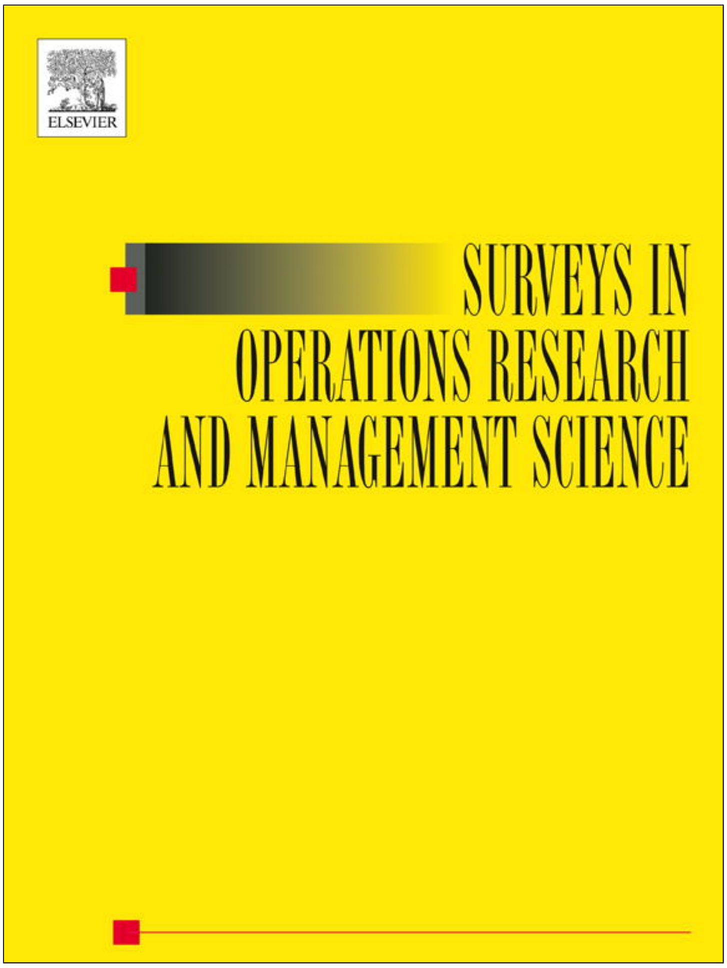

ELSEVIER
SURVEYS IN
OPERATIONS RESEARCH
AND MANAGEMENT SCIENCE

Surveys in Operations Research and Management Science 17 (2012) 97–106

Contents lists available at SciVerse ScienceDirect

Surveys in Operations Research and Management Science

journal homepage: www.elsevier.com/locate/sorms

Review

Non-convex mixed-integer nonlinear programming: A survey

Samuel Burer [a,*], Adam N. Letchford [b]

[a] Department of Management Sciences, Tippie College of Business, University of Iowa, United States
[b] Department of Management Science, Lancaster University, United Kingdom

ARTICLE INFO

Article history:
Received 9 July 2012
Accepted 8 August 2012

ABSTRACT

A wide range of problems arising in practical applications can be formulated as Mixed-Integer Nonlinear Programs (MINLPs). For the case in which the objective and constraint functions are convex, some quite effective exact and heuristic algorithms are available. When non-convexities are present, however, things become much more difficult, since then even the continuous relaxation is a global optimization problem. We survey the literature on non-convex MINLPs, discussing applications, algorithms, and software. Special attention is paid to the case in which the objective and constraint functions are quadratic.

Contents

1. Introduction

A *Mixed-Integer Nonlinear Program* (MINLP) is a problem of the following form:

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

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 $j = 0, 1, \dots, m$ are arbitrary functions mapping $\mathbb{Z}_+^{n_1} \times \mathbb{R}_+^{n_2}$ to the reals.

MINLPs constitute a very general class of problems, containing as special cases both *Mixed-Integer Linear Programs* or MILPs (obtained when the functions $f^0, \dots, f^m$ are all linear) and *Nonlinear Programs* or NLPs (obtained when $n_1 = 0$). This generality enables

* Corresponding author. Tel.: +1 3193350931.
E-mail addresses: samuel-burer@uiowa.edu (S. Burer), A.N.Letchford@lancaster.ac.uk (A.N. Letchford).

1876-7354/$ – see front matter
doi:10.1016/j.sorms.2012.08.001

one to model a very wide range of problems, but it comes at a price: even very special kinds of MINLP usually turn out to be $\mathcal{NP}$-hard.

It is useful to make a distinction between two kinds of MINLP. If the functions $f^0, \ldots, f^m$ are all *convex*, the MINLP is itself called *convex*; otherwise it is called *non-convex*. Although both kinds of MINLP are $\mathcal{NP}$-hard in general, convex MINLPs are much easier to solve than non-convex ones, in both theory and practice.

To see why, consider the *continuous relaxation* of an MINLP, which is obtained by relaxing the integrality condition from $x \in \mathbb{Z}^n_+$ to $x \in \mathbb{R}^n_+$. In the convex case, the continuous relaxation is itself convex, and therefore likely to be tractable, at least in theory. A variety of quite effective exact solution methods for convex MINLPs have been devised based on this fact. Examples include generalized Benders' decomposition [1], branch-and-bound [2], outer approximation [3], LP/NLP-based branch-and-bound [4], the extended cutting-plane method [5], branch-and-cut [6], and the hybrid methods described in [7,8]. These methods are capable of solving instances with hundreds or even thousands of variables.

By contrast, the continuous relaxation of a non-convex MINLP is itself a global optimization problem, and therefore likely to be $\mathcal{NP}$-hard (see, e.g., [9,10]). In fact, the situation is worse than this. Several simple cases of non-convex MINLPs, including the case in which all functions are quadratic, all variables are integer constrained, and the number of variables is fixed, are known to be not only $\mathcal{NP}$-hard, but even *undecidable* [11]. We refer the reader to the excellent surveys [12,13] for details.

As it happens, all of the proofs that non-convex MINLPs can be undecidable involve instances with an unbounded feasible region. Fortunately, in practice, the feasible region is usually bounded, either explicitly or implicitly. Nevertheless, the fact remains that some relatively small non-convex MINLPs, with just tens of variables, can cause existing methods to run into serious difficulties.

Several good surveys on MINLPs are available, e.g., [14–17, 12,18]. They all cover the convex case, and some cover the non-convex case. There is even research on the *pseudo-convex* case [19], involving non-convex functions that nevertheless have convex level sets. In this survey, on the other hand, we concentrate on the non-convex case. Moreover, we pay particular attention to a special case that has attracted a great deal of attention recently, and which is also of interest to ourselves: namely, the case in which all of the nonlinear functions involved are quadratic. We note that the quadratic case actually subsumes the case when all functions f^j are polynomials, although there may be substantial overhead when expressing a polynomial program as a quadratic one (see the beginning of Section 5 for details).

The paper is structured as follows. In Section 2, we review some applications of non-convex MINLPs. In Section 3, we review the key ingredients of most exact methods, including convex under-estimating functions, separable functions, factorization of non-separable functions, and standard branching versus spatial branching. In Section 4, we then show how these ingredients have been used in a variety of exact and heuristic methods for general non-convex MINLPs. Next, in Section 5, we cover the literature on the quadratic and polynomial cases. In Section 6, we list some of the available software packages, and, in Section 7, we end the survey with a few brief conclusions and topics of current and future research.

2. Applications

Many important practical problems are naturally modeled as non-convex MINLPs. We list a few examples here and recommend the references provided for further details and even more applications.

The field of chemical engineering gives rise to a plethora of non-convex MINLPs. Indeed, some of the first and most influential research in MINLPs has occurred in this field. For example, Grossmann and Sargent [20] discuss the design of chemical plants that use the same equipment "in different ways at different times". Misener and Floudas [21] survey the so-called pooling problem, which investigates how best to blend raw ingredients in pools to form the desired output. Luyben and Floudas [22] analyze the simultaneous design and control of a process, and Yee and Grossmann [23] examine heat exchanger networks in which heat from one process is used by another. See Floudas [24] and Misener and Floudas [25] for comprehensive lists of references of MINLPs arising in chemical engineering.

Another important source of non-convex MINLPs is network design. This includes, for example, water [26], gas [27], energy [28], and transportation [29] networks.

Non-convex MINLPs arise in other areas of engineering as well. These include avoiding trim-loss in the paper industry [30], airplane boarding [31], oil-spill response planning [32], ethanol supply chains [33], concrete structure design [34], and load-bearing thermal insulation systems [35]. There are also medical applications, such as seizure prediction [36].

Adams and Sherali [37] and Freire et al. [38] discuss applications of MINLPs with non-convex bilinear objective functions in production planning, facility location, distribution, and marketing.

Finally, many standard and well-studied optimization problems, each with its own selection of applications, can also be viewed quite naturally as non-convex MINLPs. These include, for example, maximum cut (or binary quadratic programming (QP)) and its variants [39–41], clustering [42], non-convex QP with binary variables [43], quadratically constrained QP [44], the quadratic traveling salesman problem (TSP) [45], TSP with neighborhoods [46], and polynomial optimization [47].

3. Key concepts

In this section, some key concepts are presented, which together form the main ingredients of all existing exact algorithms (and some heuristics) for non-convex MINLPs.

3.1. Under- and over-estimators

As mentioned above, even solving the continuous relaxation of a non-convex MINLP is unlikely to be easy. For this reason, a further relaxation step is usual. One way to do this is to replace each non-convex function $f^j(x, y)$ with a *convex under-estimating function*, i.e., a convex function $g^j(x, y)$ such that $g^j(x, y) \le f^j(x, y)$ for all (x, y) in the domain of interest. Another way is to define a new variable, say z^j, which acts as a place holder for $f^j(x, y)$, and to add constraints which force z^j to be approximately equal to $f^j(x, y)$. In this latter approach, one adds constraints of the form $z^j \ge g^j(x, y)$, where $g^j(x, y)$ is again a convex under-estimator. One can also add constraints of the form $z^j \le h^j(x, y)$, where $h^j(x, y)$ is a *concave over-estimating* function. If one wishes to solve the convex relaxation using an LP solver, rather than a general convex programming solver, one must use linear under- and over-estimators.

For some specific functions, and some specific domains, one can characterize the so-called *convex* and *concave envelopes*, which are the tightest possible convex under-estimator and concave over-estimator. A classical example, due to McCormick [48], concerns the quadratic function y_1y_2, over the rectangular domain defined by $\ell_1 \le y_1 \le u_1$ and $\ell_2 \le y_2 \le u_2$. If z denotes the additional variable, the convex envelope is defined by the two linear inequalities $z \ge \ell_2 y_1 + \ell_1 y_2 - \ell_1\ell_2$ and $z \ge u_2y_1 + u_1y_2 - u_1u_2$, and the concave envelope by $z \le u_2y_1 + \ell_1y_2 - \ell_1u_2$ and $z \le \ell_2y_1 + u_1y_2 - u_1\ell_2$. In this case, both envelopes are defined using only linear constraints.

Many other examples of under- and over-estimating functions, and convex and concave envelopes, have appeared in the literature. See the books by Horst and Tuy [49] and Tawarmalani and Sahinidis [10] for details.

We mention another important paper, that of Androulakis et al. [50]. Their approach constructs convex under-estimators of general, twice-differentiable, non-convex functions whose domain is a box (also known as a hyper-rectangle). The basic idea is to add a convex quadratic term that takes the value zero on the corners of the box, and the choice of the quadratic term is governed by a vector $\alpha \geq 0$. For example, suppose the function $f(x)$ is defined on $B := \{x : 0 \leq x \leq e\}$, where e is the all ones vector, and let $\mathrm{Diag}(\alpha)$ be the diagonal matrix having α as its diagonal. Then $f(x)$ is under-estimated by

$$f_\alpha(x) := f(x) + x^T \mathrm{Diag}(\alpha)x - \alpha^T x,$$

since $x^T \mathrm{Diag}(\alpha)x - \alpha^T x \leq 0$ for all $x \in B$ (and in fact equals zero on the corners of B). If α is chosen large, then $f_\alpha(x)$ will also be convex because its Hessian will be dominated by $\mathrm{Diag}(\alpha)$. On the other hand, as α increases, the quality of the resulting under-estimation by $f_\alpha(x)$ worsens, so the choice of α is critical.

3.2. Separable functions

A function $f(x, y)$ is said to be *separable* if there exist functions $g(x_i)$ for $i = 1, \ldots, n_1$ and functions $h(y_i)$ for $i = 1, \ldots, n_2$ such that

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

Separable functions are relatively easy to handle in two ways. First, if one has a useful convex under-estimator for each of the individual functions $g(x_i)$ and $h(y_i)$, the sum of those individual under-estimators is an under-estimator for $f(x, y)$. The same applies to concave over-estimators. Second, even if one does not have useful under- or over-estimators, one can use the following approach, due to Beale [51] and Tomlin [52].

1. Approximate each of the functions $g(x_i)$ and $h(y_i)$ with a piece-wise linear function.
2. Introduce new continuous variables, g_i and h_i, representing the values of these functions.
3. Add one binary variable for each 'piece' of each piece-wise linear function.
4. Add further binary variables, along with linear constraints, to ensure that the variables g_i and h_i take the correct values.

In this way, any non-convex MINLP with separable functions can be approximated by an MILP.

3.3. Factorization

If an MINLP is not separable, and it contains functions for which good under- or over-estimators are not available, one can often apply a process called *factorization*, also due to McCormick [48]. Factorization involves the introduction of additional variables and constraints, in such a way that the resulting MINLP involves functions of a simpler form.

Rather than presenting a formal definition, we give an example (see [53] for more details). Suppose an MINLP contains the (non-linear and non-convex) function $f(y_1, y_2, y_3) = \exp\left(\sqrt{y_1 y_2 + y_3}\right)$, where y_1, y_2, y_3 are continuous and non-negative variables. If one introduces new variables w_1, w_2, and w_3, along with the constraints $w_1 = \sqrt{w_2}$, $w_2 = w_3 + y_3$, and $w_3 = y_1 y_2$, one can rewrite the function f as $\exp(w_1)$. Then, one needs under- and over-estimators only for the relatively simple functions $\exp(w_1)$, $\sqrt{w_2}$, and $y_1 y_2$.

3.4. Branching: standard and spatial

The branch-and-bound method for MILPs, usually attributed to Land and Doig [54], is well known. The key operation, called *branching*, is based on the following idea. If an integer-constrained variable x_i takes a fractional value x_i^* in the optimal solution to the continuous relaxation of a problem, then one can replace the problem with two subproblems. In one of the subproblems, the constraint $x_i \leq \lfloor x_i^* \rfloor$ is added, and, in the other, the constraint $x_i \geq \lceil x_i^* \rceil$ is added. Clearly, the solution to the original relaxation is not feasible for either of the two subproblems.

In the global optimization literature, one branches by partitioning the domain of *continuous* variables. Typically, this is done by taking a continuous variable y_i, whose current domain is $[\ell_i, u_i]$, choosing some value β with $\ell_i < \beta < u_i$, and creating two subproblems, one with domain $[\ell_i, \beta]$ and the other with domain $[\beta, u_i]$. In addition, when solving either of the subproblems, one can replace the original under- and over-estimators with stronger ones, which take advantage of the reduced domain. This process, called 'spatial' branching, is necessary for two reasons: (i) the optimal solution to the relaxation may not be feasible for the original problem, and (ii) even if it is feasible, the approximation of the cost function in the relaxation may not be sufficiently accurate. Spatial branching is also due to McCormick [48].

We illustrate spatial branching with an example. Suppose that the continuous variable y_1 is known to satisfy $0 \leq y_1 \leq u_i$ and that, in the process of factorization, we have introduced a new variable z_i, representing the quadratic term y_i^2. If we intended to use a general convex programming solver, we could obtain a convex relaxation by appending the constraints $z_i \geq y_i^2$ and $z_i \leq u_i y_i$, as shown in Fig. 1(a). If, on the other hand, we preferred to use an LP solver, we could add instead the constraints $z_i \geq 0$, $z_i \geq u_i^2 - 2u_i y_i$, and $z_i \leq u_i y_i$, as shown in Fig. 1(b).

Now, suppose the solution of the relaxation is not feasible for the MINLP, and we decide to branch by splitting the domain of y_1 into the intervals $[0, \beta]$ and $[\beta, u_i]$. Also suppose for simplicity that we are using LP relaxations. Then, in the left branch we can tighten the relaxation by adding $\beta^2 - 2\beta y_i \leq z_i \leq \beta y_i$, while in the right branch we can add $\beta y_i \leq z_i \leq u_i y_i$ (see Fig. 2(a) and (b)).

Since MINLPs contain both integer-constrained and continuous variables, one is free to apply both standard branching or spatial branching where appropriate. Moreover, even if one applies standard branching, one may still be able to tighten the constraints in each of the two subproblems.

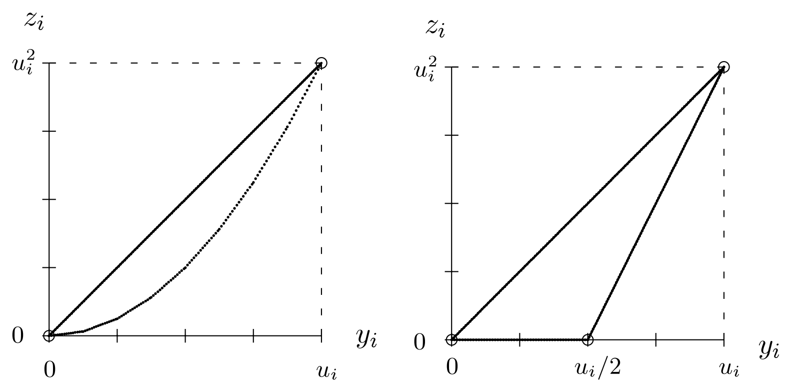
4. Algorithms for the general case

Now that we are armed with the concepts described in the previous section, we can go on to survey specific algorithms for general non-convex MINLPs.

4.1. Spatial branch-and-bound

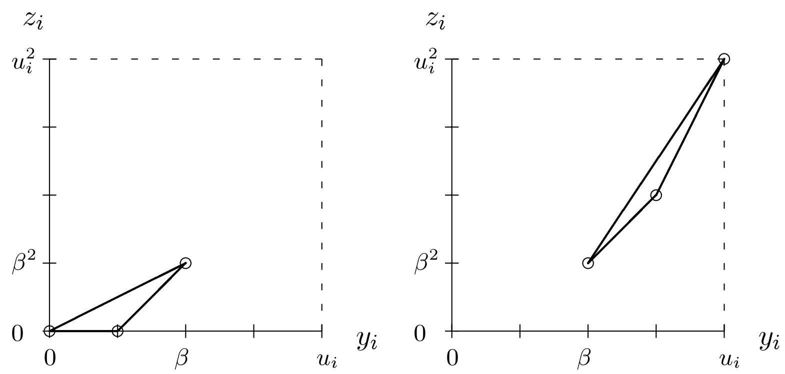
Branching, whether standard or spatial, usually has to be applied recursively, leading to a hierarchy of subproblems. As in the branch-and-bound method for MILPs [54], these subproblems can be viewed as being arranged in a tree structure, which can be searched in various ways. A subproblem can be removed from further consideration (also known as *fathomed* or *pruned*) under three conditions: (i) it is feasible for the original problem and its cost under the relaxed objective equals it true cost (to within some specified tolerance), (ii) the associated lower bound is no better than the best upper bound found so far, or (iii) it is infeasible.

This overall approach was first proposed by McCormick [48] in the context of global optimization problems. Later on, several authors (mostly from the chemical process engineering community) realized that the approach could be applied just as well to problems



(a) Convex. (b) Linear.

Fig. 1. Convex and linear approximations of the function $z_i = y_i^2$ over the domain $[0, u_i]$.



(a) Left. (b) Right.

Fig. 2. Improved linear approximations after spatial branching.

with integer variables. See, for example, Smith and Pantelides [55] or Lee and Grossmann [56].

4.2. *Branch-and-reduce*

A major step forward in the exact solution of non-convex MINLPs was the introduction of the *branch-and-reduce* technique by Ryoo and Sahinidis [57,58]. This is an improved version of spatial branch-and-bound in which one attempts to reduce the domains of the variables, beyond the reductions that occur simply as a result of branching. More specifically, one adds the following two operations: (i) before a subproblem is solved, its constraints are checked to see whether the domain of any variables can be reduced without losing any *feasible* solutions; (ii) after the subproblem is solved, sensitivity information is used to see whether the domain of any variables can be reduced without losing any *optimal* solutions.

After domain reduction has been performed, one can then generate better convex under-estimators. This in turn enables one to tighten the constraints, which can lead to improved lower bounds. The net effect is usually a drastic decrease in the size of the enumeration tree.

Branch-and-reduce is usually performed using LP relaxations, rather than more complex convex programming relaxations, due to two important facts. First and foremost, LPs can be solved more efficiently and with greater numerical stability. Second, sensitivity information is more readily available (and easier to interpret) in the case of LPs.

Tawarmalani and Sahinidis [59,60] added some further refinements to this scheme. In [59], a unified framework is given for domain reduction strategies, and, in [60], it is shown that, even when a constraint is convex, it may be helpful (in terms of tightness of the resulting relaxation) to introduce additional variables and split the constraint into two constraints. Some further enhanced rules for domain reduction, branching variable selection, and branching value have also been given by Belotti et al. [61].

4.3. *α-branch-and-bound*

Androulakis et al. [50] proposed an exact spatial branch-and-bound algorithm for global optimization of non-convex NLPs in which all functions involved are twice differentiable. This method, called α-BB, is based on their general technique for constructing under-estimators, which was mentioned in Section 3.1. In Adjiman et al. [62,63], the algorithm was improved by using tighter and more specialized under-estimators for constraints that have certain specific structures, and reserving the general technique only for constraints that do not have any of those structures. Later on, Adjiman et al. [64] extended the α-BB method to the mixed-integer case.

One advantage that α-BB has, with respect to the more traditional spatial branch-and-bound approach, or indeed branch-and-reduce, is that usually no additional variables are needed. That is to say, one can often work with the original objective and constraint functions, without needing to resort to factorization. This is because the under-estimators used do not rely on functions being factored. On the other hand, to solve the relaxations, one needs a general convex programming solver, rather than an LP solver.

4.4. *Conversion to an MILP*

Another approach that one can take is to factorize the problem (if necessary) as described in Section 3.3, approximate the resulting separable MINLP by an MILP as described in Section 3.2, and then solve the resulting MILP using any available MILP solver. To our knowledge, this approach was first suggested by Beale and Tomlin [65]. The conversion into an MILP leads to sets of binary variables with a certain special structure. Beale and Tomlin call these sets *special ordered sets of type* 2 (*SOS*2), and propose a specialized branching rule. This branching rule is now standard in most commercial and academic MILP solvers. Beale and Forrest [66] discuss a method for updating the MILP approximations dynamically and an improved branching strategy for the SOS2 variables.

Keha et al. [67] compare several different ways of modeling piece-wise linear functions (PLFs) using binary variables. In their follow-up paper [68], the authors present a branch-and-cut algorithm that uses the SOS approach in conjunction with strong valid inequalities. Vielma and Nemhauser [69] also present an elegant way to reduce the number of auxiliary binary variables required for modeling PLFs.

A natural way to generalize this approach is to construct PLFs that approximate functions of more than one variable. (In fact, this was already suggested by Beale [70] and Tomlin [52] in the context of non-convex NLPs.) A recent exploration of this idea was conducted by Martin et al. [27]. As well as constructing such PLFs, they also propose adding cutting planes to tighten the relaxation. Geißler et al. [71] is another recent reference.

Leyffer et al. [72] show that the naive use of PLFs can lead to an infeasible MILP, even when the original MINLP is clearly feasible. They propose a modified approach, called 'branch-and-refine', in which piece-wise-linear under- and over-estimators are constructed. This ensures that all of the original feasible solutions for the MINLP remain feasible for the MILP. Also, instead of branching spatially or on special ordered sets, they branch in the classical way. Finally, they refine the PLFs each time a subproblem is constructed.

4.5. Some other exact approaches

For completeness, we mention a few other exact approaches.

- Benson and Erenguc [73] and Bretthauer et al. [74] present exact algorithms for MINLPs with linear constraints and a concave objective function. Their algorithms use LP relaxations, specialized penalty functions, and cutting planes that are similar to the well-known concavity cuts of Tuy [75].
- Kesavan et al. [76] present special techniques for MINLPs in which separability occurs at the level of the vectors x and y, i.e., the functions $f^j(x, y)$ can be expressed as $g^j(x) + h^j(y)$. In fact, the authors assume that the functions $h^j(\cdot)$ are linear and that y is binary.
- Karuppiah and Grossman [77] use Lagrangian decomposition to generate lower bounds and cutting planes for general non-convex MINLPs.
- D'Ambrosio et al. [78] present an exact algorithm for MINLPs in which the non-convexities are solely manifested as the sum of non-convex univariate functions. In this sense, while the whole problem is not necessarily separable, the non-convexities are. Their algorithm, called `SC-MINLP`, involves an alternating sequence of convex MINLPs and non-convex NLPs.

4.6. Heuristics

All of the methods mentioned so far in this section have been *exact* methods. To close this section, we mention some *heuristic* methods, i.e., methods designed to find good, but not provably optimal, solutions quickly.

It is sometimes possible to convert exact algorithms for convex MINLPs into heuristics for non-convex MINLPs. Leyffer [79] does this using an MINLP solver that combines branch-and-bound with sequential quadratic programming. Nowak and Vigerske [80] do so by using quadratic under- and over-estimators of all nonlinear functions, together with an exact solver for convex all-quadratic problems.

Other researchers have adapted classical heuristic (and meta-heuristic) approaches, normally applied to 0–1 LPs, to the more general case of non-convex MINLPs. For example, Exler et al. [81] present a heuristic, based on tabu search, for certain non-convex MINLP instances arising in integrated systems and process control design. A particle-swarm optimization for MINLPs is presented in [82], [83] studies an enhanced genetic algorithm, and [84] considers an ant-colony approach. Two recent examples are that of Liberti et al. [85], whose approach involves the integration of variable neighborhood search, local branching, sequential quadratic programming, and branch-and-bound, and that of Berthold [86], who conducts large neighborhood local search by rounding the fractional solution from a relaxation. Finally, D'Ambrosio et al. [87] and Nannicini and Belotti [88] have recently presented heuristics that involve the solution of an alternating sequence of NLPs and MILPs.

5. The quadratic case (and beyond)

In this section, we focus on the case in which all of the non-linear objective and constraint functions are *quadratic*. This case has received much attention, not only because it is the most natural generalization of the linear case, but also because it has a very wide range of applicability. Indeed, all MINLPs involving *polynomials* can be reduced to quadratic MINLPs by using additional constraints and variables (e.g., the cubic constraint $y_2 = y_1^3$ can be reduced to the quadratic constraints $y_2 = y_1 w$ and $w = y_1^2$, where w is an additional variable). The papers [89,90] provide further discussion of such transformations. Moreover, even functions that are not polynomials can often be well approximated by quadratic functions in the domain of interest.

5.1. Quadratic optimization with binary variables

The simplest quadratic MINLPs are those in which all variables are binary. The literature on such problems is vast, and several different approaches have been suggested for tackling them. Among them, we mention the following.

- A seminal result due to Fortet [91] (see also [92,93]) is that a quadratic function of n binary variables can be *linearized* by adding $\mathcal{O}(n^2)$ additional variables and constraints. More precisely, any term of the form $x_i x_j$, with $i \neq j$, can be replaced with a new binary variable x_{ij}, along with constraints of the form $x_{ij} \leq x_i$, $x_{ij} \leq x_j$, and $x_{ij} \geq x_i + x_j - 1$. Note the match with McCormick's approximation of the function $y_i y_j$ in the continuous case, mentioned in Section 3.1.
- Glover [94] showed that, in fact, one can linearize such functions using only $\mathcal{O}(n)$ additional variables and constraints. See, e.g., [95,96] for related formulations. Chaovalitwongse et al. [97] and Sherali and Smith [98] provide recent, conceptually different $O(n)$ linearization approaches.
- Hammer and Rubin [99] showed that non-convex quadratic functions in binary variables can be *convexified* by adding or subtracting appropriate multiples of terms of the form $x_i^2 - x_i$ (which equal zero when x_i is binary). This approach was improved by Körner [100].
- Hammer et al. [101] present a bounding procedure, called the *roof dual*, which replaces each quadratic function with a tight linear under-estimator. Extensions of this are surveyed in Boros and Hammer [102].

- Pardalos and Rodgers [103] solve unconstrained 0–1 QPs within a branch-and-bound algorithm involving careful pre-processing and computational efficiencies.
- Poljak and Wolkowicz [104] examine several bounding techniques for unconstrained 0–1 QPs, and show that they all give the same bounds.
- Caprara [105] shows how to compute good bounds efficiently using Lagrangian relaxation, when the linearized version of the problem can be solved efficiently.

There are three other well-known approaches, that are not only highly effective, but can be adapted to quadratic problems that have a mixture of binary, integer-constrained, and/or continuous variables. These are discussed in the following three subsections.

5.2. The reformulation-linearization technique (RLT)

In their seminal 1986 paper, Adams and Sherali [106] proposed the following approach to 0–1 quadratic programs. First, the additional x_{ij} variables are introduced, along with the constraints of Fortet [91] mentioned in the previous subsection. Next, new valid constraints are derived as follows.

- Each linear inequality, say $a^T x \le b$, is multiplied by each variable in turn, to obtain n valid quadratic inequalities of the form $(a^T x)x_k \le bx_k$. Replacing each product of the form $x_i x_k$ with the single variable x_{ik}, and replacing x_k^2 with x_k, one obtains the following valid linear inequalities:

$$\sum_{i \neq k} a_i x_{ik} \le (b - a_k)x_k \quad (k = 1, \dots, n).$$

- Similarly, multiplying each linear inequality by terms of the form $1 - x_k$, one obtains n more valid quadratic inequalities of the form $(a^T x)(1 - x_k) \le b(1 - x_k)$. This yields the linear inequalities:

$$\sum_{i \neq k} a_i (x_i - x_{ik}) \le (b - a_k)(1 - x_k) \quad (k = 1, \dots, n).$$

  The original linear inequalities can then be discarded, as they are implied by the new ones.
- Next, each linear equation, say $c^T x = d$, is multiplied by each variable in turn, to obtain n valid quadratic equations. Linearizing as usual, one obtains

$$\sum_{i \neq k} c_i x_{ik} = (d - c_k)x_k \quad (k = 1, \dots, n).$$

  Unlike in the case of inequalities, there is no need to multiply equations by $1 - x_k$, since the resulting equations would be implied by the original equations and the new ones.

Later on, Sherali and Adams [107] realized that, if the above procedure is applied to a 0–1 *linear* program, the continuous relaxation of the transformed instance is stronger than that of the original instance. They also showed that one could obtain a hierarchy of increasingly stronger relaxations, by introducing variables representing products of three variables, products of four variables, and so on. They named the entire scheme the *Reformulation-Linearization Technique* (RLT).

Since then, the RLT has been extended to cover several other classes of convex and non-convex MINLPs, beyond pure 0–1 linear and quadratic problems. We will mention some of these extensions in Sections 5.5 and 5.6, but, for a full treatment, the reader is referred to the book [9].

5.3. Semidefinite relaxation

Another popular approach for generating strong relaxations of non-convex quadratic optimization (and other) problems is based on *semidefinite programming* (SDP). The starting point of this approach is as follows. Given an arbitrary vector $x \in \mathbb{R}^n$ of decision variables, define the matrix $X = xx^T$. Note that a quadratic function of x is a linear function of X. Therefore, any optimization problem involving quadratic functions can be reformulated as an optimization problem involving linear functions, together with the single non-convex constraint $X = xx^T$.

Now, note that X is real, symmetric and *positive semidefinite* (psd), and that, for $1 \le i \le j \le n$, the entry X_{ij} represents the product $x_i x_j$ (and is thus analogous to the term x_{ij} in Sections 5.1 and 5.2). Moreover, as pointed out in [108,109], the augmented matrix

$$\hat{X} := \begin{pmatrix} 1 \\ x \end{pmatrix} \begin{pmatrix} 1 \\ x \end{pmatrix}^T = \begin{pmatrix} 1 & x^T \\ x & X \end{pmatrix}$$

is also psd. This fact enables one to construct useful SDP relaxations of various quadratic optimization problems (e.g., [110–112,109, 113,114,108]).

Note that, for a Mixed-Integer Quadratic Program (i.e., an MINLP with a quadratic objective but linear constraints), one can easily combine the RLT and SDP, to obtain a relaxation that dominates those obtained by using either technique alone. Anstreicher [115] shows that this can yield significant benefits in terms of bound strength, though running times can be high.

Buchheim and Wiegele [116] use SDP relaxations and a tailored branching scheme for a special kind of Mixed-Integer Quadratic Program, in which the only constraints present are ones that enforce each variable to belong to a specified subset of $\mathbb{R}$. Note that this includes unconstrained problems with any mixture of continuous, binary and general-integer variables.

A *completely positive* matrix is one that can be factored as NN^T, where N is a component-wise nonnegative matrix. Clearly, if $x \in \mathbb{R}^n_+$, then $\hat{X}$ is completely positive rather than merely psd. One can use this fact to derive even stronger SDP relaxations; see the survey [117]. Chen and Burer [118] use such an approach within branch-and-bound to solve non-convex QPs having continuous variables and linear constraints.

5.4. Polyhedral theory and convex analysis

We have seen, in the previous three subsections, that a popular way to tackle quadratic MINLPs is to introduce new variables representing products of pairs of original variables. Once this has been done, it is natural to study the convex hull of feasible solutions, in the hope of deriving strong linear (or at least convex) relaxations.

Padberg [40] tackled exactly this topic when he introduced a polytope associated with unconstrained 0–1 quadratic programs, which he called the *Boolean quadric polytope*. The Boolean quadric polytope of order n is defined as

$$\mathrm{BQP}_n = \mathrm{conv}\left\{ x \in \{0,1\}^{n+\binom{n}{2}} : x_{ij} = x_i x_j \ (1 \le i < j \le n) \right\}.$$

Note that here, just as in the original version of the RLT, the variable x_{ij} is not defined when $i = j$. This is because squaring a binary variable has no effect.

Padberg [40] derived various valid and facet-defining inequalities for BQP_n, called *triangle*, *cut*, and *clique* inequalities. Since then, a wide variety of valid and facet-defining inequalities have been discovered. These are surveyed in the book by Deza and Laurent [119].

There are several other papers on polytopes related to quadratic versions of traditional combinatorial optimization problems. Among them, we mention [120] on the quadratic assignment polytope, [121] on the quadratic semi-assignment polytope, and [111] on the quadratic knapsack polytope. Padberg and Rijal [122] studied several quadratic 0–1 problems in a common framework.

There are also three papers on the following (non-polyhedral) convex set [123–125]:

$$\operatorname{conv}\left\{x \in [0,1]^n,\ y \in \mathbb{R}^{\binom{n+1}{2}},\ x_{ij} = x_i x_j\ (1 \le i \le j \le n)\right\}.$$

This convex set is associated with non-convex quadratic programming with box constraints, a classical problem in global optimization. Burer and Letchford [124] use a combination of polyhedral theory and convex analysis to analyze this convex set. In a follow-up paper, Burer and Letchford [126] apply the same approach to the case in which there are unbounded continuous and integer variables.

Complementing the above approaches, several researchers have looked at the convex hull of sets of the form $\{(z, x) \in \mathbb{R}^{n+1} : z = q(x), x \in D\}$, where $q(x)$ is a given quadratic function and D is a bounded (most often simple) domain [127–129]. While slightly less general than convexifying in the space of all pairs x_{ij} as done above, this approach much more directly linearizes and convexifies the quadratics of interest in a given problem. It can also be effectively generalized to the non-quadratic case (see, for example, Section 2 of [53]).

5.5. Some additional techniques

Saxena et al. [130,131] have derived strong cutting planes for non-convex MIQCQPs (mixed-integer quadratically constrained quadratic programs). In [130], the cutting planes are derived in the extended quadratic space of the X_{ij} variables, using disjunctions of the form $(a^T x \le b) \vee (a^T x \ge b)$. In [131], the cutting planes are derived in the original space by projecting down certain relaxations from the quadratic space. See also the recent survey Burer and Saxena [132]. Separately, Galli et al. [133] have adapted the 'gap inequalities', originally defined in [134] for the max-cut problem, to non-convex MIQPs.

Berthold et al. [135] present an exact algorithm for MIQCQPs that is based on the integration of constraint programming and branch-and-cut. The key is to use quadratic constraints to reduce domains, wherever possible. Misener and Floudas [25] present an exact algorithm for non-convex mixed 0–1 QCQPs that is based on branch-and-reduce, together with cutting planes derived from the consideration of polyhedra involving small subsets of variables.

Billionnet et al. [136] revisit the approach for 0–1 quadratic programs, mentioned in Section 5.1, due to Hammer and Rubin [99] and Körner [100]. They show that an optimal reformulation can be derived from the dual of an SDP relaxation. Billionnet et al. [137] then show that the method can be extended to general MIQPs, provided that the integer-constrained variables are bounded and the part of the objective function associated with the continuous variables is convex.

Adams and Sherali [37] and Freire et al. [38] present algorithms for *bilinear* problems. A *bilinear* optimization problem is one in which all constraints are linear, and the objective function is the product of two linear functions (and therefore quadratic). The paper [37] is concerned with the case in which one of the linear functions involves binary variables and the other involves continuous variables. The paper [38], on the other hand, is concerned with the case in which all variables are integer-constrained.

Finally, we mention that Nowak [138] proposes using Lagrangian decomposition for non-convex MIQCQPs.

5.6. Extensions to polynomial optimization

Many researchers have extended ideas from quadratic programs to the much broader class of polynomial optimization problems. A simple way to linearize polynomials involving binary variables was given by Glover and Woolsey [92]. The RLT approach of Sherali and Adams [9] explained in Section 5.2 creates a hierarchy of ever-tighter LP relaxations of polynomial problems. Some successful applications of the RLT approach include the solution of 0–1 polynomial programs [107], mixed-integer polynomial programs [139], and mixed-discrete problems having non-convex polynomial constraints and general convex constraints [140].

Recently, some sophisticated approaches have been developed for mixed 0–1 polynomial programs that draw on concepts from real algebraic geometry, commutative algebra, and moment theory. Relevant works include Nesterov [141], Parrilo [142], Lasserre [47], Laurent [143], and De Loera et al. [144]. The method of Lasserre [145] works for integer polynomial programs when each variable has an explicit lower and upper bound.

Michaels and Weismantel [146] make an important observation for Integer Polynomial Programming. They note that, given a non-convex polynomial, say $f(x)$, there may exist a convex polynomial, say $f'(x)$, that achieves the same value as $f(x)$ at all integer points. In principle, this could allow such non-convex programs to be made convex.

6. Software

There are five software packages that can solve non-convex MINLPs to proven optimality:

`BARON`, α-BB, `LINDO-Global`, `Couenne`, and `GloMIQO`.

`BARON` is due to Sahinidis and colleagues [57,58,10], α-BB is due to Adjiman et al. [64], and `LINDO-Global` is described in Lin and Schrage [147]. `Couenne` is due to Belotti et al. [43], and `GloMIQO` [148] relates to the technique of Misener and Floudas [25] described in Section 5.5.

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

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

The first three are actually packages for convex MINLPs, while the fourth is based on ant-colony optimization. The algorithmic approach behind `BONMIN` is described in [8], and `DICOPT` has been developed by Grossmann and co-authors (e.g., Kocis and Grossmann [149]). `LaGO` is described in Nowak and Vigerske [80], and `MIDACO` is presented in [84] and available at `midaco-solver.com`.

The package due to Liberti et al. [85], described in Section 4.6, is called *RECIPE*. The paper by Berthold et al. [135] presents an MIQCP solver for the software package `SCIP`. Finally, `GloptiPoly` [150] can solve general polynomial optimization problems.

7. Conclusions

Because non-convex MINLPs encompass a huge range of applications and problem types, the depth and breadth of techniques used to solve them should come as no surprise. In this survey, we have tried to give a fair and up-to-date introduction to these techniques.

Without a doubt, substantial successes in the fields of MILP and global optimization have played critical roles in the development of algorithms for non-convex MINLPs, and we suspect further successes will have continued benefits for MINLPs. We believe, also, that even more insights can be achieved by studying MINLPs specifically. For example, analyzing — and generating cutting planes for — the various convex hulls that arise in MINLPs (see Section 5.4) will require aspects of both polyhedral theory and convex analysis to achieve best results.

We also advocate the development of algorithms for various special cases of non-convex MINLPs. While general-purpose algorithms for MINLPs are certainly needed, since MINLP are so broad, there will always be a need for handling important special cases. Special cases can also allow the development of newer techniques (e.g., semidefinite relaxations), which may then progress to more general techniques.

Finally, we believe there will be an increasing place for heuristics and approximation algorithms for non-convex MINLPs. Most techniques so far aim for globally optimal solutions, but in practice it would be valuable to have sophisticated approaches for finding near-optimal solutions.

Acknowledgments

The first author was supported by the National Science Foundation under grant CCF-0545514, and the second author was supported by the Engineering and Physical Sciences Research Council under grant EP/D072662/1.

The authors are in debt to two anonymous referees whose comments and suggestions greatly improved the paper.

References

[1] A.M. Geoffrion, Generalized benders decomposition, J. Optim. Theory Appl. 10 (1972) 237–260.
[2] O.K. Gupta, V. Ravindran, Branch and bound experiments in convex nonlinear integer programming, Management. Sci. 31 (1985) 1533–1546.
[3] M.A. Duran, I.E. Grossmann, An outer-approximation algorithm for a class of mixed-integer nonlinear programs, Math. Program. 36 (1986) 307–339.
[4] I. Quesada, I.E. Grossmann, An LP/NLP based branch and bound algorithm for convex MINLP optimization problems, Comput. Chem. Eng. 16 (1992) 937–947.
[5] T. Westerlund, F. Pettersson, A cutting plane method for solving convex MINLP problems, Comput. Chem. Eng. 19 (1995) S131–S136.
[6] R.A. Stubbs, S. Mehrotra, A branch-and-cut method for 0–1 mixed convex programming, Math. Program. 86 (1999) 515–532.
[7] K. Abhishek, S. Leyffer, J.T. Linderoth, FilMINT: an outer-approximation-based solver for nonlinear mixed integer programs, INFORMS J. Comput. 22 (2010) 555–567.
[8] P. Bonami, L.T. Biegler, A.R. Conn, G. Cornuéjols, I.E. Grossmann, C.D. Laird, J. Lee, A. Lodi, F. Margot, N. Sawaya, An algorithmic framework for convex mixed integer nonlinear programs, Discrete Optim. 5 (2008) 186–204.
[9] H.D. Sherali, W.P. Adams, A Reformulation-Linearization Technique for Solving Discrete and Continuous Nonconvex Problems, Kluwer, Dordrecht, 1999.
[10] M. Tawarmalani, N.V. Sahinidis, Convexification and Global Optimization in Continuous and Mixed-Integer Nonlinear Programming, Springer, 2003.
[11] R. Jeroslow, There cannot be any algorithm for integer programming with quadratic constraints, Oper. Res. 21 (1) (1973) 221–224.
[12] R. Hemmecke, M. Köppe, J. Lee, R. Weismantel, Nonlinear integer programming, in: M. Jünger, et al. (Eds.), 50 Years of Integer Programming 1958–2008, Springer, Berlin, 2010, pp. 561–618.
[13] M. Köppe, On the complexity of nonlinear mixed-integer optimization, in: J. Lee, S. Leyffer (Eds.), Mixed-Integer Nonlinear Programming, in: IMA Volumes in Mathematics and its Applications, vol. 154, Springer, Berlin, 2011, pp. 533–558.
[14] P. Bonami, M. Kilinç, J. Linderoth, Algorithms and software for convex mixed integer nonlinear programs, in: J. Lee, S. Leyffer (Eds.), Mixed Integer Nonlinear Programming, in: IMA Volumes in Mathematics and its Applications, vol. 154, Springer, Berlin, 2011, pp. 1–40.
[15] C. D'Ambrosio, A. Lodi, Mixed integer nonlinear programming tools: a practical overview, 4OR 9 (2011) 329–349.
[16] C.A. Floudas, C.E. Gounaris, A review of recent advances in global optimization, J. Global Optim. 45 (2009) 3–38.
[17] I.E. Grossmann, Review of nonlinear mixed-integer and disjunctive techniques, Optim. Eng. 3 (2002) 227–252.
[18] S. Leyffer, J.T. Linderoth, J. Luedtke, A. Miller, T. Munson, Applications and algorithms for mixed integer nonlinear programming, J. Phys.: Conf. Ser. 180 (2009) 012–014.
[19] R. Pörn, T. Westerlund, A cutting plane method for minimizing pseudo-convex functions in the mixed integer case, Comput. Chem. Eng. 24 (2000) 2655–2665.
[20] I.E. Grossmann, R.W.H. Sargent, Optimum design of multipurpose chemical plants, Ind. Eng. Chem. Process Des. Dev. 18 (1979) 343–348.
[21] R. Misener, C.A. Floudas, Advances for the pooling problem: modeling, global optimization, and computational studies (survey), Appl. Comput. Math. 8 (2009) 3–22.
[22] M. Luyben, C.A. Floudas, A multiobjective optimization approach for analyzing the interaction of design and control, part 1: theoretical framework and binary distillation synthesis, Comput. Chem. Eng. 18 (1994) 933–969.
[23] T. Yee, I. Grossmann, Simultaneous optimization models for heat integration II: heat exchanger network synthesis, Comput. Chem. Eng. 14 (1990) 1165–1184.
[24] C.A. Floudas, Mixed integer nonlinear programming, in: C.A. Floudas, P.M. Pardalos (Eds.), Encyclopedia of Optimization, Springer, 2009, pp. 2234–2247.
[25] R. Misener, C.A. Floudas, Global optimization of mixed-integer quadratically-constrained quadratic programs (MIQCQP) through piecewise-linear and edge-concave relaxations, Math. Program. Series B (2012) http://dx.doi.org/10.1007/s10107-012-0555-6.
[26] C. Bragalli, C. D'Ambrosio, J. Lee, A. Lodi, P. Toth, An MINLP solution method for a water network problem, in: Proc. of ESA 2006, in: Lecture Notes in Computer Science, vol. 4168, Springer, Berlin, 2006, pp. 696–707.
[27] A. Martin, M. Möller, S. Moritz, Mixed integer models for the stationary case of gas network optimization, Math. Program. 105 (2006) 563–582.
[28] W. Murray, U. Shanbhag, A local relaxation approach for the siting of electrical substations, Comput. Optim. Appl. 33 (2006) 7–49.
[29] A. Fügenschuh, H. Homfeld, H. Schülldorf, S. Vigerske, Mixed-integer nonlinear problems in transportation applications, in: H. Rodrigues et al. (eds.) Proceedings of the 2nd International Conference on Engineering Optimization.
[30] I. Harjunkoski, T. Westerlund, R. Pörn, Numerical and environmental considerations on a complex industrial mixed integer non-linear programming (MINLP) problem, Comput. Chem. Eng. 23 (1999) 1545–1561.
[31] M. van den Briel, J. Villalobos, G. Hogg, T. Lindemann, A. Mule, America west airlines develops efficient boarding strategies, Interfaces 35 (2005) 191–201.
[32] F. You, S. Leyffer, Mixed-integer dynamic optimization for oil-spill response planning with integration of a dynamic oil weathering model, AIChE J. 57 (2011) 3555–3564.
[33] G. Corsano, A.R. Vecchietti, J.M. Montagna, Optimal design for sustainable bioethanol supply chain considering detailed plant performance model, Comput. Chem. Eng. 35 (8) (2011) 1384–1398.
[34] A. Guerra, A.M. Newman, S. Leyffer, Concrete structure design using mixed-integer nonlinear programming with complementarity constraints, SIAM J. Optim. 21 (2011) 833–863.
[35] K. Abhishek, S. Leyffer, J.T. Linderoth, Modeling without categorical variables: a mixed-integer nonlinear program for the optimization of thermal insulation systems, Optim. Eng. 11 (2010) 185–212.
[36] P.M. Pardalos, W. Chaovalitwongse, L.D. Iasemidis, J. Chris Sackellares, D.-S. Shiau, P.R. Carney, O.A. Prokopyev, V.A. Yatsenko, Seizure warning algorithm based on optimization and nonlinear dynamics, Math. Program. 101 (2004) 365–385.
[37] W.P. Adams, H.D. Sherali, Mixed-integer bilinear programming problems, Math. Program. 59 (1993) 279–305.
[38] A.S. Freire, E. Moreno, J.P. Vielma, An integer linear programming approach for bilinear integer programming, Oper. Res. Lett. 40 (2) (2012) 74–77.
[39] M.X. Goemans, D. Williamson, Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming, J. ACM 42 (1995) 1115–1145.
[40] M.W. Padberg, The boolean quadric polytope: some characteristics, facets and relatives, Math. Program. 45 (1989) 139–172.
[41] M. Anjos, A. Vannelli, Computing globally optimal solutions for single-row layout problems using semidefinite programming and cutting planes, INFORMS J. Comput. 20 (2008) 611–617.
[42] H. Sherali, J. Desai, A global optimization RLT-based approach for solving the hard clustering problem, J. Global Optim. 32 (2005) 281–306.
[43] S. Burer, On the copositive representation of binary and continuous nonconvex quadratic programs, Math. Program. 120 (2009) 479–495.
[44] J. Linderoth, A simplicial branch-and-bound algorithm for solving quadratically constrained quadratic programs, Math. Program. 103 (2005) 251–282.
[45] A. Fischer, C. Helmberg, The symmetric quadratic traveling salesman problem, Manuscript, Fakultät für Mathematik, Technishe Universität Chemnitz, Germany, 2011.
[46] I. Gentilini, F. Margot, K. Shimada, The traveling salesman problem with neighborhoods: MINLP solution. Manuscript, Tepper School of Business, Carnegie Mellon University. Optim. Methods Softw., 2011 (in press).
[47] J.B. Lasserre, Global optimization with polynomials and the problem of moments, SIAM J. Optim. 11 (2001) 796–817.
[48] G.P. McCormick, Computability of global solutions to factorable nonconvex programs. part I. convex underestimating problems, Math. Program. 10 (1976) 146–175.
[49] R. Horst, H. Tuy, Global Optimization: Deterministic Approaches, second ed., Springer-Verlag, Berlin, 1993.
[50] I.P. Androulakis, C.D. Maranas, C.A. Floudas, α-BB: a global optimization method for general constrained nonconvex problems, J. Global Optim. 7 (1995) 337–363.
[51] E.M.L. Beale, Mathematical Programming in Practice, Pitmans, London, 1968.
[52] J. Tomlin, A suggested extension of special ordered sets to non-separable non-convex programming problems, Ann. Discrete Math. 11 (1981) 359–370.
[53] P. Belotti, Disjunctive cuts for non-convex MINLP, in: Mixed Integer Nonlinear Programming, in: The IMA Volumes in Mathematics and its Applications, vol. 154, Springer, 2012, pp. 117–144.
[54] A.H. Land, A.G. Doig, An automatic method of solving discrete programming problems, Econometrica 28 (1960) 497–520.

[55] E.M.B. Smith, C.C. Pantelides, Global optimisation of non-convex MINLPs, Comput. Chem. Eng. 21 (1997) S791.
[56] S. Lee, I.E. Grossmann, A global optimization algorithm for non-convex generalized disjunctive programming and applications to process systems, Comput. Chem. Eng. 25 (2001) 1675–1697.
[57] H.S. Ryoo, N.V. Sahinidis, Global optimization of nonconvex NLPs and MINLPs with applications in process design, Comput. Chem. Eng. 19 (1995) 551–566.
[58] H.S. Ryoo, N.V. Sahinidis, A branch-and-reduce approach to global optimization, J. Global Optim. 8 (1996) 107–138.
[59] M. Tawarmalani, N.V. Sahinidis, Global optimization of mixed-integer nonlinear programs: a theoretical and computational study, Math. Program. 99 (2004) 563–591.
[60] M. Tawarmalani, N.V. Sahinidis, A polyhedral branch-and-cut approach to global optimization, Math. Program. 103 (2005) 225–249.
[61] P. Belotti, J. Lee, L. Liberti, F. Margot, A. Wächter, Branching and bounds tightening techniques for non-convex MINLP, Optim. Methods Softw. 24 (2009) 597–634.
[62] C.S. Adjiman, I.P. Androulakis, C.A. Floudas, A global optimization method, α-BB, for general twice-differentiable constrained NLPs: II. implementation and computational results, Comput. Chem. Eng. 22 (1998) 1159–1179.
[63] C.S. Adjiman, S. Dallwig, C.A. Floudas, A. Neumaier, A global optimization method, α-BB, for general twice-differentiable NLPs: I. theoretical advances, Comput. Chem. Eng. 22 (1998) 1137–1158.
[64] C.S. Adjiman, I.P. Androulakis, C.A. Floudas, Global optimization of mixed-integer nonlinear problems, AIChE J. 46 (2000) 1769–1797.
[65] E.M.L. Beale, J.A. Tomlin, Special facilities in a general mathematical programming system for non-convex problems using ordered sets of variables, in: J. Lawrence (Ed.), Proc. of the 5th Nat. Conf. in Oper. Res., Tavistock, London, 1970, pp. 447–454.
[66] E.M.L. Beale, J.J.H. Forrest, Global optimization using special ordered sets, Math. Program. 10 (1) (1976) 52–69.
[67] A.B. Keha, I.R. de Farias Jr., G.L. Nemhauser, Models for representing piecewise linear cost functions, Oper. Res. Lett. 32 (2004) 44–48.
[68] A.B. Keha, I.R. de Farias Jr., G.L. Nemhauser, A branch-and-cut algorithm without binary variables for nonconvex piecewise linear optimization, Oper. Res. 54 (2006) 847–858.
[69] J.P. Vielma, G.L. Nemhauser, Modeling disjunctive constraints with a logarithmic number of binary variables and constraints, Math. Program. 128 (2011) 49–72.
[70] E.M.L. Beale, Branch and bound methods for mathematical programming systems, Ann. Discrete Math. 5 (1979) 201–219. Discrete optimization (Proc. Adv. Res. Inst. Discrete Optimization and Systems Appl., Banff, Alta., 1977), II.
[71] B. Geißler, A. Martin, A. Morsi, L. Schewe, Using piecewise linear functions for solving MINLPs, in: J. Lee, S. Leyffer (Eds.), Mixed Integer Nonlinear Programming, in: IMA Volumes in Mathematics and its Applications, vol. 154, Springer, Berlin, 2012, pp. 287–314.
[72] S. Leyffer, A. Sartenaer, E. Wanufelle, Branch-and-refine for mixed-integer nonconvex global optimization. ANL/MCS-P1547-0908, Argonne National Laboratory, 2008, Preprint.
[73] H.P. Benson, S.S. Erenguc, An algorithm for concave integer minimization over a polyhedron, Naval Res. Logist. 37 (1990) 515–525.
[74] K.M. Bretthauer, A.V. Cabot, M.A. Venkataramanan, An algorithm and new penalties for concave integer minimization over a polyhedron, Naval Res. Logist. 41 (1994) 435–454.
[75] H. Tuy, Concave programming under linear constraints, Soviet Math. 5 (1964) 1437–1440.
[76] P. Kesavan, R.J. Allgor, E.P. Gatzke, P.I. Barton, Outer approximation algorithms for separable non-convex mixed-integer nonlinear programs, Math. Program. 100 (2004) 517–535.
[77] R. Karuppiah, I.E. Grossman, A Lagrangian based branch-and-cut algorithm for global optimization of nonconvex mixed-integer nonlinear programs with decomposable structures, J. Global Optim. 41 (2008) 163–186.
[78] C. D'Ambrosio, J. Lee, A. Wächter, An algorithmic framework for MINLP with separable non-convexity, in: J. Lee, S. Leyffer (Eds.), Mixed-Integer Nonlinear Programming, in: IMA Volumes in Mathematics and its Applications, vol. 154, Springer, Berlin, 2011, pp. 315–348.
[79] S. Leyffer, Integrating SQP and branch-and-bound for mixed integer nonlinear programming, Comput. Optim. Appl. 18 (2001) 259–309.
[80] I. Nowak, Vigerske, LaGO: a (heuristic) branch and cut algorithm for nonconvex MINLPs, Cent. Eur. J. Oper. Res. 16 (2008) 127–138.
[81] O. Exler, L.T. Antelo, J.A. Egea, A.A. Alonso, J.R. Banga, A tabu search-based algorithm for mixed-integer nonlinear problems and its application to integrated process and control system design, Comput. Chem. Eng. 32 (2008) 1877–1891.
[82] Y. Luo, X. Yuan, Y. Liu, An improved PSO algorithm for solving non-convex NLP/MINLP problems with equality constraints, Comput. Chem. Eng. 31 (2007) 153–162.
[83] C.-T. Young, Y. Zheng, C.-W. Yeh, S.-S. Jang, Information-guided genetic algorithm approach to the solution of MINLP problems, Ind. Eng. Chem. Res. 46 (2007) 1527–1537.
[84] M. Schlüter, J.A. Egea, J.R. Banga, Extended ant colony optimization for non-convex mixed integer nonlinear programming, Comput. Oper. Res. 36 (7) (2009) 2217–2229.
[85] L. Liberti, G. Nannicini, N. Mladenovic, A recipe for finding good solutions to MINLPs, Math. Program. Comput. 3 (2011) 349–390.
[86] T. Berthold, RENS: the optimal rounding. Zib-report 12-17, Konrad-Zuse-Zentrum für Informationstechnik Berlin, Berlin, Germany, April 2012.
[87] C. D'Ambrosio, A. Frangioni, L. Liberti, A. Lodi, Experiments with a feasibility pump approach for nonconvex MINLPs, in: P. Festa (Ed.), Proceedings of the 9th Symposium on Experimental Algorithms, in: Lecture Notes in Computer Science, vol. 6049, Springer, Berlin, 2010.
[88] G. Nannicini, P. Belotti, Rounding-based heuristics for nonconvex MINLPs, Math. Program. Comput. 4 (1) (2012) 1–31.
[89] N.Z. Shor, Dual quadratic estimates in polynomial and Boolean programming, Ann. Oper. Res. 25 (1–4) (1990) 163–168. Computational methods in global optimization.
[90] H.D. Sherali, C.H. Tuncbilek, Comparison of two reformulation-linearization technique based linear programming relaxations for polynomial programming problems, J. Global Optim. 10 (4) (1997) 381–390.
[91] R. Fortet, L'Algèbre de Boole et ses applications en recherche opérationnelle, Cahiers Centre Etudes Rech. Oper. no. 4 (1959) 5–36.
[92] F. Glover, E. Woolsey, Converting the 0–1 polynomial programming problem to a 0–1 linear program, Oper. Res. 22 (1974) 180–182.
[93] P. Hansen, Methods of nonlinear 0–1 programming, Ann. Discr. Math. 5 (1979) 53–70.
[94] F. Glover, Improved linear integer programming formulations of nonlinear integer problems, Management. Sci. 22 (1975) 455–460.
[95] W.P. Adams, R.J. Forrester, A simple recipe for concise mixed 0–1 linearizations, Oper. Res. Lett. 33 (2005) 55–61.
[96] W.P. Adams, R.J. Forrester, F.W. Glover, Comparisons and enhancement strategies for linearizing mixed 0–1 quadratic programs, Discrete Optim. 1 (2004) 99–120.
[97] W. Chaovalitwongse, P.M. Pardalos, O.A. Prokopyev, A new linearization technique for multi-quadratic 0–1 programming problems, Oper. Res. Lett. 32 (6) (2004) 517–522.
[98] H.D. Sherali, J.C. Smith, An improved linearization strategy for zero–one quadratic programming problems, Optim. Lett. 1 (1) (2007) 35–47.
[99] P.L. Hammer, A.A. Rubin, Some remarks on quadratic programming with 0–1 variables, RAIRO 3 (1970) 67–79.
[100] F. Körner, A tight bound for the boolean quadratic optimization problem and its use in a branch and bound algorithm, Optimization 19 (1988) 711–721.
[101] P.L. Hammer, P. Hansen, B. Simeone, Roof duality, complementation and persistency in quadratic 0–1 optimization, Math. Program. 28 (1984) 121–155.
[102] E. Boros, P.L. Hammer, Pseudo-Boolean optimization, Discrete Appl. Math. 123 (2002) 155–225.
[103] P.M. Pardalos, G.P. Rodgers, Computational aspects of a branch and bound algorithm for quadratic zero–one programming, Computing 45 (2) (1990) 131–144.
[104] S. Poljak, H. Wolkowicz, Convex relaxations of (0, 1)-quadratic programming, Math. Oper. Res. 20 (1995) 550–561.
[105] A. Caprara, Constrained 0–1 quadratic programming: Basic approaches and extensions, European J. Oper. Res. 187 (2008) 1494–1503.
[106] W.P. Adams, H.D. Sherali, A tight linearization and an algorithm for zero–one quadratic programming problems, Management Sci. 32 (10) (1986) 1274–1290.
[107] H.D. Sherali, W.P. Adams, A hierarchy of relaxations between the continuous and convex hull representations for zero–one programming problems, SIAM J. Discrete Math. 3 (3) (1990) 411–430.
[108] N.Z. Shor, Quadratic optimization problems, Sov. J. Comput. Syst. Sci. 25 (1987) 1–11.
[109] L. Lovász, A.J. Schrijver, Cones of matrices and set-functions and 0–1 optimization, SIAM J. Optim. 1 (1991) 166–190.
[110] T. Fujie, M. Kojima, Semidefinite programming relaxation for nonconvex quadratic programs, J. Global Optim. 10 (1997) 367–380.
[111] C. Helmberg, F. Rendl, R. Weismantel, A semidefinite programming approach to the quadratic knapsack problem, J. Comb. Optim. 4 (2002) 197–215.
[112] C. Lémaréchal, F. Oustry, SDP relaxations in combinatorial optimization from a Lagrangian viewpoint, in: N. Hadjisawas, P.M. Pardalos (Eds.), Advances in Convex Analysis and Global Optimization, Kluwer, Dordrecht, 2001.
[113] S. Poljak, F. Rendl, H. Wolkowicz, A recipe for semidefinite relaxation for (0, 1)-quadratic programming, J. Global Optim. 7 (1995) 51–73.
[114] M. Ramana, An algorithmic analysis of multiquadratic and semidefinite programming problems, Ph.D. Thesis, Johns Hopkins University, Baltimore, MD, 1993.
[115] K.M. Anstreicher, Semidefinite programming versus the reformulation-linearization technique for nonconvex quadratically constrained quadratic programming, J. Global Optim. 43 (2009) 471–484.
[116] C. Buchheim, A. Wiegele, Semidefinite relaxations for non-convex quadratic mixed-integer programming. Math. Program. 1–18. http://dx.doi.org/10.1007/s10107-012-0534-y.
[117] M. Dür, Copositive programming: a survey, in: M. Diehl, F. Glineur, E. Jarlebring, W. Michiels (Eds.), Recent Advances in Optimization and its Applications in Engineering, Springer, Berlin, 2010.
[118] J. Chen, S. Burer, Globally solving nonconvex quadratic programming problems via completely positive programming, Math. Program. Comput. 4 (1) (2012) 33–52.
[119] M.M. Deza, M. Laurent, Geometry of Cuts and Metrics, Springer, Berlin, 1997.
[120] M. Jünger, V. Kaibel, Box-inequalities for quadratic assignment polytopes, Math. Program. 91 (2001) 175–197.

[121] H. Saito, T. Fujie, T. Matsui, S. Matuura, A study of the quadratic semi-assignment polytope, Discrete Optim. 6 (2009) 37–50.
[122] M.W. Padberg, M.P. Rijal, Location, Scheduling, Design and Integer Programming, Springer, Berlin, 1996.
[123] K.M. Anstreicher, S. Burer, Computable representations for convex hulls of low-dimensional quadratic forms, Math. Program. 124 (2010) 33–43.
[124] S. Burer, A.N. Letchford, On non-convex quadratic programming with box constraints, SIAM J. Optim. 20 (2009) 1073–1089.
[125] Y. Yajima, T. Fujie, A polyhedral approach for nonconvex quadratic programming problems with box constraints, J. Global Optim. 13 (1998) 151–170.
[126] S. Burer, A.N. Letchford, Unbounded convex sets for non-convex mixed-integer quadratic programming. Working paper, Lancaster University, Lancaster, UK, 2011.
[127] C.A. Meyer, C.A. Floudas, Convex envelopes for edge-concave functions, Math. Program. 103 (2005) 207–224.
[128] O. Günlük, J. Lee, J. Leung, A polytope for a product of a real linear functions in 0/1 variables, in: J. Lee, S. Leyffer (Eds.), in: IMA Volumes in Mathematics and its Applications, Springer, 2011, pp. 513–532.
[129] J. Luedtke, M. Namazifar, J. Linderoth, Some results on the strength of relaxations of multilinear functions. Working paper, University of Wisconsin-Madison, 2010.
[130] A. Saxena, P. Bonami, J. Lee, Convex relaxations of non-convex mixed integer quadratically constrained programs: extended formulations, Math. Program. 124 (2010) 383–411.
[131] A. Saxena, P. Bonami, J. Lee, Convex relaxations of non-convex mixed integer quadratically constrained programs: projected formulations, Math. Program. 130 (2011) 359–413.
[132] S. Burer, A. Saxena, The MILP road to MIQCP, in: J. Lee, S. Leyffer (Eds.), Mixed-Integer Nonlinear Programming, in: IMA Volumes in Mathematics and its Applications, vol. 154, Springer, Berlin, 2011, pp. 373–406.
[133] L. Galli, K. Kaparis, A.N. Letchford, Gap inequalities for non-convex mixed-integer quadratic programs, Oper. Res. Lett. 39 (2011) 297–300.
[134] M. Laurent, S. Poljak, Gap inequalities for the cut polytope, SIAM J. Matrix Anal. 17 (1996) 530–547.
[135] T. Berthold, S. Heinz, S. Vigerske, Extending a CIP framework to solve MIQCPs, in: J. Lee, S. Leyffer (Eds.), Mixed-Integer Nonlinear Programming, in: IMA Volumes in Mathematics and its Applications, vol. 154, Springer, Berlin, 2011, pp. 427–445.
[136] A. Billionnet, S. Elloumi, M.-C. Plateau, Improving the performance of standard solvers for quadratic 0–1 programs by a tight convex reformulation: the QCR method, Discrete Appl. Math. 157 (2009) 1185–1197.
[137] A. Billionnet, S. Elloumi, A. Lambert, Extending the QCR method to general mixed-integer programs, Math. Program. 131 (2012) 381–401.
[138] I. Nowak, Lagrangian decomposition of block-separable mixed-integer all-quadratic programs, Math. Program. 102 (2005) 295–312.
[139] H.D. Sherali, W.P. Adams, A hierarchy of relaxations and convex hull characterizations for mixed-integer zero–one programming problems, Discrete Appl. Math. 52 (1) (1994) 83–106.
[140] H.D. Sherali, W.P. Adams, A reformulation-linearization technique (RLT) for semi-infinite and convex programs under mixed 0–1 and general discrete restrictions, Discrete Appl. Math. 157 (6) (2009) 1319–1333.
[141] Y. Nesterov, Global quadratic optimization via conic relaxation, in: R. Saigal, L. Vandenberghe, H. Wolkowicz (Eds.), Handbook of Semidefinite Programming, Kluwer Academic Publishers, 2000.
[142] P. Parrilo, Structured semidefinite programs and semi-algebraic geometry methods in robustness and optimization, Ph.D. Thesis, California Institute of Technology, 2000.
[143] M. Laurent, A comparison of the Sherali-Adams, Lovász-Schrijver, and Lasserre relaxations for 0–1 programming, Math. Oper. Res. 28 (2003) 470–496.
[144] J.A. De Loera, J. Lee, P.N. Malkin, S. Margulies, Hilbert's Nullstellensatz and an algorithm for proving combinatorial infeasibility, in: Proceedings of ISSAC 2008, ACM Press, New York, 2008, pp. 197–206.
[145] J.B. Lasserre, Polynomials nonnegative on a grid and discrete optimization, Trans. Amer. Math. Soc. 354 (2002) 631–649.
[146] D. Michaels, R. Weismantel, Polyhedra related to integer-convex polynomial systems, Math. Program. 105 (2006) 215–232.
[147] Y. Lin, L. Schrage, The global solver in the LINDO API, Optim. Methods Softw. 24 (2009) 657–668.
[148] R. Misener, C.A. Floudas, GloMIQO: global mixed-integer quadratic optimizer. Manuscript, Princeton University, Princeton, NJ, 2012. J. Global Optim. (in press). http://dx.doi.org/10.1007/s10898-012-9874-7.
[149] G.R. Kocis, I.E. Grossmann, Computational experience with DICOPT solving MINLP problems in process synthesis, Comput. Chem. Eng. 13 (1989) 307–315.
[150] D. Henrion, J.-B. Lasserre, J. Löfberg, GloptiPoly 3: moments, optimization and semidefinite programming, Optim. Methods Softw. 24 (2009) 761–779.