Solving MPEC Models with the KNITRO Nonlinear Solver

Richard Waltz, President
Ziena Optimization LLC

with
Jorge Nocedal (Ziena, Northwestern Univ.)
Arnaud Renaud (Artelys)
Sylvain Mouret (Artelys)

www.ziena.com

Outline

1. KNITRO Short Introduction
2. What Are MPECs and Why Are They Useful?
3. Application in OPF Models (faster alternative to MIP)

The KNITRO solver (in a nutshell)

KNITRO: short introduction

KNITRO is a *commercially supported* optimization toolbox with a focus on solving large-scale nonlinear optimization problems.

Developed by Ziena Optimization LLC since 2001, distributed/supported by Artelys

Key features

- Active-set and interior-point/barrier algorithms for continuous optimization
- MINLP algorithms and complementarity constraints for discrete optimization
- Parallel multi-start heuristic for global optimization of non-convex problems
- Other notable features:
  - presolver, crossover, run all algorithms in parallel, derivative approximations, fast infeasibility detection
  - tuner (coming soon in 9.0)

Supported interfaces

- C/C++, Java, Fortran, Python
- AMPL, AIMMS, GAMS
- MATLAB

Supported platforms: Windows 32-bit, 64-bit, Linux 64-bit, Mac OS X 64-bit

Widely used in academia and industry

Mathematical Programs with Equilibrium Constraints (MPEC)

Mathematical Program w/ Equilibrium Constraint (MPEC)

Math **P**rogramming with **E**quilibrium **C**onstraints
aka: Complementarity Constraints (MPCC)

$$\min f(x)$$
$$\text{s.t. } g(x) \geq 0$$
$$h(x) = 0$$
$$\boxed{0 \leq x_1 \perp x_2 \geq 0}$$

Applications:
- Strategic bidding
- Economic models
- Contact problems
- Traffic equilibrium
- Disjunctive conditions

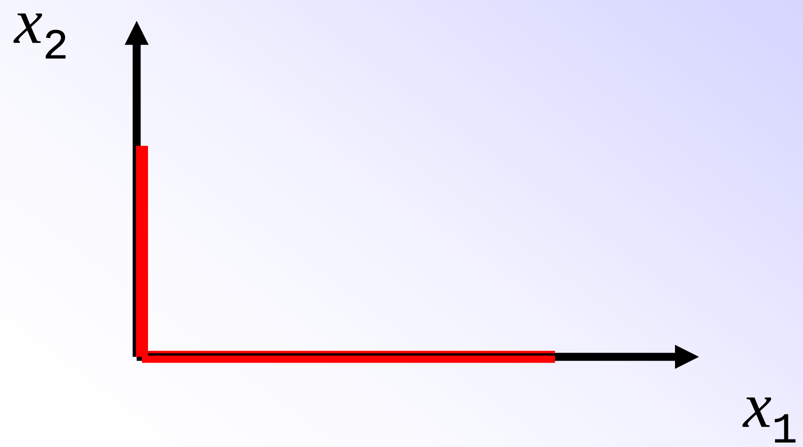
The complementarity condition means

$x_1 \geq 0$, $x_2 \geq 0$ and

either $x_1 = 0$ or $x_2 = 0$ (equivalently, $x_1 x_2 = 0$)

Why is MPEC Hard?

$$\min f(x)$$
$$\text{s.t. } g(x) \geq 0$$
$$h(x) = 0$$
$$\boxed{0 \leq x_1 \perp x_2 \geq 0}$$



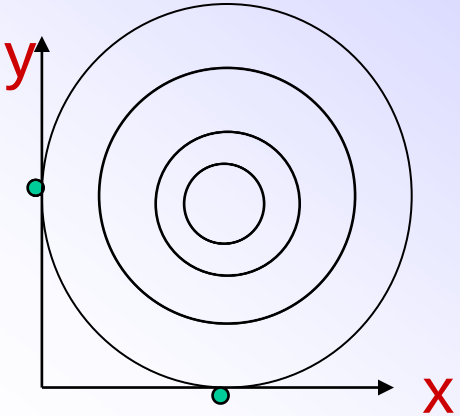
- Optimality conditions for NLP ***assume*** constraints satisfy some nice conditions (*constraint qualification*); *Constraint qualifications do not hold for MPECs*
- No interior path for interior-point methods
- Inconsistent constraint linearizations for SQP methods
- Unbounded multipliers

Why is MPEC Hard ?   (example)

$$\min \quad (x-1)^2 + (y-1)^2$$

$$\mathrm{c}_1 = x \geq 0, \qquad \mathrm{c}_2 = y \geq 0$$

$$\mathrm{c}_3 = xy = 0$$



$$\nabla c_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \qquad \nabla c_3 = \begin{pmatrix} y \\ x \end{pmatrix}$$

$$\text{When } x = 0, \quad \nabla c_3 = \alpha \nabla c_1$$

Linear dependent constraint gradients at every feasible point!

Why is MPEC Hard (in practice)?

What does this mean in practice?

- Numerical difficulties for the solver (e.g. singular linear systems)
- Challenges identifying solutions (finding multipliers to satisfy KKT conditions)

How to deal with it in practice?

- Need to identify and apply special treatment to these constraints
- *Relax*, *regularize* or *penalize* them in some way

Recent research breakthroughs allow reformulation for interior-point solvers via **penalization**

$$
\begin{aligned}
&\min f(x) \\
&\text{s.t. } g(x) \geq 0 \\
&h(x) = 0 \\
&\boxed{0 \leq x_1 \perp x_2 \geq 0}
\end{aligned}
\quad \longrightarrow \quad
\begin{aligned}
&\min f(x) \boxed{+\pi\, x_1^T x_2} \\
&\text{s.t.} \quad g(x) \geq 0 \\
&h(x) = 0 \\
&x_1 \geq 0, x_2 \geq 0
\end{aligned}
$$

KNITRO MPEC Benchmarks

Comparing CPU times on a sample of MPECs

Mittelmann Benchmarks (http://plato.asu.edu/bench.html)

problem	filter	LOQO	KNITRO
bem-milanc30-s	648	f	f
frictionalblock_5	f	f	1
frictionalblock_6	f	f	f
incid-set1-32	147	56	3
incid-set1c-32	100	50	1
incid-set2-32	165	42	2
incid-set2c-32	99	f	1
pack-comp1-32	150	f	1
pack-comp1c-32	15	f	1
pack-comp1p-32	234	f	1
pack-comp2-32	301	37	1
pack-comp2p-32	186	f	1
pack-rig1-32	71	f	f
pack-rig1c-32	14	f	1

problem	filter	LOQO	KNITRO
pack-rig1p-32	64	29	1
pack-rig2-32	f	f	f
pack-rig2c-32	f	f	479
pack-rig2p-32	39	f	1
pack-rig3-32	65	f	1
pack-rig3c-32	15	f	1
siouxfls1	90	f	f
siouxfls	213	f	f

Binary variables vs MPECs

Binary variable: *x=0 or x=1* (disjunction)

NLP/MPEC form:

Let *y=x-1* and represent as $0 \leq x \perp y \geq 0$

Comparison:

- MIP vs. NLP/MPEC
- MIP gives global solution (but may be too hard to solve)
- NLP/MPEC form creates lots of non-convexity
  - Can create many local solutions (and locally infeasible points!)
  - …but can solve quickly (use as heuristic for MIP)

THE OPF PROBLEM

THE OPF MODEL

AC-OPF model

- Polar PQV formulation
- Limits on voltage magnitudes
- Maximum intensity levels on lines (nonlinear inequality constraints)
- Limits on production levels
- Kirchhoff law at each node (nonlinear equality constraints)

Generators operational constraints

- The active (P) and reactive (Q) power of a production group are
  - bounded if the group is ON
  - zeros if the group is OFF

Objective

- **Minimize losses** → similar to minimizing total production

Artelys | OPTIMIZATION SOLUTIONS

MINLP APPROACH

MINLP FORMULATION

- Group is OFF

$$P = 0$$

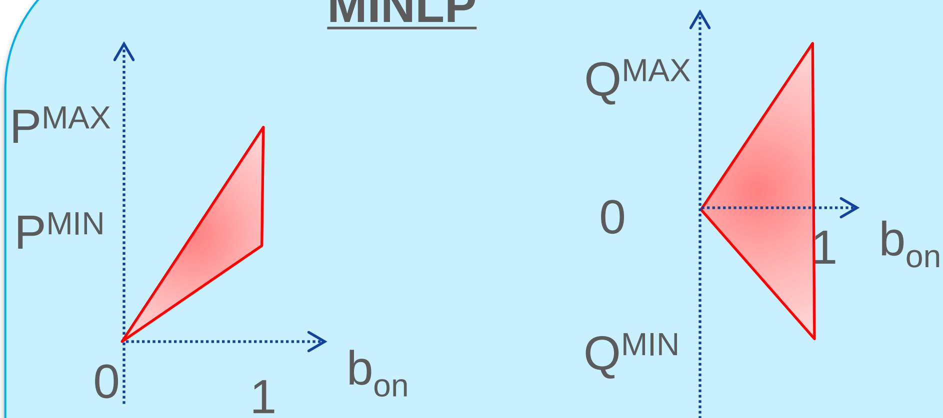
$$Q = 0$$

- Group is ON

$$P \in \left[P^{MIN}, P^{MAX}\right]$$

$$Q \in \left[Q^{MIN}, Q^{MAX}\right]$$

MINLP



$$b_{on} \in \{0,1\}$$

$$P^{MIN} \cdot b_{on} \leq P \leq P^{MAX} \cdot b_{on}$$

$$Q^{MIN} \cdot b_{on} \leq Q \leq Q^{MAX} \cdot b_{on}$$

GENERATION OF PROBLEM INSTANCES

- Use of an instance from the literature (IEEE)
  - Network composed of 662 nodes
  - Input data : topology, productions and consumptions, limits on power and voltage
  - Some phase-shifting transformers have variable transformer ratio
- Generation of 100 problem instances
  - Modifying production groups by adding a minimum capacity limit
    - Minimum capacity set to half of the maximum capacity
  - Perturbation of consumption (±50%)
    - Conservation of the total demand in active/reactive power

MINLP FORMULATION

- Algorithm : Branch-and-Bound of KNITRO 8.1.1
  - Intel Core 2 Duo @ 2.80 Ghz 4Go RAM
- Average 13.2 seconds by instance
- 100 % success rate
- Can we do faster ?

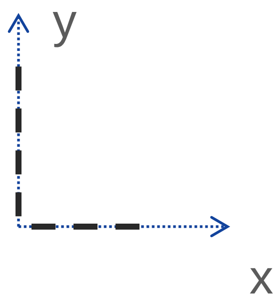
MPEC FORMULATIONS

MODELING STARTUP/SHUTDOWN OF PRODUCTION GROUP

- Use of complementarity constraints (MPEC)
  - $0 \leq x \perp y \geq 0 \Leftrightarrow (x = 0 \; or \; y = 0) \text{ and } (x, y \geq 0)$
  - Non-convex constraints on continuous variables

- Various mathematical formulations are possible:
  - $0 \leq x, y \quad xy \leq \varepsilon$
  - $0 \leq x, y$ and addition of a penalty term $\pi \cdot x \cdot y$

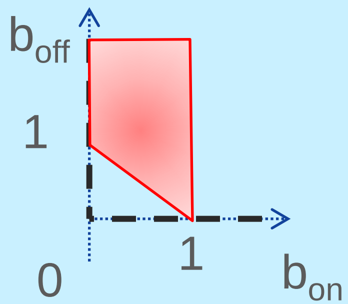
- Geometric representation:



MPEC-MINLP REFORMULATIONS

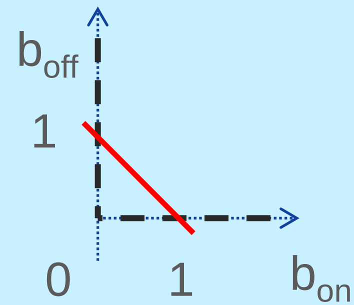
- Reformulation of $b_{on} \in \{0,1\}$ as an intersection of
  - linear polyhedron: $$P^{MIN} \cdot b_{on} \le P \le P^{MAX} \cdot b_{on}$$ $$Q^{MIN} \cdot b_{on} \le Q \le Q^{MAX} \cdot b_{on}$$
  - complementarity constraints

MPEC-MINLP1



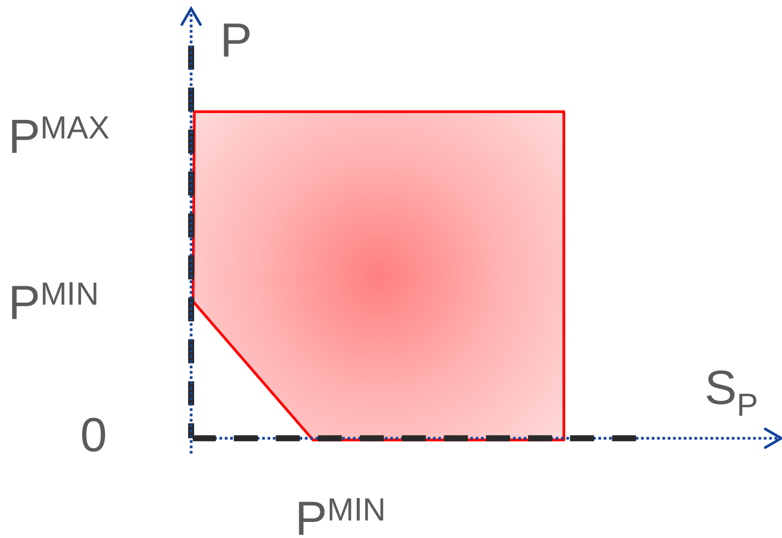
$$0 \le b_{on} \perp b_{off} \ge 0$$
$$b_{on} \le 1$$
$$b_{on} + b_{off} \ge 1$$

MPEC-MINLP2



$$0 \le b_{on} \perp 1 - b_{on} \ge 0$$

MPEC-NLP REFORMULATIONS



- Reformulation of constraints on active/reactive powers P and Q instead of b_{on}
  - Leads to MPEC constraints linking P, Q and a new continuous slack variable S_P

MPEC-NLP1

$$0 \le P + Q^2 \perp S_P \ge 0$$

$$P + S_P \ge P^{MIN}$$

$$0 \le P \le P^{MAX}$$

$$Q^{MIN} \le Q \le Q^{MAX}$$

MPEC-NLP2

$$0 \le P + Q^+ + Q^- \perp S_P \ge 0$$

$$Q = Q^+ - Q^-, Q^+ \ge 0, Q^- \ge 0$$

$$P + S_P \ge P^{MIN}$$

$$0 \le P \le P^{MAX}$$

$$Q^{MIN} \le Q \le Q^{MAX}$$

COMPUTATIONAL EXPERIMENTS

NUMERICAL RESULTS

Average results on the 100 instances

With KNITRO 8.1.1 on Intel Core 2 Duo @ 2.80 Ghz 4Go RAM

Formulation	CPU time (s)	Losses (SU)	% ON	% SUCCESS
MINLP	13.2	3.12	98.6	100
MPEC-MINLP1	1.89	5.08	80.6	80
MPEC-MINLP2	0.85	3.80	94.4	98
MPEC-NLP1	1.33	3.16	99.7	98
MPEC-NLP2	2.03	4.01	91.8	100

- The MPEC heuristics are faster than the MINLP approaches
- The various approaches converge to different solution points
  - with varying numbers of active generators
  - with varying losses

NUMERICAL RESULTS

Average results on the 100 instances

With KNITRO 8.1.1 on Intel Core 2 Duo @ 2.80 Ghz 4Go RAM

Formulation	CPU time (s)	Losses (SU)	% ON	% SUCCESS
MINLP	13.2	3.12	98.6	100
MPEC-MINLP2	0.85	3.80	94.4	98
MPEC-NLP1	1.33	3.16	99.7	98

- MPEC-NLP1 converge towards solutions close to MINLP
  - in terms of losses (near-optimal solutions)
  - in terms of active production groups
- MPEC-MINLP2 is faster than MPEC-NLP1
  - but losses are higher (sub-optimal solutions)

CONCLUSION

- MPEC models may be a good heuristics alternative to solve non-convex MINLP problems
  - Branch-and-bound tends to be slow
  - Good when combinatorial choices are not too hard
- Some success in applying this technique to AC-OPD models
  - Use complementarity constraints to model startup/shutdown decisions
- Future work (iTesla project)
  - Apply to large instances
    - >7000 nodes, >8000 lines
  - Analyze robustness on a large number of instances

Artelys | OPTIMIZATION SOLUTIONS

THANK YOU FOR YOUR ATTENTION

TRIAL VERSION CAN BE DOWNLOADED AT

HTTP://WWW.ZIENA.COM/TRIAL.HTML

MORE INFO AT

HTTP://WWW.ARTELYS.COM/KNITRO

TRY KNITRO!

- Trial version can be downloaded at
- http://www.ziena.com/trial.html

- More info at
- http://www.artelys.com
- info-knitro@artelys.com