A Tutorial on Non-Convex Mixed-Integer Non-Linear Programming

Adam N. Letchford[1]

Department of Management Science
Lancaster University Management School

April 2011

[1]funded by the EPSRC under grant EP/D072662/1

Outline

Introduction

Definition

A Mixed-Integer Non-Linear Programme (MINLP) is a problem of the following form:

$$\min \left\{ f^0(x,y) : \ f^j(x,y) \geq 0 \, (j = 1, \ldots, m), \ x \in \mathbb{Z}_+^{n_1}, \ y \in \mathbb{R}_+^{n_2} \right\},$$

where n_1 is the number of integer-constrained variables, n_2 is the number of continuous variables, m is the number of constraints, and $f^j(x,y)$ for $j = 0, \ldots, m$ are arbitrary functions mapping $\mathbb{Z}_+^{n_1} \times \mathbb{R}_+^{n_2}$ to the reals.

Introduction (cont.)

Definition

A MINLP is convex if all of the functions $f^j(x,y)$ for $j = 0, \ldots, m$ are all convex; otherwise it is non-convex.

In this talk, we focus on the non-convex case.

Applications

Several important practical problems are most naturally modelled as non-convex MINLPs. For example:

- design of water, gas or electricity networks
- design of chemical processes
- scheduling gas- or coal-fired power stations
- some portfolio optimisation problems
- some cutting and packing problems
- some facility selection, location and layout problems
- various problems in computational chemistry/biology.

Complexity

Non-convex MINLPs are typically much harder to solve (to proven optimality) than convex ones.

Indeed, given a convex MINLP, one can compute an initial lower bound simply by solving the continuous relaxation of the problem. This relaxation will be a convex NLP, which is likely to be relatively easy to solve.

The continuous relaxation of a non-convex MINLP, on the other hand, is a non-convex NLP. Non-convex NLPs (sometimes called global optimisation problems) are themselves $\mathcal{NP}$-hard!

Complexity (cont.)

In fact, non-convex MINLPs are worse than $\mathcal{NP}$-hard:

Theorem (Jeroslow, 1973)

Testing whether there exists an integer solution to a system of non-convex quadratic constraints is undecidable.

(If however each variable is lower- and upper-bounded explicitly, then non-convex MINLPs become 'merely' $\mathcal{NP}$-hard.)

Complexity (cont.)

Moreover, it is not easy to devise effective heuristics for non-convex MINLP.

Indeed, even finding a feasible solution can be surprisingly tricky in some cases.

Later on, we will see that some exact methods for convex MINLP can be converted into heuristics for non-convex MINLP.

But for now, we concentrate on exact approaches.

Branch-and-Reduce

Branch-and-Reduce (Tawarmalani & Sahinidis, 2002) is an elegant exact technique for both global optimisation and non-convex MINLP.

The first step is to construct an LP relaxation of the problem.

This is done by introducing additional variables representing nonlinear terms, along with valid linear constraints linking them to the original variables.

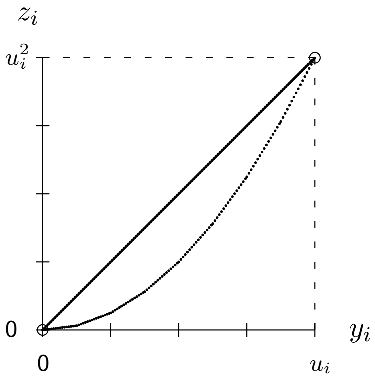
Branch-and-Reduce (cont.)

For example, suppose that the real variable y_i is known to satisfy $0 \leq y_i \leq u_i$, and that the quadratic term y_i^2 appears in one of the nonlinear functions.

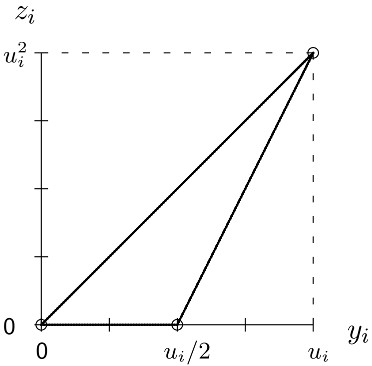
We could:

- introduce a new variable, say z_i, representing y_i^2
- introduce the valid linking constraints $0 \leq z_i \leq u_i y_i$ and $z_i \geq u_i^2 - 2u_i y_i$.

True relationship between y_i and z_i



The linear outer-approximation to the relationship



Branch-and-Reduce (cont.)

Now, if the solution of the LP relaxation is not feasible, we can branch in one of two ways:

- Branch on an integer variable x_i in the usual way.
- Branch on a *continuous* variable y_i, splitting its domain into two intervals.

However, when we branch, we don't just add one extra constraint, as in MILP or convex MINLP.

Instead, the entire LP relaxation is modified, so as to reduce the feasible region as much as possible.

Branch-and-Reduce (cont.)

For example, suppose that we branch on the real variable y_i according to the disjunction:

$$(0 \leq y_i \leq \alpha) \vee (\alpha \leq y_i \leq u_i).$$

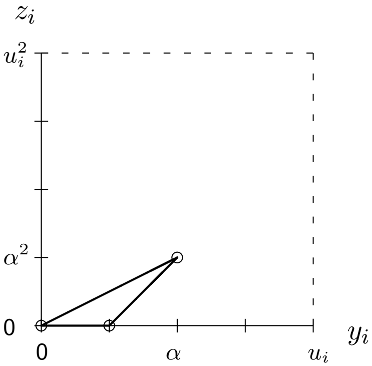
Then, in the left branch we can tighten the relaxation by adding:

$$z_i \leq \alpha y_i \text{ and } z_i \geq \alpha^2 - 2\alpha y_i,$$

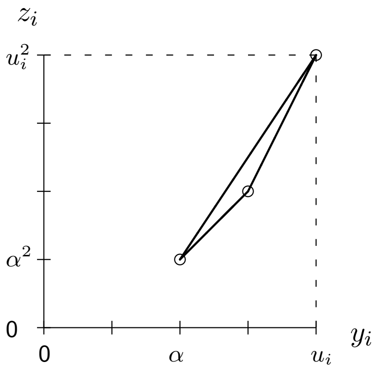
and in the right branch we can tighten the relaxation by adding:

$$\alpha y_i \leq z_i \leq u_i y_i.$$

The left branch



The right branch



Special Cases I: Separable Functions

Definition

A function $f(x, y)$ is called separable if it can be written as:

$$\sum_{i=1}^{n_1} g^i(x_i) + \sum_{i=1}^{n_2} h^i(y_i),$$

where $g^1(\cdot), \ldots, g^{n_1}(\cdot)$ and $h^1(\cdot), \ldots, h^{n_2}(\cdot)$ are univariate functions.

Definition

An MINLP is called separable if all of the functions $f^0(x, y), \ldots, f^m(x, y)$ are themselves separable.

Special Cases I (cont.)

Beale & Tomlin (1970) proposed to tackle separable MINLPs as follows:

1. approximate each of the univariate functions by a piece-wise linear function
2. introduce one auxiliary binary variable for each piece
3. reformulate the problem as an MILP.

Vielma & Nemhauser (2008) presented an elegant way to substantially reduce the number of auxiliary binary variables.

Special Cases II: Quadratic Functions

A lot of attention has been given to the case in which all objective and constraint functions are quadratic.

Glover (1975) showed that a quadratic function of n binary variables can be linearised using $\mathcal{O}(n)$ additional variables and constraints. (Several variants of this have appeared since.)

Hammer & Rubin (1970) and Körner (1988, 1990) showed that non-convex 0-1 QPs can be convexified using identities of the form $x_i = x_i^2$.

This approach has been generalised to non-convex MIQP (Billionnet *et al.*, 2009) and non-convex MIQCQP (Galli & L., 2011).

Special Cases II (cont.)

A more popular approach is to introduce $\mathcal{O}(n^2)$ additional variables, representing products of original variables.

Much attention has been given to unconstrained 0-1 QP (e.g., Padberg, 1989; Barahona, 1989; De Simone, 1989; Deza & Laurent, 1996).

Some has been done on specific constrained 0-1 QPs, such as the quadratic assignment problem (Jünger & Kaibel, 2001; Padberg & Rijal, 1996, etc.) and the quadratic knapsack problem (Helmberg *et al.*, 2000).

Powerful techniques for general 0-1 QP were provided by Sherali & Adams (1990) and Lovász & Schrijver (1991).

Special Cases II (cont.)

Observe that, if x is any vector of decision variables, then the matrix

$$Y = \begin{pmatrix} 1 \\ x \end{pmatrix} \begin{pmatrix} 1 \\ x \end{pmatrix}^T = \begin{pmatrix} 1 & x^T \\ x & xx^T \end{pmatrix}$$

is positive semidefinite.

This leads to SDP relaxations (Sherali & Adams, 1990; Lovász & Schrijver, 1991; Kojima & Tuncel, 1999; etc.)

Moreover, if the variables are non-negative, Y will be completely positive, not just psd.

This has inspired even stronger relaxations (e.g., Lasserre, 2001; Burer, 2006).

Special Cases II (cont.)

Finally, Saxena *et al.* (2010), Burer & L. (2010) and Galli *et al.* (2011) have derived strong cutting planes for general non-convex MIQP.

Saxena *et al.* do this using disjunctive programming (a.k.a. lift-and-project) techniques.

Burer & L. study the convex hull of feasible solutions directly, using a combination of polyhedral theory and convex analysis.

Galli *et al.* show how to adapt the 'gap inequalities', originally defined by Laurent & Poljak (1996) for the max-cut problem, to non-convex MIQP.

Special Cases III: Polynomial Functions

Recently, some sophisticated approaches have been developed for mixed 0-1 polynomial programs.

These draw on concepts from real algebraic geometry, commutative algebra and moment theory.

Relevant works include Nesterov (2000), Parrilo (2000), Lasserre (2001), Laurent (2003), De Loera *et al.* (2008).

Software

There are three software packages that can solve non-convex MINLPs to proven optimality, using branch-and-reduce:

`BARON`, `Alpha-BB` and `Couenne`.

Some packages for convex MINLP can be used to find *heuristic* solutions for non-convex MINLP:

`BONMIN`, `DICOPT` and `LaGO`.

Finally, `GloptiPoly` can solve general polynomial optimisation problems.

Concluding Remarks

- Non-convex MINLP has many important practical applications.
- It is harder to solve than convex MINLP, in both theory and practice.
- Branch-and-reduce is the only available approach for the general case.
- For some special cases, one can do better.
- There is still a clear need for improved theory, algorithms and software.
- Nobody seems to have looked at stochastic variants.