Complete search for constrained global optimization

Arnold Neumaier

University of Vienna

Vienna, Austria

In a **constraint satisfaction problem (CSP)**,
one asks about **existence**, and one or several examples:

Can 13 *balls of radius r touch a ball of radius R?*

For $r = R$, the answer is No! (Fejes Toth, $\sim$ 1956)

In a **global optimization problem (GOP)**,
one asks about an **extremum**,
and a configuration where it is achieved.

What is the smallest possible R?

It is $R \approx 1.09r$.

In a **constraint projection problem (CPP)**,
one asks about **exhaustion** of the solution set,
and display of a suitable low-dimensional projection of it.

What is the set of possible (r, R)?

$\Sigma = \{(r, R) \mid R \geq 1.09r\}$ *(apart from roundoff).*

In a competitive world, only the best (safest, cheapest, ...) is good enough. This is why optimization (and often global optimization) is very frequent in application.

Global optimization is one of the oldest of sciences, part of the art of successful living.

maximize	**service (or money? or happiness?)**
s.t.	**gifts and abilities**
	hopes and expectations (ours; others)
	bounded stress

Thousands of years of experience ...

. . . resulted in the following algorithmic framework recommended by St. Paul (ca. 50 AD):

"Consider everything. Keep the good. Avoid evil whenever you recognize it." (*1 Thess. 5:21–22*)

In modern terms, this reads:

Do global search by branch and bound!

Here is my personal global optimization problem:

"Be perfect,
as our father in heaven is perfect."
(Jesus, ca. AD 30)

A never ending challenge...

On the mathematical level, the quest for perfection is

rigorous global optimization

Why global optimization?

There are a number of problem classes where it is indispensable to do a complete search.

- **Hard feasibility problems (e.g., robot arm design), where local methods do not return useful information since they generally get stuck in local minimizers of the merit function, not providing feasible points**

- **Computer-assisted proofs (e.g., the recent proof of the Kepler conjecture by HALES), where inequalities must be established with mathematical guarantees**

- **Semi-infinite programming, where the optimal configurations usually involve global minimizers of auxiliary problems**

- **Safety verification problems, where treating nonglobal extrema as worst cases may severely underestimate the true risk**

- **Many problems in chemistry, where often only the global minimizer (of the free energy) corresponds to the situation matching reality**

This talk uses slides made for various occasions, including joint work with

- Hermann Schichl **(Vienna, Austria)**
- Oleg Shcherbina **(Vienna, Austria)**
- Waltraud Huyer **(Vienna, Austria)**
- Tamas Vinko **(Szeged, Hungary)**

within the COCONUT project

(COCONUT = Continuous Constraints – Updating the Technology)

`www.mat.univie.ac.at/coconut`

funded 2000–2004 by the European Community

overhead (dichotomies)

Solver	**Minos**	**LGO**	**BARON**	**ICOS**	**GlobSol**
access language	**GAMS**	**GAMS**	**GAMS**	**AMPL**	**Fortran90**
optimization?	+	+	+	−	+
integer constraints	−	+	+	−	−
search bounds	−	**required**	**recommended**	−	**required**
black box eval.	+	+	−	−	−
complete	−	(−)	+	+	+
rigorous	−	−	−	+	+
local	++	+	+	+	(+)
CP	−	−	+	++	+
other interval	−	−	− +	++	
convex/LP	−	−	++	+	−
dual	+	−	+	−	−
available	+	+	+	+	+
free	−	−	−	+	+

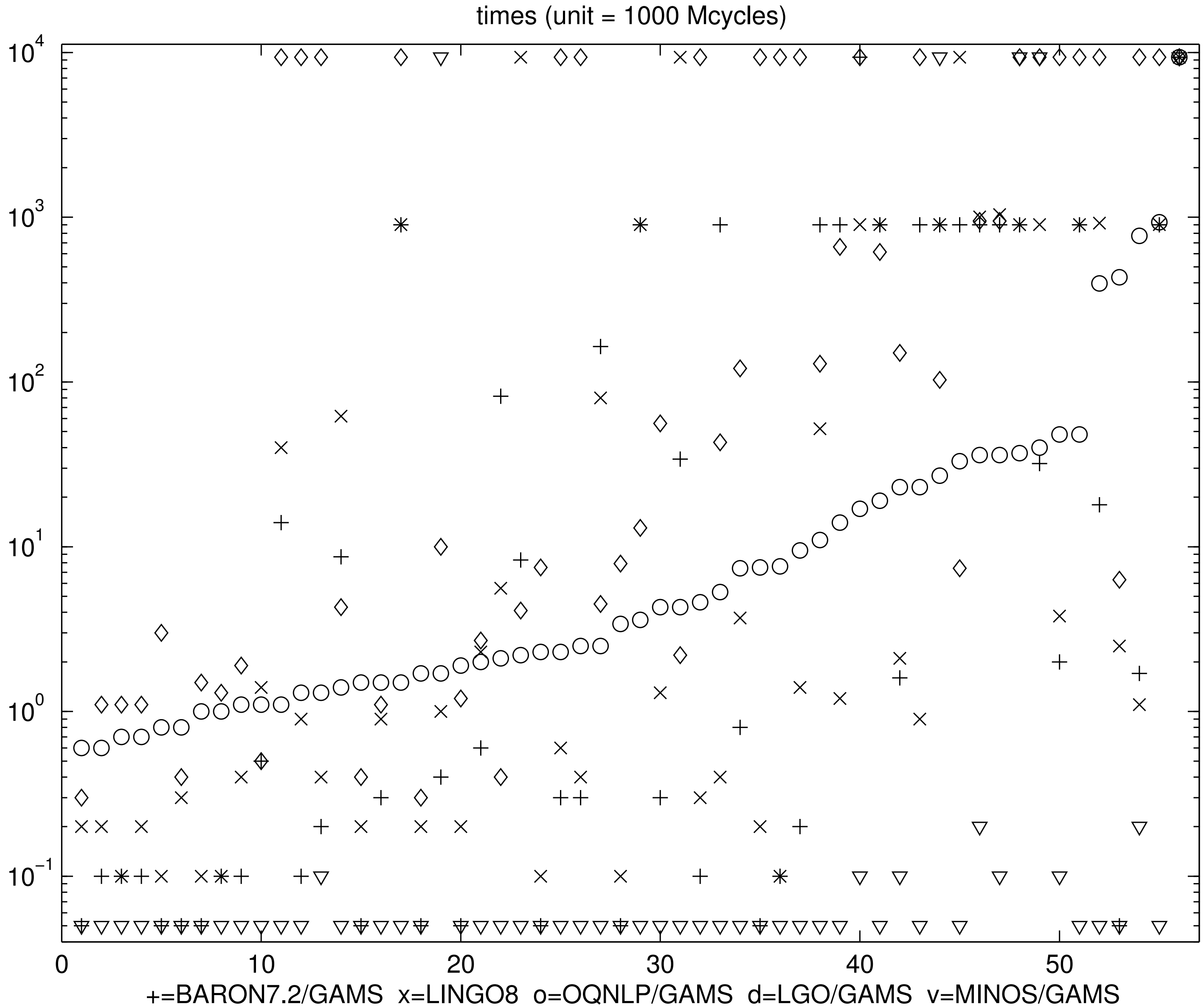
Solver	Premium Solver	LINGO Global	αBB	GloptiPoly	OQNLP
access language	Visual Basic	LINGO	MINOPT	Matlab	GAMS
optimization?	+	+	+	(+)	+
integer constraints	+	+	+	−	+
search bounds	+	-	?	−	+
black box eval.	−	−	−	−	+
complete	+	+	+	+	−
rigorous	(+)	−	−	−	−
local	+	+	+	−	+
CP	+	+	−	−	−
interval	++	+	+	−	−
convex	+	++	++	+	−
dual	−	+	−	++	−
available	+	+	−	+	+
free	−	−	−	+	−

The COCONUT test set

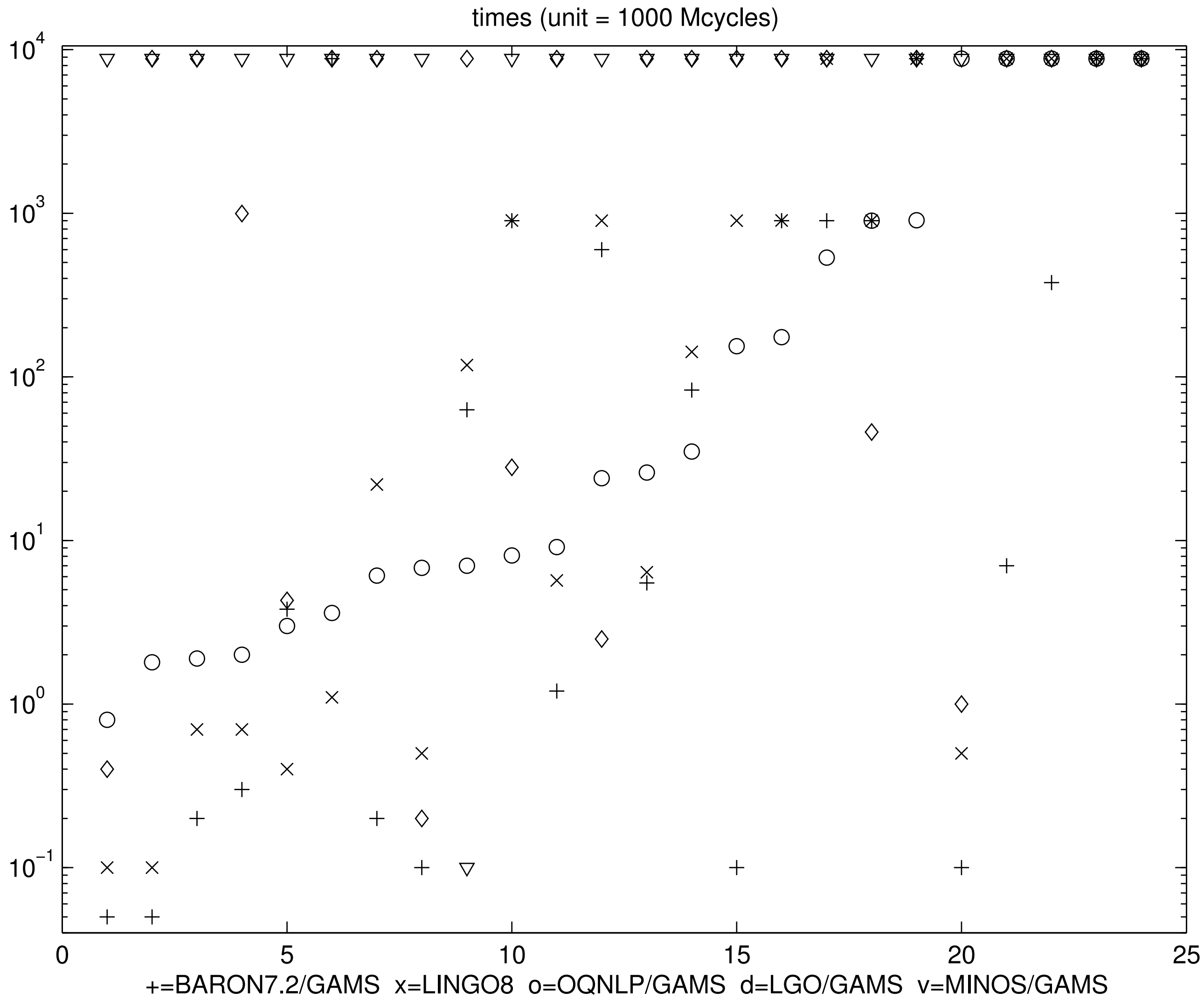
Number of variables	$1-9$	$10-99$	$100-999$	≥ 1000	
	size 1	**size 2**	**size 3**	**size 4**	**total**
Library 1 (GLOBALLIB from GAMS)	**84**	**90**	**44**	**48**	**266**
Library 2 (CUTE from Vanderbei)	**347**	**100**	**93**	**187**	**727**
Library 3 (CSP from EPFL)	**225**	**76**	**22**	**6**	**329**
total	**656**	**266**	**159**	**241**	**1322**

The test set actually used is slightly smaller, and we didn't test size 4.

Detailed test results are available on the COCONUT homepage (`www.mat.univie.ac.at/coconut`).



Size 2 models from GlobalLib ($10 - 99$ variables)
MINOS found global minimizer in 56 out of 80 cases



Size 2 models from GlobalLib ($10 - 99$ variables)
MINOS did not find global minimizer in 24 out of 80 cases

overhead (summary tables)

New, promising solvers:

- **COCOS** (Hermann Schichl, **Vienna**)
- **LaGO** (Ivo Nowak, **Berlin**)

Degrees of rigor

- **incomplete (heuristics; e.g., smoothing techniques)**
- **asymptotically complete (no means to know when a global minimizer has been found; e.g., multiple random start, pure branching)**
- **complete (knows after a finite time that an approximate global minimizer has been found to within prescribed tolerances, assuming exact calculation).**
- **rigorous (complete, with full rounding error control)**

(Often, the label deterministic is used to characterize the last two categories of algorithms; however, this label is slightly confusing since many incomplete and asymptotically complete methods are deterministic, too.)

Incomplete methods are currently still the only choice available for difficult large-scale problems such as protein folding, radiation therapy planning, optimal design and packing. But even this might change in the near future.

As we have seen, complete, but nonrigorous mehods are already today superior to the best general purpose heuristic methods in small and medium dimensions (< 1000).

However, even high quality MILP codes which have already a long commercial tradition may fail due to roundoff!

I noticed the fact that poor handling of roundoff poblems may result in a loss of solutions when I wrote in 1986 my first banch and bound code for covering solution curves of algebraic equations:

Every now and then a pixel was missing in the pictures.

Geometric visualization software had to cope with the same problem, and today they all use carefully designed algorithms with safe rounding in critical computations.

In the optimization community, awareness of this problem is only slowly growing.

CPLEX 8.0 and all but one MILP solver from NEOS failed in 2002 to handle a simple 20 variable MILP problem with small integer coefficients and solution, claiming that no solution exists.

But rigorous safeguards are now available, and will probably be soon part of commercial packages.

We look at the case $s = 6$ of the

20 variable integer linear problem

$$
\begin{array}{ll}
\min & -x_{20} \\
\text{s.t.} & (s+1)x_1 - x_2 \geq s-1, \\
& -sx_{i-1} + (s+1)x_i - x_{i+1} \geq (-1)^i(s+1) \ \textbf{for } i = 2:19, \\
& -sx_{18} - (3s-1)x_{19} + 3x_{20} \geq -(5s-7), \\
& 0 \leq x_i \leq 10 \ \textbf{for } i = 1:13, \\
& 0 \leq x_i \leq 10^6 \ \textbf{for } i = 14:20, \\
& \textbf{all } x_i \textbf{ integers}
\end{array}
$$

CPLEX 8.0 on a LINUX platform, returned (both with and without presolve) after 16 iterations at the root node the message

'Integer infeasible. Current MIP best bound is infinite.'

No further diagnostic information was available.

Surprisingly, upon adding the additional constraint $x_{20} \leq 10$ to this 'infeasible' problem CPLEX produced the solution

$$x = (1, 2, 1, 2, ..., 1, 2)^T.$$

It is easily checked that this is a feasible point (probably the only one).

Thus the negative result CPLEX produced on the original problem was erroneous.

6 other MIP solvers (or MINLP solvers with AMPL input) from NEOS (June 2002):

GLPK, XPRESS-MP, MINLP: *'integer infeasible'*

BONSAIG: *'no solution found'*

XPRESS: *'global search complete – no integer solution found'*

Only FortMP solved the original problem correctly.

The solution is a nondegenerate vertex of the linear programming relaxation (but not of the solution of the LP relaxation).

The coefficient matrix of the linear constraints active at the solution is nonsingular but extremely ill-conditioned; the numerical rank is 19.

$\Rightarrow$ Most solvers suffer from rounding errors introduced through ill-conditioning.

WARNING: A high proportion of real life linear programs (72% according to ORDÓÑEZ & FREUND**, and still 19% after preprocessing) are ill-conditioned!**

Primal linear program:

$$\begin{aligned} \min \quad & c^T x \\ \text{s.t.} \quad & \underline{b} \le Ax \le \overline{b}, \end{aligned} \tag{1}$$

Corresponding dual linear program:

$$\begin{aligned} \max \quad & \underline{b}^T y - \overline{b}^T z \\ \text{s.t.} \quad & A^T(y - z) = c, \;\; y \ge 0, \;\; z \ge 0. \end{aligned} \tag{2}$$

Introduce boxes:

$$\mathbf{b} := [\underline{b}, \overline{b}] = \{\tilde{b} \in \mathbb{R}^n \mid \underline{b} \le \tilde{b} \le \overline{b}\},$$

Assume

$$Ax \in \mathbf{b} \quad \Rightarrow \quad x \in \mathbf{x} = [\underline{x}, \overline{x}].$$

From an approximate solution of the dual program we calculate an approximate multiplier $\lambda \approx z - y$**, and a rigorous interval enclosure for**

$$r := A^T\lambda - c \in \mathbf{r} = [\underline{r}, \overline{r}].$$

Since $c^T x = (A^T\lambda - r)^T x = \lambda^T A x - r^T x \in \lambda^T \mathbf{b} - \mathbf{r}^T \mathbf{x}$,

$$\mu := \inf(\lambda^T \mathbf{b} - \mathbf{r}^T \mathbf{x})$$

is the desired rigorous lower bound for $c^T x$**.**

In well-conditioned cases, the bound is quite accurate, while in ill-conditioned cases, it is so poor that it warns the user (or the algorithm) that something went wrong and needs special attention.

Safeguarding MILP solutions is more involved but can also be done.

With these slides we crossed the line between pure advertisement and mathematical analysis.

Of course, the great success of the current generation of complete global solvers is due mainly to improvements in our ability to analyze global optimization problems mathematically.

For history, a thorough introduction, a comprehensive overview over the techniques in use, and extensive references see my survey

A. Neumaier

Complete search in continuous global optimization and constraint satisfaction,

pp. 1-99 in: Acta Numerica 2004,

Cambridge Univ. Press 2004.

Complexity

Already in the case where only bound constraints are present, global optimization problems and constraint satisfaction problems are

- **undecidable on unbounded domains (Wenxing Zhu 2004), and**
- **NP-hard on bounded domains.**

This implies natural limits on the solvability of such problems in practice.

In particular, methods which work with direct attack (analytic transformations without problem splitting) lead to transformed problems of exponentially increasing size,
while branch-and-bound methods split the problems into a worst case exponential number of subproblems.

It is very remarkable that in spite of this, many large-scale problems can be solved efficiently.

This is achieved by carefully balancing the application of the available tools and using the internal structure which realistic problems always have.

Complete search techniques I

a) Direct attack is feasible for polynomial systems of moderate degree (up to about 20)

- semidefinite relaxations
- Gröbner basis methods
- resultant-based techniques

Complete search techniques II

Branch-and-bound methods are the choice for larger problems. Basic approaches use

- **constraint propagation**
- **outer approximation (linear, convex, conic, semidefinite)**
- **DC (difference of convex function) techniques**
- **interval Newton and related methods**

Complete search techniques III

Further efficiency is gained by

- use of optimality conditions
- multiplier techniques
  (duality, Lagrangian relaxation)
- cut generation
- adding redundant constraints
- graph decomposition techniques

Complete search techniques IV

Efficiency and reliability also require the use of

- **local optimization for upper bounds**
- **clever box selection heuristics**
- **adaptive splitting heuristics**
- **reliable stopping criteria**
- **combination heuristics**
- **safeguarding techniques**

While there would be a lot to say about all the techniques, I'll concentrate in the following on interval techniques, since, in the optimization community, this is the least well-known set of tools.

In particular, conic and semidefinite relaxations and polyhedral outer approximation are already covered in other talks at this conference.

The organizers arranged the parallel semiplenary talk to be by Paul Tseng on second-order cone relaxations.

Perhaps to make sure that no one escapes getting informed about global optimization techniques?

Unfortunately, I have only one copy of myself; otherwise a second copy would sit in his lecture!

Interval techniques became initially known mainly as a tool to control rounding errors. However, it is much less known – and much more important – that their real strength is the ability to control nonlinearities in a fully satisfying algorithmic way.

A full account of the theoretical background for interval techniques in finite dimensions is available in my book

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

The book is still up to date (with a few minor exceptions). While it is officially out of print, if you order it at Cambridge University Press, they'll print an extra copy especially for you. (Apparently, this is still profitable for them.)

Challenges for the future I

- **Ensuring reliability (safe bounds, finite termination analysis, certificates)**
- **Integrating MIP and SDP techniques into a branch-and-bound framework**
- **unbounded variables**
- **unconstrained/bound constrained problems (the more constraints the easier the problem! $\Rightarrow$ bounded residual estimation preferable to least squares)**

Challenges for the future II

- **Problems with severe dependence (volume preserving recurrences imply heavy wrapping)**
- **Problems with symmetries (optimal design of experiments, chemical cluster optimization)**
- **sensitivity analysis**
- **parametric global optimzation**
- **constraint projection problems**

Challenges for the future III

- **Differential constraints**
  (optimal control of chemical plants;
  space mission design)

- **Integral constraints**
  (expectations; value at risk,
  engineering safety factors)

- **Other desirables**
  (black box functions; expensive functions;
  nonsmoothness; noise; small discontinuities;
  uncertain domain of definition; SNOBFIT

final slide (references)

A. Neumaier
Complete search in continuous global optimization and constraint satisfaction,
pp. 1-99 in: Acta Numerica 2004,
Cambridge Univ. Press 2004.

Global (and Local) Optimization site

`www.mat.univie.ac.at/~neum/glopt.html`

COCONUT homepage

`www.mat.univie.ac.at/coconut`

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