Solution of Structured Complementarity Problems

Michael C. Ferris

University of Wisconsin-Madison

ICCOPT II
August 13, 2007

Modeling languages: state-of-the-art

- Optimization models improve understanding of underlying systems and facilitate operational/strategic improvements
- Key link to applications, prototyping of optimization capability
- Widely used in:
    - engineering - operation/design
    - economics - policy/energy modeling
    - military - operations/planning
    - finance, medical treatment, supply chain management, etc.
- Interface to solutions: facilitates automatic differentiation, separation of data, model and solver
- Modeling languages no longer novel: typically represent another tool for use within a solution process.

Modeling Language Limitations

- Data (collection) remains bottleneck in many applications
  - Tools interface to databases, spreadsheets, Matlab
- Problem format is old/traditional

$$\min_x f(x) \text{ s.t. } g(x) \leq 0, h(x) = 0$$

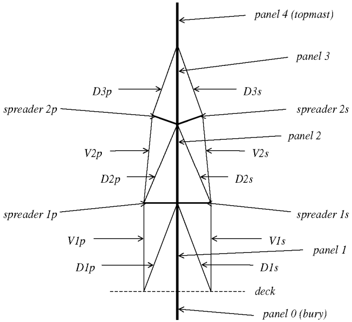
  - Support for integer, sos, semicontinuous variables
  - Limited support for logical constructs
  - Support for complementarity constraints

Optimal Yacht Rig Design

- Current mast design trends use a large primary spar that is supported laterally by a set of tension and compression members, generally termed the rig
- Reduction in either the weight of the rig or the height of the VCG will improve performance

Complementarity Feature

- Stays are tension-only members (in practice) – Hookes Law
- When tensile load becomes zero, the stay goes slack (low material stiffness)



$$0 \geq s \perp s - k * dl \leq 0$$

s: axial load
k: member spring constant
dl: member length extension

MPEC: complementarity constraints

$$\begin{array}{ll} \min_{x,s} & f(x,s) \\ \text{s.t.} & g(x,s) \leq 0, \\ & 0 \geq s \perp h(x,s) \leq 0 \end{array}$$

- Complementarity "⊥" constraints available in AMPL and GAMS
- NLPEC: use the convert tool to automatically reformulate as a parametric sequence of NLP's
- Solution by repeated use of standard NLP software
- Southern Spars Company (NZ): improved from 5-0 to 5-2 in America's Cup!

Other new types of constraints

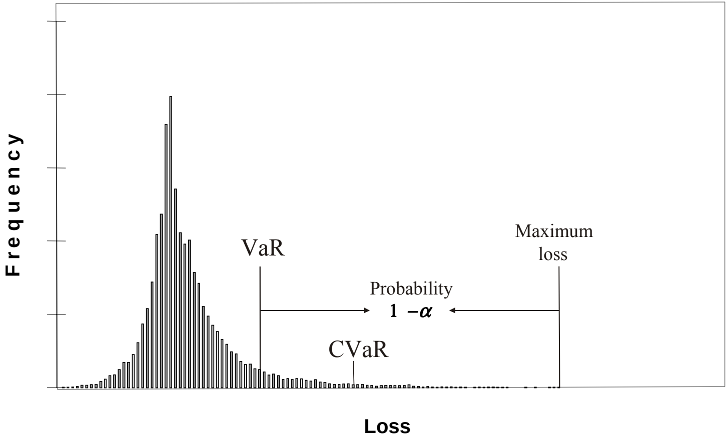
- range constraints $L \leq Ax - b \leq U$
- robust programming (probability constraints, stochastics)

$$f(x, \xi) \leq 0, \forall \xi \in \mathcal{U}$$

- conic programming $a_i^T x - b_i \in K_i$
- soft constraints
- rewards and penalties

Some constraints can be reformulated easily, others not!

CVaR constraints: mean excess dose (radiotherapy)



Move mean of tail to the left!

ENLP (Rockafellar): Primal problem

$$\min_{x \in X} f_0(x) + \theta(f_1(x), \ldots, f_m(x))$$

"Classical" problem:

$$\begin{array}{ll} \min\limits_{x_1, x_2, x_3} & \exp(x_1) \\ \text{s.t.} & \log(x_1) = 1 \\ & x_2^2 \le 2 \\ & x_1/x_2 = \log(x_3), 3x_1 + x_2 \le 5, x_1 \ge 0, x_2 \ge 0 \end{array}$$

Soft penalization of red constraints:

$$\begin{array}{ll} \min\limits_{x_1, x_2, x_3} & \exp(x_1) + 5 \left\| \log(x_1) - 1 \right\|^2 + 2 \max(x_2^2 - 2, 0) \\ \text{s.t.} & x_1/x_2 = \log(x_3), 3x_1 + x_2 \le 5, x_1 \ge 0, x_2 \ge 0 \end{array}$$

ENLP: Primal problem

$$\min_{x \in X} f_0(x) + \theta(f_1(x), \ldots, f_m(x))$$

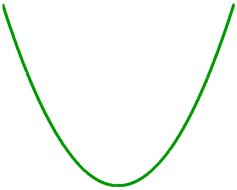
$$X = \left\{ x \in \mathbf{R}^3 : 3x_1 + x_2 \leq 5, x_1 \geq 0, x_2 \geq 0 \right\}$$

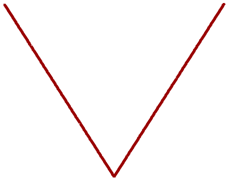
$$f_1(x) = \log(x_1) - 1, f_2(x) = x_2^2 - 2, f_3(x) = x_1/x_2 - \log(x_3)$$

$$\theta_1(u) = 5 \left\| u \right\|^2, \theta_2(u) = 2 \max(u, 0), \theta_3(u) = \psi_{\{0\}}(u)$$

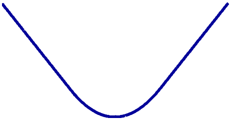
θ nonsmooth due to the max term; θ separable in example.

Examples of different θ





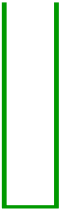
but solution reformulations are very different



$$\theta(u) = \begin{cases} \gamma u - \frac{1}{2}\gamma^2 & \text{if } u \geq \gamma \\ \frac{1}{2}u^2 & \text{if } u \in [-\gamma, \gamma] \\ -\gamma u - \frac{1}{2}\gamma^2 & \text{else} \end{cases}$$

Huber function used in robust statistics.

More general θ functions



In general any piecewise linear penalty function can be used: (different upside/downside costs). Also cone constraints. General form:

$$\theta(u) = \sup_{y \in Y} \{y' u - k(y)\}$$

θ can take on ∞ and may be nonsmooth; it is convex.

Specific choices of k and Y

$$\theta(u) = \sup_{y \in Y} \{ y' u - k(y) \}$$

- L_2: $k(y) = \frac{1}{4\lambda} y^2$, $Y = (-\infty, +\infty)$
- L_1: $k(y) = 0$, $Y = [-\rho, \rho]$
- L_∞: $k(y) = 0$, $Y = \Delta$, unit simplex
- Huber: $k(y) = \frac{1}{4\lambda} y^2$, $Y = [-\rho, \rho]$
- Second order cone constraint: $k(y) = 0$, $Y = C^\circ$

Elegant Duality

For these θ (defined by $k(\cdot)$, Y), duality is derived from the Lagrangian:

$$\mathcal{L}(x, y) = f_0(x) + \sum_{i=1}^{m} y_i f_i(x) - k(y)$$
$$x \in X, y \in Y$$

- Dual variables in Y not simply ≥ 0 or free.
- Saddle point theory, under convexity.
- Dual Problem and Complete Theory.
- Special case: ELQP - dual problem is also an ELQP.

Implementation: convert tool

```
$echo nlp2mcp > convert.opt

e1.. obj =e= exp(x1);
e2.. log(x1)-1 =e= 0;
e3.. sqr(x2)-2 =e= 0;
e4.. x1/x2 =e= log(x3);
e5.. 3*x1 + x2 =l= 5;

$onecho > enlpinfo.scr
e2 sqr 5
e3 plus 2
$offecho

solve mod using nlp min obj;
```

Library of different θ functions implemented.

First order conditions

- Solution via reformulation. One way:

$$\begin{array}{rrcl} 0 \in & \nabla_x \mathcal{L}(x,y) & + & N_X(x) \\ 0 \in & -\nabla_y \mathcal{L}(x,y) & + & N_Y(y) \end{array}$$

  $N_X(x)$ is the normal cone to the closed convex set X at x.

- Automatically creates an MCP:

```
model enlp /   gradLx.x,
               -gradLy.y /;
solve enlp using mcp;
```

- Already available!
- To do: extend X and Y beyond simple bound sets.

Alternative Reformulations

Convert does symbolic/numeric reformulations. Alternative NLP formulations also possible.

$$k(y) = \frac{1}{2} y' Q y, \quad X = \{x : Rx \leq r\}, \quad Y = \{y : S'y \leq s\}$$

Defining

$$Q = DJ^{-1}D', \quad F(x) = (f_1(x), \ldots, f_m(x))$$

$$\begin{array}{ll} \min & f_0(x) + s'z + \frac{1}{2} wJw \\ \text{s.t.} & Rx \leq r, z \geq 0, F(x) - Sz - Dw = 0 \end{array}$$

Can set up better (solver) specific formulation.

Embedded models

- Bilevel programs:

$$
\begin{aligned}
\min_{x,y} \quad & f(x,y) \\
\text{s.t.} \quad & g(x,y) \le 0, \\
& y \text{ solves } \min_{s} v(x,s) \text{ s.t. } h(x,s) \le 0
\end{aligned}
$$

- A different embedded model that arises frequently is:

$$
\begin{aligned}
\min_{x} \quad & f(x,y) \\
\text{s.t.} \quad & g(x,y) \le 0 \quad (\perp \lambda \ge 0) \\
& H(x,y,\lambda) = 0 \qquad (\perp y \text{ free})
\end{aligned}
$$

Example

- Nash Games: x^* is a Nash Equilibrium if

$$x_i^* \in \arg\min_{x_i \in X_i} \ell_i(x_i, x_{-i}^*, q), \forall i \in \mathcal{I}$$

x_{-i} are the decisions of other players.

- Quantities q given exogenously, or via complementarity:

$$0 \leq H(x, q) \perp q \geq 0$$

- Convert reformulates automatically for appropriate solvers, e.g. forms KKT conditions

Discrete-Time Finite-State Stochastic Games

Central tool in analysis of strategic interactions among forward-looking players in dynamic environments

Example: The Ericson & Pakes (1995) model of dynamic competition in an oligopolistic industry

Exactly in the format described above.

Applications

- Advertising (Doraszelski & Markovich 2007)
- Capacity accumulation (Besanko & Doraszelski 2004,...)
- Collusion (Fershtman & Pakes 2000, 2005, de Roos 2004)
- Consumer learning (Ching 2002)
- Firm size distribution (Laincz & Rodrigues 2004)
- Learning by doing (Benkard 2004,...)
- Mergers (Berry & Pakes 1993, Gowrisankaran 1999)
- Network externalities (Jenkins et al 2004,...)
- Productivity growth (Laincz 2005)
- R&D (Gowrisankaran & Town 1997,...)
- Technology adoption (Schivardi & Schneider 2005)
- International trade (Erdem & Tybout 2003)
- Finance (Goettler, Parlour & Rajan 2004,...).

Results

S	Var	rows	non-zero	dense(%)	Steps	RT (m:s)
20	2400	2568	31536	0.48	5	0 : 03
50	15000	15408	195816	0.08	5	0 : 19
100	60000	60808	781616	0.02	5	1 : 16
200	240000	241608	3123216	0.01	5	5 : 12

Convergence for $S = 200$

Iteration	Residual
0	1.56(+4)
1	1.06(+1)
2	1.34
3	2.04(−2)
4	1.74(−5)
5	2.97(−11)

Conclusions

- Complementarity constraints within optimization problems
- Practical/usable implementation of Rockafellar's ENLP approach within a modeling system
- System can easily formulate and solve second order cone programs, robust optimization, soft constraints via piecewise linear penalization (with strong supporting theory)
- Embedded optimization models reformulated for appropriate solution engine
- Enhance library of (implemented) θ functions
- Exploit structure of θ in solvers
- Extend complementarity solvers to VI solvers