MINLP : a practical, user inspired perspective.

Paris (FR)
08/04/2016

Andrea Cassioli, PhD
andrea.cassioli@mosek.com

www.mosek.com

Overview

Intro

Motivation

Some of you will be or already are

- an academic,
- an OR specialist/consultant,
- working for a solver provider,
- developping your own software,
- about to change field but still use optimization tools or
- hating OR for the rest of your life....

Optimization in the spotlight

Optimization in under the spotlight more than ever!

Big Data, prescriptive analytics have put optimization in many research curricula, the *data scientists.*

It is important to understand how people outside the optimization community see optimization.

"Definitions"

user

A person with a decent background in math but not specifically in OR. He/She often speaks an application-oriented jargon that may conlfict with that of optimization people.

solver

The whole set of tools used to solve a MINLP problem.

MINLP at glance

What

$$\begin{aligned} \min_{x,y} \quad & f(x,y) \\ s.t. \quad & \\ & g_j(x,y) \leq 0 \qquad j = 1, \ldots, m \\ & x \subseteq \mathbb{R}^n \\ & y \subseteq \mathbb{Z}^m \end{aligned}$$

Assume $m > 1$ or g_j non-convex for some j.

When

MINLP is an hot topic since the very early days in the optimization community.

Even from the commercial point of view, we can find a MINLP solver on the market already in the '70 (SCICONIC)

How

Plenty of approaches, among them

- Branch-and-Bound
- Outer Approximation
- Meta-Heuristic
- you name it...

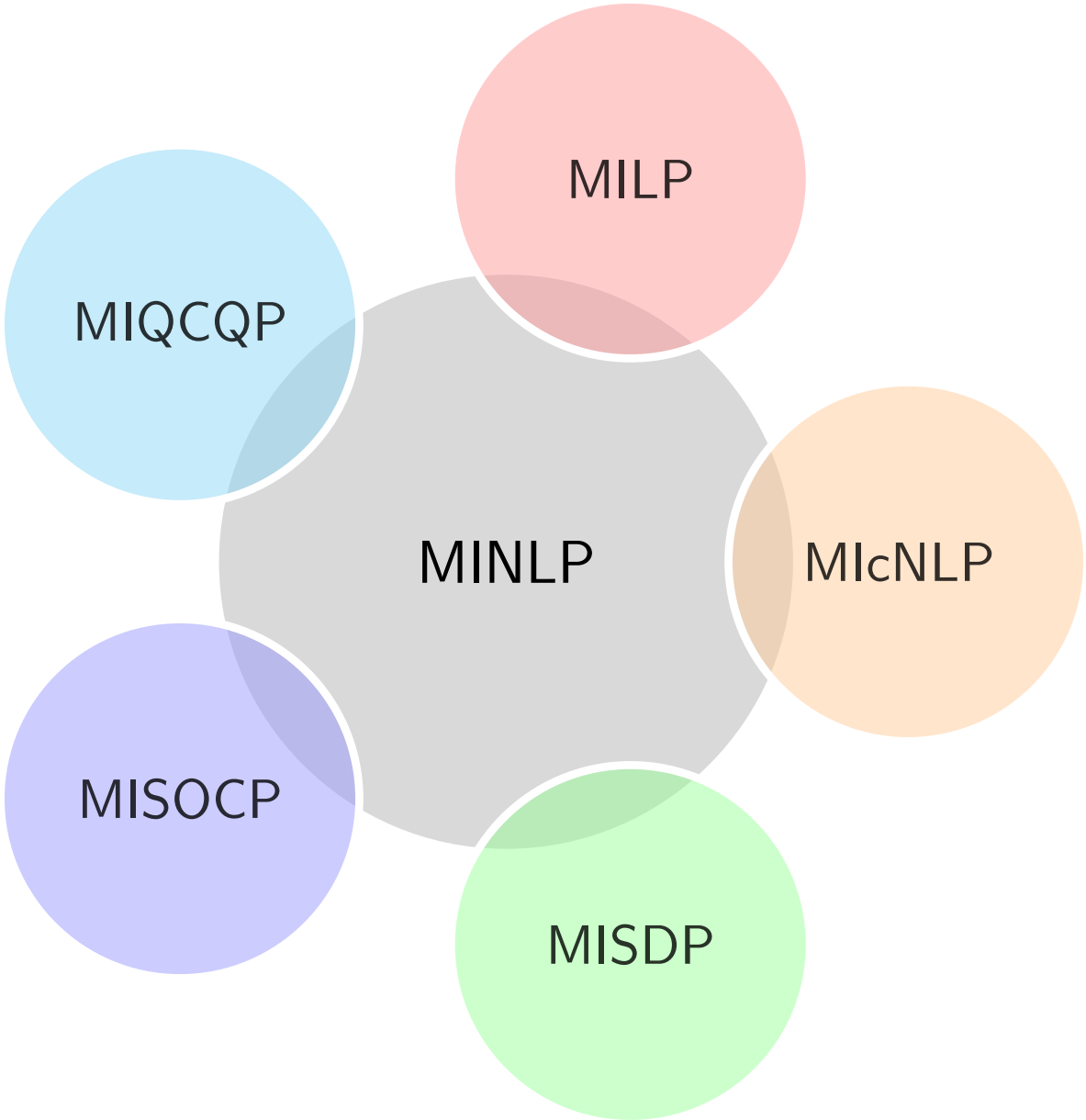
Taxonomy

To make a taxonomy is difficult.

Convexity

The convexity of the continuous relaxation is probably the most important discriminant.

A colourful set of bubbles



Choose the right bubble

Good modeling to choose the right bubble:

- the right problem type
- problem practical solvability
- solver choice
- trade-off between performance and accuracy

Users expectations

First axiom

Axiom

The user is right.

Corollary

If anything goes wrong, the solver must be blamed.

What to look for

Key factors

- robustness
- performance
- reproducibility/determinism

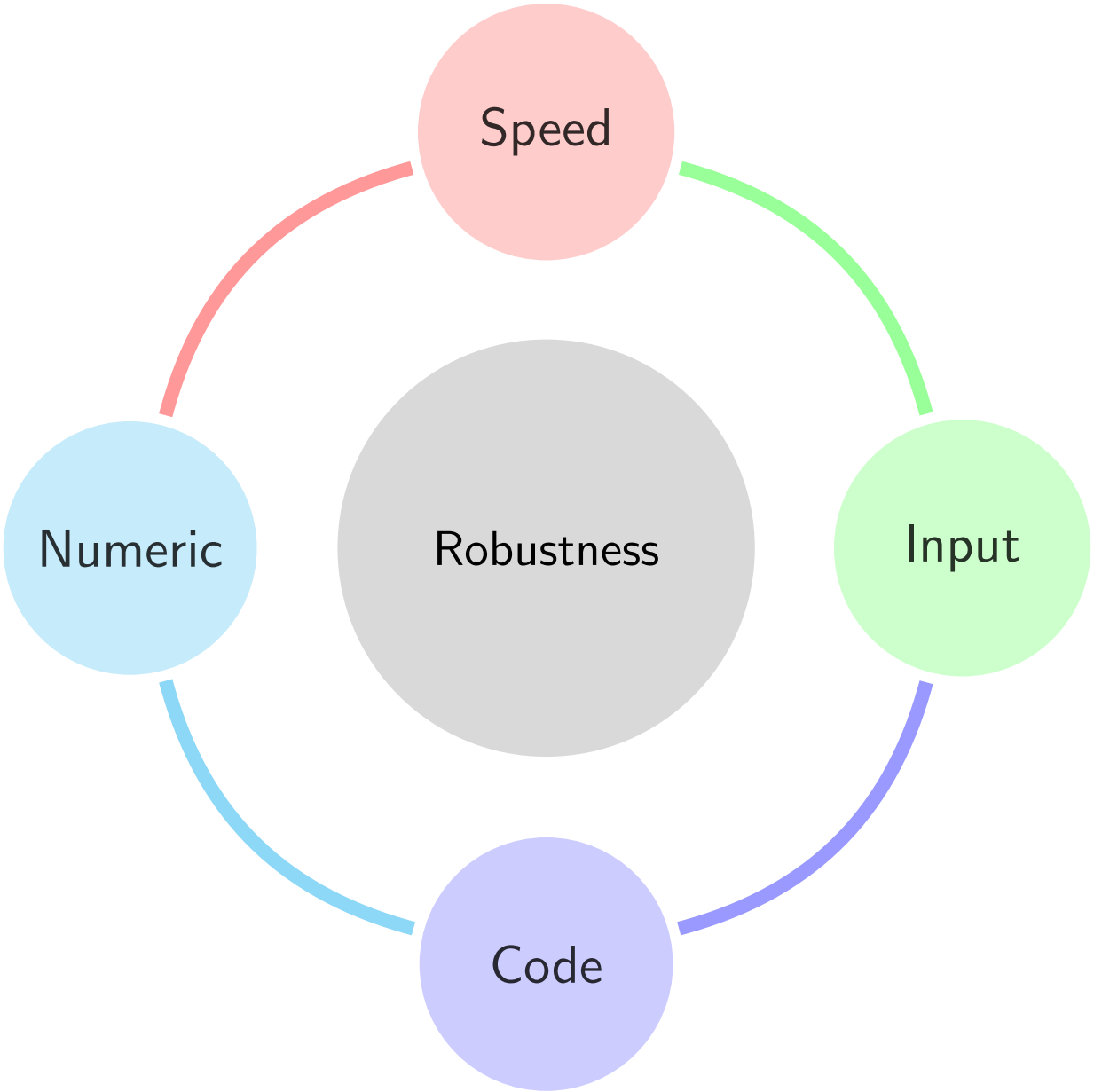
Robustness is king.

From a user in the finance industry:

> *"Guys, in our stress tests we record* **one instance over 2 million** *in which the solver gives an optimal solution but the return code is meaningless. Shall we be worried? Could you investigate asap??"*

During monthly sanity checks....

Robustness



Performance

Performance is of course important but also:

- parallel scalability
- external factors

Proper benchmark is not trivial at all!

Reproducibility/Determinism

This is a surprising popular request.

- testing
- sanity checks
- regulatory purposes
- fell like reliability!

Modeling

Modeling

Modeling is by far the most important thing from a user perspective:

- determine problem practical solvability
- lead the solver choice
- may introduce a trade-off between performance and accuracy

Example 1

A simple problem

We want to optimize energy production from n power plant under the following requirements

- each plant produce x_i energy
- the total energy must be T
- each plant runs with **inefficiency**

$$f_i(x_i) = \frac{(a_i - x_i)^2}{a_i}, a_i > 0$$

- when $x_i = 0$ then $f_i(x_i)$ must be 0.

Source:
`http://yetanothermathprogrammingconsultant.blogspot.dk/2016/02/two-nonlinear-formulations.html`

Example 1

Intuitive formulation

$$\min \quad \sum_i y_i f_i(x_i)$$

$$s.t.$$

$$\sum_i x_i = T$$

$$x_i \leq M_i y_i \qquad i = 1, \ldots, n$$

$$x_i \in [0, M_i] \qquad i = 1, \ldots, n$$

$$y_i \in \{0, 1\} \qquad i = 1, \ldots, n$$

Example 1

A better formulation

$$\min \quad \sum_i \frac{d_i^2}{a_i}$$

$$s.t.$$

$$\begin{aligned}
x_i &\leq y_i M_i && i = 1, \ldots, n, \\
-M_i(1 - y_i) \leq s_i &\leq M_i(1 - y_i) && i = 1, \ldots, n, \\
d_i &= a_i - x_i + s_i && i = 1, \ldots, n, \\
x_i &\geq 0 && i = 1, \ldots, n, \\
y_i &\in \{0, 1\} && i = 1, \ldots, n.
\end{aligned}$$

Example 1

Lesson

Simple reformulation leads from a non convex MINLP to a convex MIQP.

Example 2

Intuitive formulation

$$\min \quad \sum_i x_i^{3/2}$$

$$s.t.$$

$$\sum_i x_i = T$$

$$x_i \geq 0 \qquad i = 1, \ldots, n$$

$$y_i \in \{0, 1\} \qquad i = 1, \ldots, m.$$

Some constraints involving y_i intentionally left out.

Example

A conic formulation

$$
\begin{aligned}
\min \quad & \sum_i t_i \\
s.t. \quad & \\
& (s_i, t_i, x_i) \in \mathcal{Q}_r^3 && i = 1, \ldots, n \\
& (x_i, 1/8, s_i) \in \mathcal{Q}_r^3 && i = 1, \ldots, n \\
& \sum_i x_i = T, \\
& x_i \geq 0 && i = 1, \ldots, n, \\
& y_i \in \{0, 1\} && i = 1, \ldots, m.
\end{aligned}
$$

Example 2

Lesson

Simple reformulation leads from a general convex MINLP to a convex MISOCP.

Bad news:
Many users have no will to work on the problem formulation, but they like to tweak parameters instead.

Model accuracy I

Do I really need such a detailed/complex model?

Intuitive formulations may be uselessly rich and detailed:

Approximation? Relaxation? Convexification?

Model accuracy II

Do I really need the global optimum?

In many application a decent improvement is enough!

That leads often to meta/math/heuristics.....

Model accuracy III

Do I really need such a high quality solution?

Sometimes the user might be fine with good but suboptimal solutions. It is not always aware of that.

Modeling Issues

Bad news

Users already struggle with LPs and QCQPs!

Things can only get worse with MINLP....

Scaling

Bad scaling is the most common mistake and support request!
Typically:

- coefficients of very different size

$$3x + 0.001y + 10^8 z = 0$$

- very large bounds

$$x \in [0, 10^{10}], y \in [0, 0.0001]$$

- unecessary bounds (*just in case...*)

$$x \in [0, 10^{10}]$$

Linear dependencies

Be aware of *numerical* linear dependency!

They are a subtle issue that arises

- typically in discretization, OA, piecewise underestimators,
- sometimes from automatic routines,
- may fool even establish LP solvers!

Help the user

What to do?

- Smarter solvers/modeling frameworks
- More dissemination
- More skilled practitioners

Software

Overview

Quite a number of MINLP solvers are available, but no need for a list!

Many packages start to be mature and established.

The same applies to modeling tools.

Benchmark

A critical issue:

Along collections of libraries, we would like to see more real life instances, in all sectors!

Recently very nice tendency: put your code+instances on GitHub (or the like) to help people reproduce your tests.

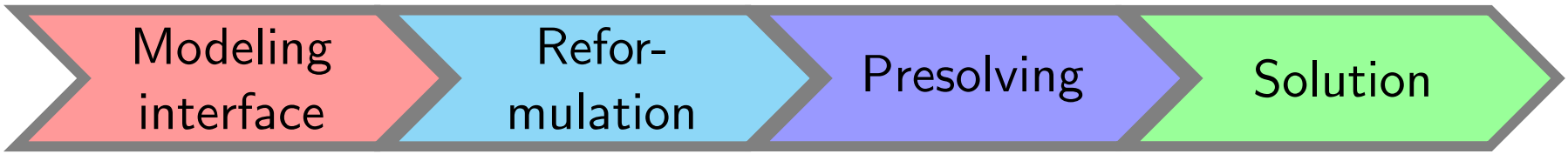
Modeling framework or not?

Modeling frameworks

- can make magic,
- allow to change solver easily,
- allow to move the burden from the solver to a specific piece of code and
- focus more on the user interface and usability.

However they may reduce the user control on what is actually fed in the solver!

Overall scheme



SDP slowly rising

Present

SDP has been around for quite some time, but never went much out of the academic world.

- intrinsic higher complexity
- practical implementation challenge
- difficult to achieve the right trade-off speed vs. accuracy

Moving forward

Things are changing: interest is growing outside the optimization community

- power flow
- robotics
- optimal control
- statistic

Growing interest among commercial users.

Easier to use

Better interfaces/modeling tools like

- CVX
- Yalmip
- Julia Convex/JuMP

have helped a lot users!

SDP as basic tool

Driving forces are now:

- Polynomial Optimization
- relaxations of MIQCQP
- MISDP

SDP solvers are put under great pressure and testing!

Future?

Possible turning point close ahead, provided that

- some practical issues will be solved
- some more widespread application will start pushing SDP into commercial.

Note: SDP needs very skilled people!

Conclusion

A glorious past

MINLP is quite an established technology

- tons of papers and books,
- several approaches and
- millions of lines of code already out there!

... a vibrant present ...

Just look around to all your fellow course mates!

... and a bright future!

MINLP is going to spread even more!

- core solvers getting better every day,
- theory still in development,
- computational resources more and more accessible to all!

Thank you!

Andrea Cassioli, PhD

andrea.cassioli@mosek.com

www.mosek.com