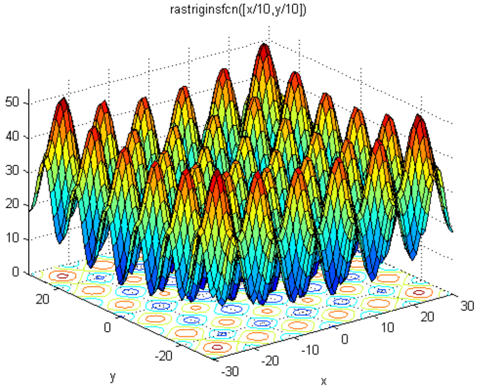
Solving Global Optimization Problems by Interval Computation

Milan Hladík

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

Seminar on Computational Aspects of Optimisation
April 27, 2015

Example



One of the Rastrigin functions.

Global Optimization Questions

Questions

- Can we find global minimum?
- Can we prove that the found solution is optimal?
- Can we prove uniqueness?

Answer

- **Yes** (under certain assumption) by using *Interval Computations*.

Interval Computations

Notation

An interval matrix

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

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

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

or at least its tight interval enclosure.

Interval Arithmetic

Interval Arithmetic (proper rounding used when implemented)

For arithmetical operations $(+, -, \cdot, \div)$, their images are readily computed

$$\begin{aligned}
\mathbf{a} + \mathbf{b} &= [\underline{a} + \underline{b}, \overline{a} + \overline{b}], \\
\mathbf{a} - \mathbf{b} &= [\underline{a} - \overline{b}, \overline{a} - \underline{b}], \\
\mathbf{a} \cdot \mathbf{b} &= [\min(\underline{a}\underline{b}, \underline{a}\overline{b}, \overline{a}\underline{b}, \overline{a}\overline{b}), \max(\underline{a}\underline{b}, \underline{a}\overline{b}, \overline{a}\underline{b}, \overline{a}\overline{b})], \\
\mathbf{a} \div \mathbf{b} &= [\min(\underline{a} \div \underline{b}, \underline{a} \div \overline{b}, \overline{a} \div \underline{b}, \overline{a} \div \overline{b}), \max(\underline{a} \div \underline{b}, \underline{a} \div \overline{b}, \overline{a} \div \underline{b}, \overline{a} \div \overline{b})].
\end{aligned}$$

Some basic functions $\mathbf{x}^2$, $\exp(\mathbf{x})$, $\sin(\mathbf{x})$, ..., too.

Can we evaluate every arithmetical expression on intervals?
Yes, but with overestimation in general due to dependencies.

Example (Evaluate $f(x) = x^2 - x$ on $\mathbf{x} = [-1, 2]$)

$$\begin{aligned}
\mathbf{x}^2 - \mathbf{x} = [-1, 2]^2 - [-1, 2] = [-2, 5], \\
\mathbf{x}(\mathbf{x} - 1) = [-1, 2]([-1, 2] - 1) = [-4, 2], \\
(\mathbf{x} - \tfrac{1}{2})^2 - \tfrac{1}{4} = ([-1, 2] - \tfrac{1}{2})^2 - \tfrac{1}{4} = [-\tfrac{1}{4}, 2].
\end{aligned}$$

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

Interval Linear Equations

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

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

Theorem (Oettli–Prager, 1964)

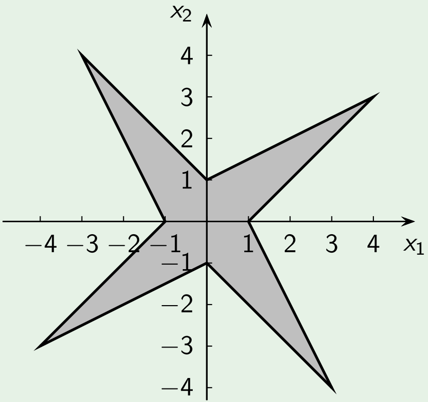
The solution set Σ is a non-convex polyhedral set described by

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

Interval Linear Equations

Example (Barth & Nuding, 1974))

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



Interval Linear Equations

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

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

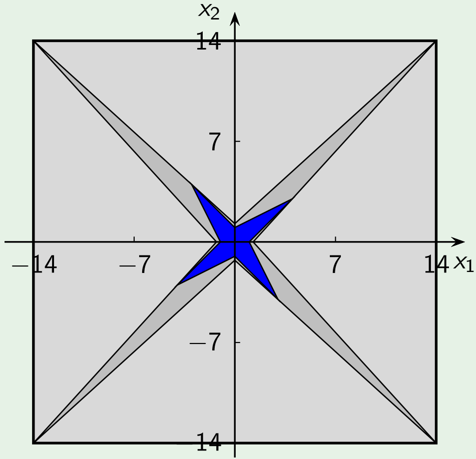
Remark

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

Interval Linear Equations

Example (Barth & Nuding, 1974))

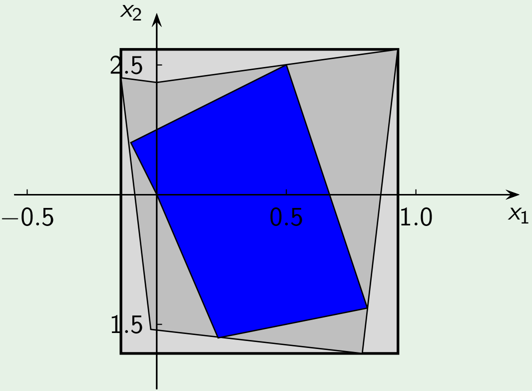
$$\begin{pmatrix} [2,4] & [-2,1] \\ [-1,2] & [2,4] \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} [-2,2] \\ [-2,2] \end{pmatrix}$$



Interval Linear Equations

Example (typical case)

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



Nonlinear Equations

System of nonlinear equations

Let $f : \mathbb{R}^n \mapsto \mathbb{R}^n$. Solve

$$f(x) = 0, \quad x \in \mathbf{x},$$

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

Interval Newton method (Moore, 1966)

- letting $x^0 \in \mathbf{x}$, the Interval Newton operator reads

$$N(\mathbf{x}) := x^0 - \nabla f(\mathbf{x})^{-1} f(x^0)$$

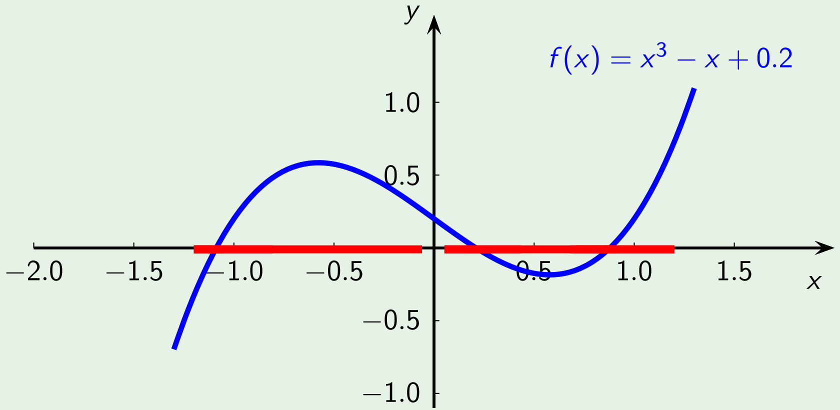
- $N(\mathbf{x})$ is computed from interval linear equations

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

- iterations: $\quad \mathbf{x} := \mathbf{x} \cap N(\mathbf{x})$
- fast (loc. quadratically convergent) and rigorous (omits no root in $\mathbf{x}$)
- if $N(\mathbf{x}) \subseteq \operatorname{int} \mathbf{x}$, then there is a unique root in $\mathbf{x}$

Interval Newton method

Example



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

Eigenvalues of Interval Matrices

Eigenvalues

- For $A \in \mathbb{R}^{n\times n}$, $A = A^T$, denote its eigenvalues $\lambda_1(A) \geq \cdots \geq \lambda_n(A)$.
- Let for $\mathbf{A} \in \mathbb{IR}^{n\times n}$, denote its eigenvalue sets

$$\lambda_i(\mathbf{A}) = \{\lambda_i(A) \colon A \in \mathbf{A},\ A = A^T\}, \quad i = 1, \dots, n.$$

Theorem

- *Checking whether* $0 \in \boldsymbol{\lambda}_i(\mathbf{A})$ *for some* $i = 1, \dots, n$ *is NP-hard.*
- *We have the following enclosures for the eigenvalue sets*

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

- *By Hertz (1992)*

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

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

Global Optimization

Global optimization problem

Compute global (not just local!) optima to

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

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).

Global Optimization by Interval Techniques

Basic idea

- Split the initial box $\mathbf{x}^0$ into sub-boxes.
- If a sub-box does not contain an optimal solution, remove it.
  Otherwise split it into sub-boxes and repeat.

Motivating Example – System Solving

Example

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

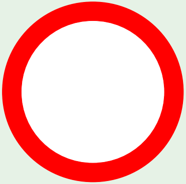


Figure: Exact solution set

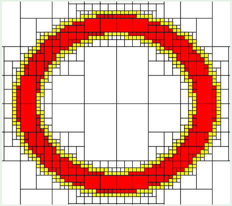
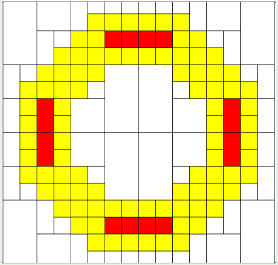


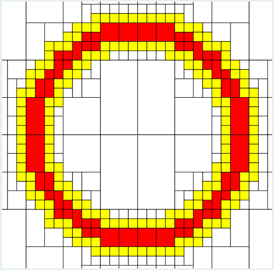
Figure: Subpaving approximation

Motivating Example – System Solving

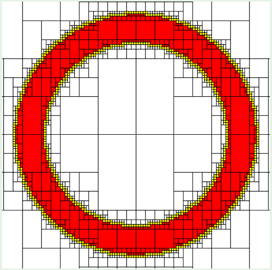
Example (thanks to Elif Garajová)



$\varepsilon = 1.0$
time: 0.952 s



$\varepsilon = 0.5$
time: 2.224 s



$\varepsilon = 0.125$
time: 9.966 s

Interval Approach to Global Optimization

Branch & prune 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

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

- 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 $\mathbf{x}$

Contracting and Pruning

Constraint propagation

Iteratively reduce domains for variables such that no feasible solution is removed by handling the relations and the domains.

Example

Consider the constraint

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

- eliminate x

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

  thus, the domain for x is $[0, 3] \cap [-13, 1] = [0, 1]$
- eliminate y

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

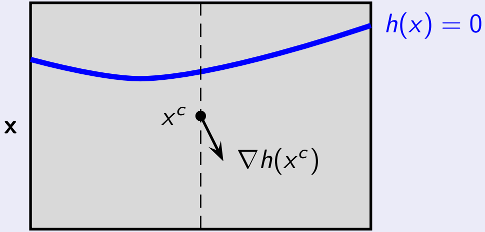
  thus, the domain for y is $[3, 5] \cap [1.5,\ 3.5] = [3,\ 3.5]$

Feasibility Test

Aim

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

- if no equality constraints, take e.g. $x^* := x^c$
- 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 $\nabla h(x^c)$



Feasibility Test

Aim

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

- if no equality constraints, take e.g. $x^* := x^c$
- 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 $\nabla h(x^c)$
- in general, if $k > 1$, transform the matrix $\nabla h(x^c)$ to a row echelon form by using a complete pivoting, and fix components corresponding to the right most columns
- we can include $f(x) \leq c^*$ to the constraints

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
- Lipschitz constant approach
- αBB algorithm
- ...

Lower Bounds: 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

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

αBB algorithm (Androulakis, Maranas & Floudas, 1995)

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

$$g(x) := f(x) + \alpha(x - \underline{x})^T(x - \overline{x}), \quad \text{where } \alpha \geq 0.$$

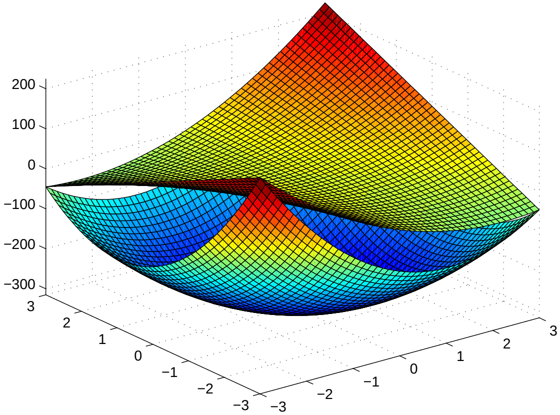
- We want $g(x)$ to be convex to easily determine $\underline{g}(\mathbf{x}) \leq \underline{f}(\mathbf{x})$.
- In order that $g(x)$ is convex, its Hessian

$$\nabla^2 g(x) = \nabla^2 f(x) + 2\alpha I_n$$

must be positive semidefinite on $x \in \mathbf{x}$. Thus we put

$$\alpha := -\tfrac{1}{2}\underline{\lambda}_{\min}(\nabla^2 f(\mathbf{x})).$$

Illustration of a Convex Underestimator



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

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)

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

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

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, volume 13 of *Nonconvex Optimization and Its Applications*.
Kluwer, Dordrecht, 1996.

A. Neumaier.
Complete search in continuous global optimization and constraint satisfaction.
Acta Numerica, 13:271–369, 2004.

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/`
- *Alias* (by Jean-Pierre Merlet, COPRIN team), A C++ library for system solving, with Maple interface, `http://www-sop.inria.fr/coprin/logiciels/ALIAS/`
- *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 `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`)