Novel Problem Types and Algorithms

By
Arne Stolbjerg Drud
adrud@arki.dk
On behalf of GAMS Development Corp

Background

Objective:

- Move New Model Types / Algorithm Types from a Research Environments to a General User Environment.
- Help new tools become mature.
- Utilize Cross fertilization

Requirements for the **Research Environment**:
Realistic problems / models that are challenging but not unrealistic given the current level of theory and technology.

Requirements for a **General User Environment**:
Tools that can solve interesting problems, preferably without too much expert / algorithm specific knowledge.

Examples:

- MPEC: Mixed Complementarity Problems with Equilibrium Constraints.
- MINLP: Mixed Integer Nonlinear Programming
- GLOBAL: Global vs. Local for NLP and MINLP.

MPEC:

What is an MPEC model?

The Mathematical formulation of an MPEC model is

$$
\begin{aligned}
& \text{Max or Min } f(x,y) \\
\text{s.t.} \quad & g(x,y) \leq 0 \\
& 0 \leq h(x,y) \perp y \geq 0
\end{aligned}
$$

where $\perp$ means that h must be "complementary" to y, i.e. either h_i or y_i is zero for each index i. General lower and upper bounds on the variables can be handled using similar complementarity concepts.

The h(x,y) constraints represent the optimality conditions for the solution of a sub-model where y is the set of variables and x is considered a fixed vector.

x will often represent the decision of a leader or top level decision maker and y will represent the decision of the follower or lower level decision maker.

y is a function of x (the lower level decides y when x is known). However, y is in the general case not a nice function of x.

MPEC:

Where do MPEC models come from?

MPEC models are very natural in economic analysis (government decisions vs. the decisions of optimizing agents, for example optimal tariff calculations). Objective can be utility maximization subject to revenue constraints (g constraints).

Traffic models: Traffic control mechanisms (toll prices and locations) vs. the free travel decisions of the consumers. Objective can be minimization of congestion or delays.

Optimal Design: Find a design (upper level) that can withstand the worst case load within a set of possible loads (lower level model).

Inverse Problems in Structural Engineering: Find some design parameters (upper level) that can explain observed cracks in a structure, defined through the lower level model.

MPEC:

How are MPEC models solved today?

Difficulty: y(x) is a non-smooth (potentially one-to-many mapping). Try Smoothing.

Theory: The basic complementarity constraint can be defined through a complementarity function with the following property:
$\Phi(r,t) = 0 \Leftrightarrow 0 \leq r \perp t \geq 0$

Examples of such functions are

- $\Phi(r,t) = \min(r,t)$
- $\Phi(r,t) = (r^2 + t^2)^{1/2} - r - t$

The functions are in general smooth for r and t both strictly positive but non-smooth when r or t is zero.

The complementarity constraints are smoothed through the use of some perturbed complementarity functions. Examples are:

- $\Phi(r,t,\mu) = (r^2 + t^2 + \mu)^{1/2} - r - t = 0$
- $\Phi(r,t,\mu) = r - \mu \log(1+\exp((r-s)/ \mu)) = 0$

where μ is a small number.

The smoothed version of the constraints $0 \leq h(x,y) \perp y \geq 0$ can be formulated in various ways:

- Directly as $\Phi(y,h(x,y),\mu) = 0$
- With an auxiliary variable $s = h(x,y)$ and the complementarity is equivalent to $\Phi(y,s,\mu) = 0$
- Each individual complementarity function can be set to zero or their sum can be set to zero.

The resulting overall model is a smooth NLP. μ is started at some positive value and $x(\mu)$ and $y(\mu)$ are traced as μ is reduced toward zero.

MPEC:

How are MPEC models solved today?

Practice: There are many choices to be made in the framework outlined above (Ferris et.al. list 23 alternative formulations), and the best method is not known. It probably depend on the model in some unknown way. Experimentation is therefore necessary.

How do we facilitate the general framework and the experimentation?

- GAMS/CONVERT will rewrite an MPEC model to an NLP model: The output is the source code of an NLP model.
- An option file controls the choice of Φ function, the type of reformulation, the initial value for μ, and an update method.
- The NLP solver is selected from among the standard GAMS solvers.
- MPECLib has over 90 models from the literature available for immediate experimentation

Challenges for NLP Solvers:

- Although the Φ functions are mathematically nice, they can be nasty from a numerical point of view.
- Local solutions can be a problem.
- Global solvers should be used on smaller models. However, we loose the continuation property from slow decrease of μ.

Challenges for GAMS:

- The Φ functions can be complex in terms of the existing GAMS operators. Repetition of complex arguments may require duplication of code.
- Numerically stable implementations can be hand-coded in a programming language but are not straight forward in a GAMS context.
- Efficient addition of new multi-argument functions is therefore an ongoing development activity at GAMS.

MINLP:

What is an MINLP?

$$
\begin{array}{ll}
 & \text{Max or Min } f(x,y) \\
\text{s.t.} & g(x,y) = 0 \\
 & l \leq x \leq u \\
 & y \text{ discrete (or binary)}
\end{array}
$$

where g has some nonlinear components.

Where do MINLP models come from?

Any NLP model with some discrete component becomes an MINLP model. They are very common in areas such as:

Engineering design
Fixed charge problems
Financial Optimization
and in many other applications

MINLP:

How are MINLP models solved today?

There are today two basically different approaches:

Outer Approximation:

- The model is linearized and the resulting MIP model is solved with on the todays efficient linear MIP solvers to give a suggestion for the discrete variable y.
- An NLP model is solved for fixed y.
- A new linearization is created using the new point and the process repeats until some stopping criteria is met.

Branch and Bound:

- The relaxed MINLP models is solve (y is considered continuous).
- If one of the y-components say y_i is not integer the model is split into two sub-models by restricting $y_{i.}$
- Sub-models are fathomed or solved in the same way as in Linear Branch and Bound. The (local) NLP solutions are used as bounds.

MINLP:

Outer Approximation:

Outer Approximation as implemented in **DICOPT** (Grossman et.al.) has been very important in demonstrating that MINLP models are a relevant tool in practice.

Outer Approximation works well when the model has a significant combinatorial component and the NLP models are fairly easy to solve. Any progress in MIP and NLP technology is automatically transferred to the solution of the MIP and NLP sub-problems.

Outer Approximation will not work well on models where many of the sub-NLP models are infeasible or fail. Failure can happen because the sub-NLP models can be difficult to solve: we may not have good initial values, and user intervention is not possible. The sub-MIP models may also be numerically difficult because the constraints are linearizations computed in ‘bad’ points.

MINLP:

Branch and Bound:

Branch and Bound requires the solution of a potentially very large number of NLP models and was therefore not considered viable initially. Progress in NLP technology and in more efficient restart of related sub-NLPs has made Branch and Bound an interesting alternative.

Branch and Bound works well on models with few discrete variables but significant nonlinear components. The gradual change of sub-models from the relaxed RMINLP to the more restricted sub-NLP nodes makes the solution of the sub-NLP relatively reliable and efficient.

Branch and Bound can be very slow on models with a significant combinatorial component. The number of sub-NLP become too large. The many developments in MIP (e.g. cuts) cannot easily be transferred to the nonlinear environment.

Implementations: SBB in GAMS, LINGO, Frontline etc.

MINLP:

Common challenges:

- What are good formulations?
- Can we develop preprocessing / cut generation using ideas from MIP?
- Use of alternative NLP solvers when sub-NLPs fail to solve. (Implemented in SBB. Can also be used with DICOPT).
- Combination of Outer Approximation and Branch and Bound.
- Local Solutions to the sub-NLP models.

With today’s technology and lack of preprocessing, good formulations seems to be more important in MINLP than in MIP.

GLOBAL:

We are concerned with the **Global Optima** of models with Nonlinear Components.

All NLP and MINLP models are in principle Global Optimization problems. We have in the past accepted "Locally Optimal" solutions simply because we could not compute Global Optima.

The development of new methods is changing this.

GLOBAL:

Methods (currently available or in progress with GAMS):

BARON (Branch and Reduce Optimization Navigator): Uses convex under-estimators and convex relaxations to compute guarantied bounds in a Branch and Bound context. Works on NLPs and MINLPs. The size of models solvable with **BARON** is limited because of the rigorous framework.

OQNLP (OptQuest Multi-start method with standard NLP sub-solvers): Uses structured search methods to generate intelligent initial points to standard local NLP solvers. Will generate good solutions, but without guaranties. Good stopping criteria are needed. Works on NLPs and MINLPs. **OQNLP** can be used for larger models than BARON.

LGO (Lipschitz Global Optimization): Global Optimizer based on Lipschitz continuity. Gives statistical measures of the quality of the solution. Works on NLPs. Experience with **LGO** is still limited.

Common features of MPEC World, MINLP World, and GLOBAL World at www.gamsworld.org

- Libraries of Models of various sizes and degrees of difficulty. Models are both from academic and commercial sources. Has suggested initial points and points with best know solution for many models.

- Translation server that can translate the models into the format for many systems such as AMPL, LINGO, MINOPT and others.

- Tools for running collections of models and analyzing results (Performance World).

- Bibliography.

- Discussion lists.

Use of the models, submission of new models and of better solutions, as well as participation in discussions in general are invited.