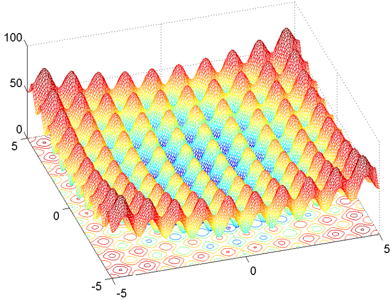
R. B. Kearfott.
Rigorous Global Search: Continuous Problems.
Kluwer, Dordrecht, 1996.

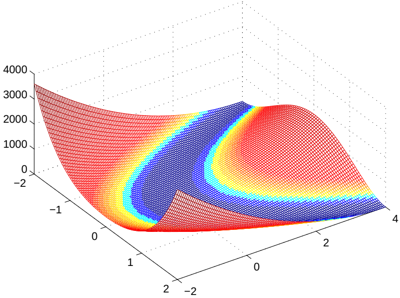
A. Neumaier.
Complete Search in Continuous Global Optimization and Constraint Satisfaction.
Acta Numerica, 13:271–369, 2004.

Next Section

Minimization of $f(x_1, \ldots, x_n)$ — usual testing functions:

- Rastrigin's function $20 + x_1^2 + x_2^2 - 10(\cos 2\pi x_1 + \cos 2\pi x_2)$
- Banana (Rosenbrock) function $(1 - x_1)^2 + 100(x_2 - x_1^2)^2$
- hidden minimum function





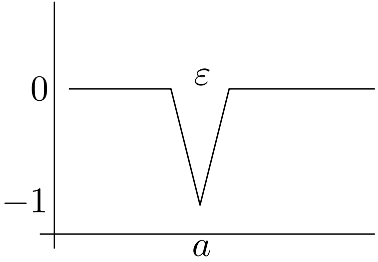
AlgoIss: Computation models

How to represent the function f?

- Special optimization problems: analytical expression (e.g. $\frac{1}{2}x^{\mathrm{T}}Hx + c^{\mathrm{T}}x + \alpha$ for quadratic programming)
- A general function: oracle model — there is a blackbox f s.t. on a query x it returns $f(x)$
- In the oracle model: complexity often measured by the number of oracle queries
- Problem: How to measure the size of input?

AlgoIss: Example

Hidden-minimum function:



Theorem

There is no finite upper bound on the number of steps for an optimization algorithm to locate the global minimum.

Proof.

For a finite number of testing points, ε can be chosen so small so that the algorithm cannot distinguish between the zero function and the hidden-minimum function. □

Algolss: Further problems

Approximate solution

- The minimizer may be irrational — a problem with representation by a Turing machine
- Example: $\min\{x^{\mathrm{T}}Ax : x^{\mathrm{T}}x \leq 1\} = \lambda_{\mathsf{min}}(A)$ for A negative definite. The number $\lambda_{\mathsf{min}}(A)$ is often irrational even for a rational matrix A; the minimizer is its corresponding eigenvector.
- Possible solution: real-number computation model. [Drawback: we lose finite-time convergence of many weakly polynomial methods, such as the Ellipsoid Method or IPMs.]
- Another solution: find approximate optima.

Algolss: Approximate minimum

Definition

A point x is an ε-approximate minimum is $f(x) - f(x^*) \leq \varepsilon$, where x^* is the true minimizer.

Remarks.

- Weak definition — though $f(x)$ is close to $f(x^*)$, the distance $\|x - x^*\|_\infty$ can still be extremely large. So sometimes one also adds the requirement: "… and $\|x - x^*\|_\infty \leq \varepsilon$".
- This problem vanishes for Lipschitz functions.

Theorem

If f is L-Lipschitz, then the ε-approximate global minimum of f over a unit cube can be found in $\approx (\frac{1}{2}\frac{L}{\varepsilon})^n$ steps and not faster.

AlgoIss: Approximate minimum (contd.)

Theorem

If f is L-Lipschitz, then the ε-approximate global minimum of f over the unit cube $[0, e]$ can be found in $\approx (\frac{1}{2}\frac{L}{\varepsilon})^n$ steps and not faster.

Proof idea.

- **Upper bound.** Cover the cube $[0, e]$ by a regular n-dimensional grid with distance $\frac{2\varepsilon}{L}$ between neighbor points and evaluate f in each grid point; then take the minimum. Then $\|x - x^*\|_\infty \leq \frac{\varepsilon}{L}$ for some grid point x and $f(x) - f(x^*) \leq L\|x - x^*\| \leq \varepsilon$.
- **Lower bound.** Let $\varepsilon' > \varepsilon$. Define a hidden-minimum function f_v

$$f_v(x) = \begin{cases} 0 & \text{if} \|v - x\|_\infty \geq \frac{\varepsilon'}{L}, \\ L\|v - x\|_\infty - \varepsilon' & \text{if} \|v - x\|_\infty < \frac{\varepsilon'}{L}. \end{cases}$$

  Then, v is the minimizer. Idea: any algorithm that uses less than $(\frac{1}{2}\frac{L}{\varepsilon})^n$ oracle queries to f leaves some region of $[0, e]$ "uninspected"; so we can place v into that region. Thus the algorithm cannot find it and it cannot distinguish between f_v and the zero function.

Algolss: Summary and special cases

- In general, global minimization is nonrecursive (to recall: an algorithm for the question "$\min f(x_1, \ldots, x_n) \leq^? 0$" would solve Hilbert's Tenth Problem. This holds true even for the case $n = 1$.
- So we must inspect the general problem by subcases.

Polynomials

- Recursive by Tarski's quantifier elimination, but extremely slow.
- Idea: the question "does a given polynomial $p(x_1, \ldots, x_n)$ attain a value f_0?" can be written as an arithmetical formula

$$(\exists x_1) \cdots (\exists x_n) p(x_1, \ldots, x_n) = f_0 \\ \& \ \underline{x}_1 \leq x_1 \leq \overline{x_1} \ \& \ \cdots \ \& \ \underline{x}_n \leq x_n \leq \overline{x_n}. \quad (1)$$

- Tarski proved that Theory of Real Closed Fields is decidable. So in principle, we can enumerate all proofs until we find a proof of (1) or its negation. This proves recursivity.

Algolss: Summary and special cases

Convex optimization

- "Nice" case: local minimum = global minimum
- However, in general nothing can be proved without additional assumptions
- ε-approximate minimization of a differentiable convex L-Lipschitz function can be done in time $O(n^2(\log n + \log \frac{L}{\varepsilon}))$ in the oracle model

Further problems we must face: Example

- Optimization under quadratic constraints $x^{\mathrm{T}} Hx + c^{\mathrm{T}} x \leq \gamma$ — is the feasibility problem in NP?
- Problem: a feasible point cannot be used as an NP-witness, since it can happen that bit-size of (a unique) feasible point is exponential in bitsize of H, c, γ.

Algolss: Summary and special cases

Quadratic programming $\min x^{\mathrm{T}} Hx + c^{\mathrm{T}} x$ s.t. $Ax \leq b$

- The convex case (H psd): polynomial time
- A single eigenvalue of H is negative: NP-hard
- H general: optimization over an *ellipsoid*: polynomial time
- H general: optimization over a *simplex*: NP-hard
- H general: "is given x a local minimum of $x^{\mathrm{T}} Hx + c^{\mathrm{T}} x$?": NP-hard
- further results: the form $\min x^{\mathrm{T}} Hx + c^{\mathrm{T}} x$ s.t. $x \in \mathbf{x}$ with H nsd of a fixed rand: polynomial time