Disjunctive Cuts for Mixed Integer Nonlinear Programming Problems

Pierre Bonami[*], Jeff Linderoth[†], Andrea Lodi[‡]

December 29, 2012

Abstract

We survey recent progress in applying disjunctive programming theory for the effective solution of mixed integer nonlinear programming problems. Generation of effective cutting planes is discussed for both the convex and nonconvex cases.

1 Introduction

We consider mixed integer nonlinear programming problems (MINLPs) of the form

$$\begin{aligned} z_{\mathrm{MINLP}} = \min\ & c^\top x \\ & g_i(x) \le 0 \quad i = 1, \ldots, m, \\ & x_j \in \mathbb{Z} \qquad j = 1, \ldots, p, \\ & x_j \in \mathbb{R}, \qquad j = p+1, \ldots, n, \end{aligned} \tag{1}$$

where $1 \le p \le n$, $c \in \mathbb{R}^n$ and for $i = 1, \ldots, m$, $g_i : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ is continuously differentiable. We denote by X the set of feasible solutions to (1).

In this review, we consider the solution of (1) by implicit enumeration approaches. In these methods, X is relaxed to a convex set C in order to obtain a lower bound on the value of z_{MINLP}. The relaxed set C is then refined recursively in the algorithm.

[*]Laboratoire d'Informatique Fondamentale de Marseille, CNRS, Aix-Marseille Univ., Parc Scientifique et Technologique de Luminy, 163 avenue de Luminy - Case 901, F-13288 Marseille Cedex 9, France, `pierre.bonami@lif.univ-mrs.fr`. Supported by ANR grant BLAN06-1-138894.

[†]Department of Industrial and Systems Engineering, University of Wisconsin-Madison, 1513 University Ave., Madison, WI, 53706, `linderoth@wisc.edu`. Supported by the US Department of Energy under grants DE-FG02-08ER25861 and the National Science Foundation under grant CCF-0830153.

[‡]DEIS, University of Bologna, viale Risorgimento 2, 40136 Bologna, Italy. `andrea.lodi@unibo.it`. Supported by MIUR under the PRIN2009 grant.

We will distinguish two different cases. We call (1) a *convex MINLP* if the feasible region of the continuous NLP relaxation obtained by dropping the integrality requirements on the first p variables is a convex set. In that case C is typically the set of continuous feasible solutions to (1) (i.e., $C \stackrel{\text{def}}{=} \{x \in \mathbb{R}^n : g_i(x) \le 0, i = 1, \ldots, m\}$).

In the general case, when the continuous relaxation is not convex, one has to obtain a convex relaxation in an alternative way. In this paper, we will assume that such a relaxation is given and we refer the reader to Tawarmalani and Sahinidis [48] for methods for constructing convex relaxations.

The focus of this paper is on generating *cutting planes* for MINLPs, i.e., finding inequalities that *(i)* are valid for the mixed integer solutions X of (1) and *(ii)* refine (restrict) the convex relaxation C of the problem. This is not to be confused with the so-called *outer approximation* (OA) constraints that are used to define a linear programming (LP) relaxation of the convex relaxation. Indeed, a common approach for solving *convex* MINLPs is to construct a mixed integer linear programming problem that is equivalent to (1). This approach called outer approximation was pioneered by Duran and Grossmann [25] and gives rise to several variants of algorithms using branch-and-bound to solve (1). We call polyhedral outer approximation a set $\mathcal{C} \stackrel{\text{def}}{=} \{x \in \mathbb{R}^n : Ax \ge b\}$ such that $C \subseteq \mathcal{C}$. Duran and Grossmann gave an explicit algorithm to build an outer approximation $\mathcal{C}$ such that minimizing $c^\top x$ over $\mathcal{X} \stackrel{\text{def}}{=} \mathcal{C} \cap (\mathbb{Z}^p \times \mathbb{R}^{n-p})$ and over X gives the same value. Roughly speaking, the basic version of Outer Approximation iteratively constructs $\mathcal{C}$: the algorithm

1. starts with a (generally) weaker LP relaxation $\mathcal{C}' \supseteq \mathcal{C} \supseteq C$;

2. solves the associated mixed integer linear programming problem (MILP) $\{\min c^\top x : \mathcal{C}' \cap (\mathbb{Z}^p \times \mathbb{R}^{n-p})\}$;

3. tests if the mixed integer solution of step 2. is MINLP feasible. If this is not the case, it amends $\mathcal{C}'$ with on OA constraint aimed at cutting off such a mixed integer solution, and iterates step 2. Otherwise, return the solution.

In other words, the OA linear inequalities are used to cut off mixed integer solutions of the auxiliary MILP, instead of tightening the convex relaxation of the original MINLP. In this paper we concentrate on strengthening a convex relaxation directly. We refer the reader specifically interested in OA techniques to the recent survey [17].

Over the last thirty years, cutting planes have become one of the essential ingredients in the solution of mixed integer linear programs. Among the most used methods Gomory Mixed Integer Cuts [29], Mixed Integer Rounding (MIR) cuts [39], Knapsack Covers [12] and Flow Covers [40, 43] are all employed in modern commercial and open-source MILP codes to strengthen linear programming relaxations. We refer to [22] for a recent survey of the associated theory and to [36] for the impact of these techniques in computations.

When moving from MILP to MINLP, many difficulties arise in the generation of cutting planes and the state of the art is much less developed than in MILP. In particular, so far, solvers for MINLP, whether commercial or academic, only scarcely rely on cutting planes.

A special case worth mentioning is mixed integer second-order cone programming where generalizations of the various techniques mentioned previously have been proposed: Atamtürk and Narayan [2] extended the concept of MIR, Cezik and Iyengar [19] extended Chvátal-Gomory cuts [28, 20] and lift-and-project cuts [7]. Drewes [24] made extensive computational experiments of these techniques.

Our focus here is on methods that address general MINLPs by using *disjunctive* arguments originally developed in the context of MILP by Balas [5]. In recent years, several authors have extended the theory and algorithms to the nonlinear case. At the end of the 90s, Ceria and Soares [18] extended the theorems on unions of polyhedra to general convex sets, and Stubbs and Mehrotra [47] extended the lift-and-project procedures of Balas, Ceria and Cornuéjols [7, 8] to the case of 0-1 convex MINLPs. These pioneering works laid out the basic theory to apply disjunctive programming techniques to MINLP but several technicalities made it difficult to move to practice. In effect, they were not used until very recently when several authors proposed different techniques with positive computational results. Saxena, Bonami and Lee [44, 45] addressed the case of indefinite quadratic constraints. Belotti [13] used similar techniques to strengthen the convex relaxations of factorable MINLPs. Finally, Kılınc, Linderoth and Luedtke [33, 35] and Bonami [15] independently revisited the case of convex MINLPs and proposed two different separation procedures. Our intent here is to review the common basic ideas behind these works.

There are many ways of generating cutting planes for MINLPs. Specifically, cuts can be either linear or nonlinear and can be derived by exploiting either convex NLP relaxations or LP ones. Although, we do not restrict our treatment of the subject to any of the specific options above, we will be mainly concerned with *linear* cutting planes, thus on LP relaxations. Indeed, by restricting the attention to linear cuts, then it is (relatively) easy to show that it is enough to concentrate on LP relaxations of C. A formal proof of this result is presented in Section 2.

In Section 3, we recall the basic construction of the concavity cut by Tuy [50]. This is for historical reasons, as it is for sure one of the first cutting plane construction in MINLP, but also because it gives a simple geometrical intuition. We then focus on the disjunctive programming techniques. Specifically, in Section 4 we introduce disjunctive programming basics and notation. In Section 5 we survey approaches for convex MINLPs, while in Section 6 we discuss the nonconvex MINLPs.

2 Linear Relaxations are enough for Linear Cuts

A basic question one may ask is wether it is really necessary to consider nonlinear convex sets when one is generating valid inequalities, or if one could always generate the same inequalities using polyhedral outer approximations. Here, we show that, under some common technical conditions, when one is restricted to linear valid inequalities (or cutting planes) the answer to this question is negative, i.e., it is not necessary to consider nonlinear sets. We also show an example that asserts the necessity of the technical conditions. Note nevertheless that the result is not constructive and in particular does not give any indication of how to obtain a good polyhedral outer approximation that can be used for generating cuts. This subject will be revisited in the context of lift-and-project cuts in Sections 5.1 and 5.2.

We suppose that the set C is bounded. Let l and u be large enough lower and upper bounds so that $C \subseteq \{x \in \mathbb{R}^n : l_j \leq x_j \leq u_j\}$. Let $\mathcal{Y} \stackrel{\text{def}}{=} \{y \in \mathbb{Z}^p : l_j \leq y_j \leq u_j,\, j = 1, \ldots, p\}$ be the set of all possible assignments for the integer-constrained variables. For each $y \in \mathcal{Y}$, we define

$$\xi(y, c) \stackrel{\text{def}}{=} \begin{cases} \arg\min\limits_{x \in C} \{c^\top x : x_j = y_j,\, j = 1, \ldots, p\} & \text{if it exists,} \\ \arg\min\limits_{x \in \mathbb{R}^n} \{\sum\limits_{i=1}^{m} \max(g_i(x), 0) : x_j = y_j,\, j = 1, \ldots, p\} & \text{otherwise.} \end{cases}$$

Hypothesis 1. *We suppose that, for a given $c \in \mathbb{R}^n$ and for every $y \in \mathcal{Y}$, a constraint qualification holds in the point $\xi(y, c)$.*

We can now recall the statement of the fundamental theorem by Duran and Grossmann [25].

Theorem 2 ([25, 27, 16])**.** *Consider a* convex *MINLP* (1) *and suppose that Hypothesis 1 holds. Then, there exists a matrix A and a vector b such that $C \subseteq \{x \in \mathbb{R}^n : Ax \geq b\}$ and*

$$\min_{x \in \mathbb{Z}^p \times \mathbb{R}^{n-p}} \{c^\top x : Ax \geq b\} = \min_{x \in X} c^\top x.$$

We can now show that if Hypothesis 1 holds for a given objective function $\alpha^\top x$, every valid inequality with left-hand-side $\alpha^\top x$ can be obtained by using an outer approximation.

Theorem 3. *Let $\alpha^\top x \geq \beta$ be a valid linear inequality for X. Suppose that the Hypothesis 1 holds with the objective function $\alpha^\top x$, then there exists a matrix A and a vector b such that $C \subseteq \{x \in \mathbb{R}^n : Ax \geq b\}$ and the inequality $\alpha^\top x \geq \beta$ is valid for the set $\mathcal{X} = \{x \in \mathbb{Z}^p \times \mathbb{R}^{n-p} : Ax \geq b\}$.*

Proof. Because Hypothesis 1 holds, by Theorem 2, there exists a matrix A and a vector b such that $C \subseteq \{x \in \mathbb{R}^n : Ax \geq b\}$ and

$$\hat{\beta} = \min_{x \in \mathcal{X}} \ \alpha^\top x = \min_{x \in X} \ \alpha^\top x.$$

Because $\alpha^\top x \geq \beta$ is valid for X, then $\beta \leq \hat{\beta}$ and therefore $\alpha^\top x \geq \beta$ is valid for $\mathcal{X}$ as well. $\square$

In the next example, we show that Hypothesis 1 is really necessary. In other words, if it is not satisfied, there exist valid inequalities that cannot be obtained by using an outer approximation.

Example 4. We consider a mixed integer set $X \in \mathbb{R}^2$ consisting of the intersection of a ball B centered in $(0.5, 0.5)$ and of radius 0.5, i.e.,

$$B \stackrel{\text{def}}{=} \left\{ (x_1, x_2) \in \mathbb{R}^2 : (x_1 - \frac{1}{2})^2 + (x_2 - \frac{1}{2})^2 \leq \frac{1}{4} \right\},$$

and $X = B \cap (\mathbb{Z} \times \mathbb{R})$. Figure 1 gives a geometrical representation of X and B, with the vertical lines defining the integrality of the first component x_1.

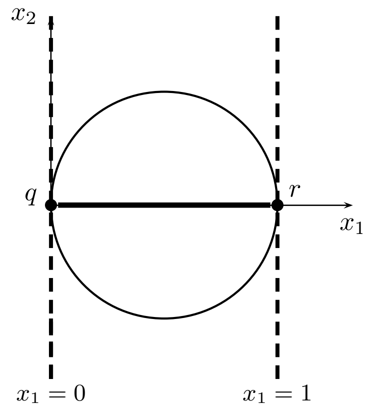


Figure 1: Illustration of the example. Set B is the inside of the circle, X is the intersection of B with the two vertical lines, i.e., points q and r.

The convex envelope of X is the segment joining the two points $q = (0, 0.5)$ and $r = (1, 0.5)$ and therefore the inequality $x_2 \geq 0.5$ is valid for X.

Let the unit vector $e_2 = (0, 1)$ be the objective function to be minimized. Note that the points q and r are $\xi(0, e_2)$ and $\xi(1, e_2)$, respectively. However, no constraint qualification holds in the two points, and therefore, Hypothesis 1 does not hold.

We now show that the inequality $x_2 \geq 0.5$ cannot be obtained using a polyhedral outer approximation for X, i.e., a polyhedron P containing B. To do that, note that the only valid inequality for B going through q is the tangent to B

in q, i.e., $x_1 = 0$, and therefore q cannot be an extreme point of any polyhedron P containing B. Moreover, every other valid inequality for B that intersects the vertical line $x_1 = 0$, intersects it either above or below q. Therefore, for any polyhedron P containing B there exists $\varepsilon_1 > 0$ such that $q \pm \varepsilon_1 e_2 \in P$. Finally, note that $q \pm \varepsilon_1 e_2 \in \mathbb{Z} \times \mathbb{R}$, and therefore $q \pm \varepsilon_1 e_2$ also belongs to $P \cap (\mathbb{Z} \times \mathbb{R})$.

Similarly for r, for any polyhedral outer approximation P, $\exists \varepsilon_2 > 0$ such that $r \pm \varepsilon_2 e_2 \in P \cap (\mathbb{Z} \times \mathbb{R})$.

It follows that the inequality $x_2 \geq 0.5$ is not valid for $P \cap (\mathbb{Z} \times \mathbb{R})$, since it would cut $q - \varepsilon_1 e_2$ and $r - \varepsilon_2 e_2$.

Nevertheless, for all practical purposes, given any $\varepsilon > 0$, the inequality $x_2 \geq 0.5 - \varepsilon$ can be obtained by using an outer approximation. $\square$

3 The Concavity Cut

In this section we discuss the classical work on concavity cuts by Tuy [50] that is for sure one of the first examples of cutting plane generation in MINLP, and gives a simple geometrical intuition for constructing cuts for both convex and nonconvex MINLPs. Tuy's construction was originally made in the context of concave minimization over a polyhedron. At the beginning of the 70s, the construction was used by Balas to derive intersection cuts [3, 4]. Since then, intersection cuts have been a fundamental tool in MILP (see, e.g., [21] for a recent survey). In that way, Tuy's construction is strongly related to disjunctive programming.

Given a polyhedral outer approximation $\mathcal{C} = \{x \in \mathbb{R}^n : Ax \geq b\}$ of the feasible region of X, and natural first step to take in order to solve (1) is to minimize $c^\top x$ over $\mathcal{C}$. Let $\overline{x}$ be a vertex of $\mathcal{C}$ attaining this minimum. Clearly, if $\overline{x} \in X$, it is also the minimum of (1).

Otherwise, if $\overline{x} \notin X$, suppose that we know a concave function $h(x)$ such that the constraint $h(x) \leq 0$ is verified by every point of X but not by $\overline{x}$. Finding such a function $h(x)$ might be non-trivial. In Tuy's original work, it was the objective function to be minimized. If (1) is not a convex MINLP, $h(x) \leq 0$ may be one of the original constraints of the problem. It is important to note that $h(x)$ may also reflect integrality constraints of the problem: for example taking $h(x) = (k + 1 - x_j)(x_j - k)$, for some $j \in \{1, \dots, p\}$ and $k \in \mathbb{Z}$ expresses the *simple disjunction* that if $x_j \in \mathbb{Z}$ then either $x_j \leq k$ or $x_j \geq k + 1$.

We now build the concavity cut by Tuy's method. To avoid technicalities, we assume here that $\mathcal{C}$ is full-dimensional and that $\overline{x}$ is a non-degenerate vertex of $\mathcal{C}$. These two hypotheses imply that $\mathcal{C}$ has exactly n edges $r^1, \dots, r^n$ emanating from $\overline{x}$ and the cone $T \stackrel{\text{def}}{=} \{x \in \mathbb{R}^n : x = \overline{x} + \sum_{j=1}^n \lambda_j r^j,\ \lambda \geq 0\}$ is full dimensional (i.e., $r^1, \dots, r^n$ are linearly independent). Note that the set $T \cap \{x \in \mathbb{R}^n : h(x) > 0\}$ is convex and does not contain any point of X.

For $j = 1, \dots, n$, let $\overline{\lambda}_j$ be a positive scalar such that $h(\overline{x} + \overline{\lambda}_j r^j) \leq 0$ ($\overline{\lambda}_j$ always exists since $h(\overline{x}) < 0$) and $\theta_j = \overline{x} + \overline{\lambda}_j r^j$. Since T is full dimensional, there is a unique hyperplane $\gamma^\top x = \gamma_0$ going through $\theta_1, \dots, \theta_n$, and by our definitions and assumptions $\gamma^\top \overline{x} \neq \gamma_0$. Without loss of generality, suppose that $\gamma^\top \overline{x} < \gamma_0$, then, by concavity of $h(x)$, $T \cap \{x \in \mathbb{R}^n : \gamma^\top \overline{x} < \gamma_0\} \subseteq T \cap \{x \in \mathbb{R}^n : h(x) > 0\}$, and therefore the inequality $\gamma^\top x \geq \gamma_0$ is satisfied by all points in X. Note that the portion of $\mathcal{C}$ cut by the inequality is bigger as the multiplier $\overline{\lambda}_j$ is bigger. Therefore, it is recommendable to choose $\overline{\lambda}_j$ such that θ_j is the intersection of the half-line $\overline{x} + \lambda r^j$ with $h(x) = 0$ if it exists. If there is no such intersection, $\overline{\lambda}_j$ can be arbitrarily large and the cut defined to be parallel to the ray r^j (see [32] for technical details).

Figure 2, gives a graphical illustration of the geometrical construction in two dimensions. It is worth noting that although Tuy's algorithm for concave minimization over a polytope does not converge, recently a proof of convergence was given for a slightly modified version of the algorithm [41].

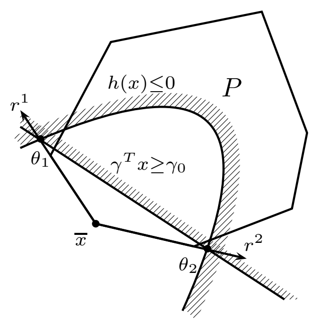


Figure 2: Geometrical construction of a concavity cut.

Tuy's construction illustrates how a nonconvex constraint can be used to strengthen a polyhedral relaxation of an MINLP. However, the construction has several limitations that make its use in the general setting difficult. First, the construction relies on a polyhedral relaxation and its generalization to more general convex sets is not straightforward. Moreover, in a general setting, one has to find systematically concave functions to construct the cut. In a sense, the various methods we review in this paper can be seen as ways to address these limitations.

4 Disjunctive Programming Basics

Disjunctive programming is interested in the description of the convex hull of unions of convex sets. The main results of disjunctive programming were initially established for unions of polyhedral sets by Balas [5, 6] and were later generalized to general convex sets. We briefly recall those results here. The results were established by Ceria, Soares [18] and/or Stubbs, Mehrotra [47].

We are given a collection of s closed convex sets $K^1, \ldots, K^s \subset \mathbb{R}^n$. Each set K^l, $l = 1, \ldots, s$ has the following representation

$$K^l \equiv \left\{x \in \mathbb{R}^n : g_i^l(x) \leq 0,\, i = 1, \ldots, m_l\right\},$$

where for $i = 1, \ldots, m_l$, $g_i^l : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ is continuously differentiable and convex.

Our intent is to be able to describe $\operatorname{conv}\left(K^1 \cup \ldots \cup K^s\right)$. This set is not closed in general (for example, take $s = 2$, K^1 an infinite line and K^2 a point outside of K^1). We will be only able to describe its topological closure. We denote by D the closure of the convex hull of this union: $D \stackrel{\text{def}}{=} \operatorname{cl}\left(\operatorname{conv}\left(K^1 \cup \ldots \cup K^s\right)\right)$.

The main theorem of disjunctive programming is a description of D by a convex set in a higher dimensional space. A tractable representation of the set uses the so-called perspective function that we define now. Let $f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ be a function, the perspective of f, denoted by $\tilde{f}$, is defined as $\tilde{f} : \mathbb{R}^{n+1} \to \mathbb{R}$ such that

$$(x, \lambda) \mapsto \tilde{f}(x, \lambda) \stackrel{\text{def}}{=} \begin{cases} \lambda f(x/\lambda) & \text{if } \lambda > 0 \\ +\infty & \text{otherwise.} \end{cases}$$

The main property of the perspective $\tilde{f}$ is that if f is convex, $\tilde{f}$ is convex and positively homogeneous (see [31], Chapter B, Section 2.2).

We can now state the main theorem.

Theorem 5. *[18] Let K^l, for $l = 1, \ldots, s$, and D be defined as above. Suppose additionally that $K^l \neq \emptyset$. Then,*

$$D \equiv \operatorname{proj}_x \operatorname{cl} \left\{ \begin{array}{ll} \tilde{g}_i^l(x^l, \lambda_l) \leq 0 & l = 1, \ldots, s,\ i = 1, \ldots, m_t \\ x = \displaystyle\sum_{l=1}^{s} x^l & \\ 1 = \displaystyle\sum_{l=1}^{s} \lambda_l & \\ x \in \mathbb{R}^n & \\ x^l \in \mathbb{R}^n & l = 1, \ldots, s \\ \lambda_l > 0 & l = 1, \ldots, s \end{array} \right\}.$$

In the following sections, we will study the use of Theorem 5 to generate cuts in various MINLP settings. Cuts are found by solving the so-called separation problem for D: given a point $\bar{x} \in \mathbb{R}^n$, either show that $\bar{x} \in D$ or find an hyperplane $\alpha^\top x = \beta$ such that $\alpha^\top x \geq \beta\ \forall x \in D$ and $\alpha^\top \bar{x} < \beta$

First, we recall how the separation problem can be solved in the simple case where the sets K^l are polyhedral. In that setting, disjunctive programming has been widely and successfully used. The basic results established by Balas [5, 6] were in particular used by Balas, Ceria and Cornuéjols in their seminal work on lift-and-project cuts [7, 8] for mixed integer linear programs.

Suppose now that for $l = 1, \ldots, s$, $K^l = \{x \in \mathbb{R}^n : A^l x \geq b^l\}$ (where A^l is an $n \times m_l$ matrix and $b^l \in \mathbb{R}^{m_l}$). By application of Theorem 5 we have

$$D \equiv \text{proj}_x \left\{ \begin{array}{ll} A^l x^l \geq b^l \lambda_l & l = 1, \ldots, s \\ x = \displaystyle\sum_{l=1}^{s} x^l & \\ 1 = \displaystyle\sum_{l=1}^{s} \lambda_l & \\ x \in \mathbb{R}^n & \\ x^l \in \mathbb{R}^n & l = 1, \ldots, s \\ \lambda_l \geq 0 & l = 1, \ldots, s \end{array} \right\}.$$

Balas [6] showed that separating a point from D can be done by solving a linear program as shown in the next theorem.

Theorem 6 ([6]). *$\bar{x} \in D$ if and only if the optimal value of the following Cut-Generation Linear Program (CGLP) is non-negative.*

$$\begin{array}{ll} \min \ \alpha^\top \bar{x} - \beta & \\ {u^l}^\top A^l \leq \alpha & l = 1, \ldots, s; \\ {u^l}^\top b^l \geq \beta, & l = 1, \ldots, s; \\ u^l \geq 0, & l = 1, \ldots, s; \\ \displaystyle\sum_{l=1}^{s} {\xi^l}^\top u^l = 1, & \end{array} \tag{CGLP}$$

where $\xi^l \in \mathbb{R}_+^{m_l}$, $l = 1, \ldots, m_l$.

Furthermore, if $(\alpha, \beta, u^1, v^1, \ldots, u^s, v^s)$ is a feasible solution of CGLP of negative cost, then $\alpha^\top x \geq \beta$ is a valid inequality for Q that cuts off $\bar{x}$.

The constraint $\sum_{l=1}^{s} {\xi^l}^\top u^l = 1$ of CGLP is often referred to as the *normalization constraint*. Note that it can be omitted from the statement of CGLP (i.e., the theorem remains true if $\xi^l = 0$), but it plays a central role in the practical efficiency and numerical stability of the cut (see, e.g., [9, 11, 26]). A common setting is to take $\xi^l = (1, 1, \ldots, 1)$.

5 Disjunctive Cuts for Convex MINLPs

We now study how integrality of certain variables can be used to strengthen the continuous relaxations of convex MINLPs. In that case, the standard way to construct a disjunctive relaxation is to intersect the feasible region of the continuous relaxation C with a so-called *simple disjunction* of the form $(x_j \leq k) \vee (x_j \geq k + 1)$ for some $k \in \mathbb{Z}$. Stubbs and Mehrotra [47] gave a mechanism for generating a linear inequality by finding a separating hyperplane from the set D corresponding to a simple disjunction.

Applying the disjunction $(x_j \le k) \vee (x_j \ge k+1)$ to the continuous relaxation of (1) yields the two convex sets

$$
\begin{aligned}
C_j^{k\downarrow} &\stackrel{\text{def}}{=} \{x \in \mathbb{R}^n \mid x_j \le k, g_i(x) \le 0 \quad i = 1, \dots, m\}, \\
C_j^{k\uparrow} &\stackrel{\text{def}}{=} \{x \in \mathbb{R}^n \mid x_j \ge k+1, g_i(x) \le 0 \quad i = 1, \dots, m\}.
\end{aligned}
$$

We denote by C_j^k the convex hull of the union of $C_j^{k\downarrow}$ and $C_j^{k\uparrow}$. Theorem 5 implies that C_j^k may be represented in an extended space as

$$
\tilde{C}_j^k = \left\{ (x, \tilde{y}, \tilde{z}, \lambda, \mu) \;\middle|\; \begin{array}{l} x = \tilde{y} + \tilde{z}, \\ \tilde{g}_i(\tilde{y}, \lambda) \le 0, \ \forall i = 1, \dots, m \\ \tilde{g}_i(\tilde{z}, \mu) \le 0, \ \forall i = 1, \dots, m \\ \tilde{y}_j \le \lambda k, \\ \tilde{z}_j \ge \mu(k+1), \\ \lambda + \mu = 1, \ \lambda \ge 0, \ \mu \ge 0, \\ \tilde{y} \in \mathbb{R}^n, \tilde{z} \in \mathbb{R}^n, \end{array} \right\},
$$

so that $\text{proj}_x(\tilde{C}_j^k) = \text{conv}(C_j^{k\downarrow} \cup C_j^{k\uparrow}) = C_j^k$.

A linear inequality separating a (fractional) point $\bar{x}$ with $k < \bar{x}_j < k+1$ from $\tilde{C}_j^k$ (and equivalently C_j^k) can be sought by projecting $\bar{x}$ onto $\tilde{C}_j^k$. The projection is accomplished by solving the convex optimization problem

$$
\min_{(x, \tilde{y}, \tilde{z}, \lambda, \mu) \in \tilde{C}_j^k} \quad d(x) \stackrel{\text{def}}{=} \|x - \bar{x}\|. \tag{2}
$$

Stubbs and Mehrotra demonstrated how to obtain a disjunctive inequality from a subgradient of the distance function at an optimal solution to (2).

Theorem 7 ([47])**.** *Let $\bar{x} \notin C_j^k$, x^* be an optimal solution of* (2)*, and let $\xi \in \partial d(x^*)$. Then, $\xi^\top(\bar{x} - x^*) < 0$ and $\xi^\top(x - x^*) \ge 0 \ \forall x \in C_j^k$.*

The procedure in Stubbs and Mehrotra is a generalization of the procedure described in the case of the union of polyhedra in [6], which has been implemented with great success as the so-called lift-and-project cut.

The separation problem (2) suggested by Stubbs and Mehrotra has two undesirable computational properties. First, the generation of a disjunctive inequality requires the solution of a nonlinear program of twice the number of variables as the original problem. Second, the description of the set $\tilde{C}_j^k$ onto which the point to be separated is projected contains (perspective) functions that are not everywhere differentiable. These two properties have hindered the use of disjunctive cutting planes for convex 0-1 MINLP. Stubbs and Mehrotra [47] report computational results only on four instances, the largest of which has $n = 30$ variables. An implementation of the Stubbs and Mehrotra procedure for the special case of Mixed Integer Second Order Cone Programming appears in the PhD thesis of Drewes [24].

Recently, a method that circumvents the difficulty associated with solving nonlinear programs by instead solving a sequence of cut-generating linear programs has been proposed in [33, 35]. A disjunctive cut is generated at each iteration of the procedure, and in the limit, the inequality is as strong as the one suggested by Stubbs and Mehrotra. Also recently, Bonami [15] suggested a method to generate disjunctive inequalities for MINLP in the space of the original variables. In Bonami's construction, a cut-generating convex programming problem is solved, but the number of nonlinear constraints is double the number in the original problem. This is an extension of the clever intuition of Balas and Perregaard [10], which allows this separation problem to be solved in the original space of variables for MILP. We now give an overview to each of these recent methods.

5.1 Disjunctive Cuts via Linear Programming Only

The basic idea of the LP-only approach of [33, 35] is to avoid solving the difficult nonlinear program (2), by instead solving a sequence of linear programs whose limiting solution is also a solution to (2). To describe the method, let $B \supseteq C$ be a relaxation of the original continuous relaxation, and let

$$B_i^{k\downarrow} \stackrel{\text{def}}{=} \{x \in B \mid x_i \leq k\} \quad \text{and} \quad B_i^{k\uparrow} \stackrel{\text{def}}{=} \{x \in B \mid x_i \geq k+1\}.$$

Inequalities valid for $\text{conv}(B_i^{k\downarrow} \cup B_i^{k\uparrow})$ are also valid for $\text{conv}(C_i^{k\downarrow} \cup C_i^{k\uparrow})$. Further, if B is restricted to be a polyhedron, and an appropriate norm is used in the definition of the distance in (2), the separation problem is a *linear* program. In fact, the dual of the linear program is related to the CGLP described in Theorem 6, the exact relation depending on the norm used to define the distance in (2) or equivalently on the normalization condition in the CGLP.

Zhu and Kuno [51] were the first to suggest such an approach, and they propose the outer approximating set

$$B \stackrel{\text{def}}{=} \{x \in \mathbb{R}^n \mid g(\bar{x}) + \nabla g(\bar{x})^\top (x - \bar{x}) \leq 0\},$$

where $\bar{x} \in \arg\min_{x \in C}\{c^\top x\}$ is an optimal solution to the continuous NLP relaxation. Their computational results show that adding disjunctive inequalities obtained from this polyhedral outer approximation can improve the solution time for small test instances.

The method of Zhu and Kuno could be iterated by dynamically adding the generated disjunctive inequalities to the set B. The updated set B is a tighter relaxation of C, so subsequently stronger disjunctive inequalities are generated. A natural question to ask is if by iterating the Zhu and Kuno procedure one can obtain a disjunctive inequality of the same strength as from solving the problem (2). Kılınç, Linderoth and Luedtke answer this question in the negative by giving an example where by iteratively adding disjunctive inequalities to the set B, (but not to the CGLP, so that only rank-one inequalities are generated), the final inequality generated is weaker than that generated by the Stubbs and Mehrotra procedure [34]. The example demonstrates that in order to generate stronger disjunctive inequalities, one must obtain tighter relaxations of the sets $C_i^{k\downarrow}$ and $C_i^{k\uparrow}$.

The method of [33, 35] iteratively updates polyhedral outer approximations of $C_i^{k\downarrow}$ and $C_i^{k\uparrow}$ so that in the limit the inequality is of the same strength as if the nonlinear separation problem (2) was solved. At iteration t, the method has two finite sets of points $\mathcal{K}_-^t, \mathcal{K}_+^t \subset \mathbb{R}^n$ about which linearizations of the nonlinear functions are taken, resulting in two polyhedral sets

$$\mathcal{F}_i^{k\downarrow,t} \stackrel{\text{def}}{=} \left\{x \in \mathbb{R}^n \mid x_i \leq k, \quad g(\bar{x}) + \nabla g(\bar{x})^\top (x - \bar{x}) \leq 0 \ \forall \bar{x} \in \mathcal{K}_-^t \right\} \text{ and}$$
$$\mathcal{F}_i^{k\uparrow,t} \stackrel{\text{def}}{=} \left\{x \in \mathbb{R}^n \mid x_i \geq k+1, \quad g(\bar{x}) + \nabla g(\bar{x})^\top (x - \bar{x}) \leq 0 \ \forall \bar{x} \in \mathcal{K}_+^t \right\}.$$

Since the sets $\mathcal{F}_-^t$ and $\mathcal{F}_+^t$ are polyhedral, a disjunctive inequality can be obtained as suggested by Theorem 6, and since $C_i^{k\downarrow} \subseteq \mathcal{F}_i^{k\downarrow,t}$ and $C_i^{k\uparrow} \subseteq \mathcal{F}_i^{k\uparrow,t}$ $\forall t$, the inequality is valid. The algorithm starts with $\mathcal{K}_-^0 = \mathcal{K}_+^0 = \emptyset$ and augments the sets based on the solution of the CGLP. Specifically, at iteration t, the dual of the separation problem is solved, yielding a solution point x^t, as well as two points, called *friends* by [23]. The friends points y^t and z^t have the property that $x^t = \lambda y^t + (1-\lambda^t) z^t$ for some $\lambda \in [0, 1]$. The key insight of the work [33, 35] is that augmenting the sets $\mathcal{K}_-^t$ and $\mathcal{K}_+^t$ with the friend points (respectively) y^t and z^t is sufficient for an iterative procedure to generate an inequality (in the limit) as strong as if the nondifferentiable, nonlinear program (2) was solved.

A computational advantage of the LP-based approach lies in the fact that a valid inequality is produced at every iteration. Thus, the process can be terminated early if it is observed that the cuts are not effective reducing the solution integrality gap.

Kılınç, Linderoth and Luedtke [33, 35] applied their separation procedure in the context of the LP/NLP based branch and bound of FilMINT [1]. They compared experimentally the total solution time taken to solve 207 publicly available instances with and without their lift-and-project cuts. They report that

- for 148 instances solved without cuts in less than five minutes, generating cuts does not decrease computating times but the increase is very limited (4% on average);
- for 30 instances that take more than 5 minutes but less than 3 hours to solve without cutting planes the solution is more than 3 times faster with cutting plane;
- 12 instances that cannot be solved in less than 3 hours without cutting planes are solved with the same time limit when using cuts.

5.2 Disjunctive Cuts via Nonlinear Programming

As noted in Section 5, the model proposed by Stubbs and Mehrotra is difficult to solve as a nonlinear program because of its size and its non-differentiability. We present another mechanism proposed by Bonami [15] that circumvents these difficulties in the specific setting where a given point $\overline{x}$ is to be separated from a disjunctive relaxation obtained by using simple disjunction.

First, since C_j^k is a convex set, a separating hyperplane exists if and only if $\overline{x} \notin C_j^k$ or equivalently if there is no $\lambda, \mu \in \mathbb{R}_+$ and $\tilde{y}, \tilde{z} \in \mathbb{R}_+^n$ such that $(\overline{x}, \tilde{y}, \tilde{z}, \lambda, \mu) \in \tilde{C}_j^k$. We now focus on the solution of this system of inequalities (i.e., the construction of an hyperplane is delayed to the end of the section) and show that in our special case $\tilde{C}_j^k$ can be substantially simplified.

Without loss of generality, we can assume that in the formulation of $\tilde{C}_j^k$, $\tilde{y}_j = \lambda k$ and $\tilde{z}_j = \mu(k+1)$. Using the equations $\lambda + \mu = 1$ and $\overline{x}_j = \tilde{y}_j + \tilde{z}_j$, we obtain that $\overline{x}_j = k + \mu$. In the remainder we denote by f_0 the quantity $\overline{x}_j - k$. We have established that $\mu = f_0$ and $\lambda = 1 - f_0$ are fixed (note that by assumption $0 < f_0 < 1$ and therefore the constraints are well defined). Finally, we can eliminate the variables $\tilde{y}$ by using the equation $\tilde{y} = \overline{x} - \tilde{z}$. Summing up everything, we obtain that $\overline{x} \in C_j^k$ if and only if the system

$$
\begin{aligned}
&\tilde{g}_i(\overline{x} - \tilde{z}, 1 - f_0) \leq 0, && \forall i = 1, \ldots, m \\
&\tilde{g}_i(\tilde{z}, f_0) \leq 0, && \forall i = 1, \ldots, m \\
&\tilde{z}_j = f_0(k+1), && \\
&\tilde{z} \in \mathbb{R}^n &&
\end{aligned}
\tag{3}
$$

admits a solution.

To determine if (3) has a solution, Bonami [15] proposes to solve the convex non-linear program

$$
\begin{aligned}
\max\ & \tilde{z}_j \\
& g_i\left(\frac{\tilde{z}}{f_0}\right) \leq 0 \qquad i = 1, \ldots, m, \\
& g_i\left(\frac{\overline{x} - \tilde{z}}{1 - f_0}\right) \leq 0 \quad i = 1, \ldots, m. \\
& \tilde{z} \in \mathbb{R}^n,
\end{aligned}
\tag{MNLP}
$$

where the last equation system (3) is used to define the objective function.

It is shown in [15] that the optimal value of (MNLP) is smaller than $f_0(k+1)$ if and only if $\overline{x} \notin C_j^k$.

The nonlinear program (MNLP) has a technical deficiency in that it does not satisfy any constraint qualification whenever $\overline{x}$ is an extreme point of the convex region C. To circumvent the problem, it is proposed to solve the perturbed version of the problem

$$
\begin{aligned}
\max\ & \tilde{z}_j \\
& g_i\left(\frac{\tilde{z}}{f_0}\right) \le f_0(k+1) - \tilde{z}_j \qquad i = 1, \dots, m, \\
& g_i\left(\frac{\overline{x} - \tilde{z}}{1 - f_0}\right) \le f_0(k+1) - \tilde{z}_j \quad i = 1, \dots, m. \\
& \tilde{z} \in \mathbb{R}^n.
\end{aligned}
\tag{MNLP'}
$$

This problem shares the property of (MNLP) that its optimal value is smaller than $f_0(k+1)$ if and only if $\overline{x} \notin C_j^k$. Contrary to the model proposed by Stubbs and Mehrotra, (MNLP') does not require the introduction of additional variables and differentiability is maintained.

Finally, one needs to compute the equation of the separating hyperplane. It is proposed in [15] to build a linear model similar to the one used in the previous section except that no constraint generation is done. In particular, it is shown that by choosing $\mathcal{K}_-^0 = \left\{\frac{\overline{x}-\tilde{z}}{1-f_0}\right\}$ and $\mathcal{K}_+^0 = \left\{\frac{\tilde{z}}{f_0}\right\}$ a cut is always found by the CGLP, provided that $\overline{x} \notin C_j^k$ and the optimal solution of (MNLP') satisfies a constraint qualification.

The technique was tested in the NLP based branch and bound of Bonmin [16], by comparing the total solution time with and without cuts on a set of 80 instances that take more than 1000 nodes to solve with pure branch and bound. The results are that 63 instances can be solved without cuts and 77 can be solved with cuts. On instances solved by both techniques the number of branch-and-bound nodes is reduced on average by 21% and the CPU time by 19%. The author notes nevertheless that, in these experiment, cuts have a positive impact on only about half of the testset.

6 Disjunctive Cuts for Nonconvex MINLPs

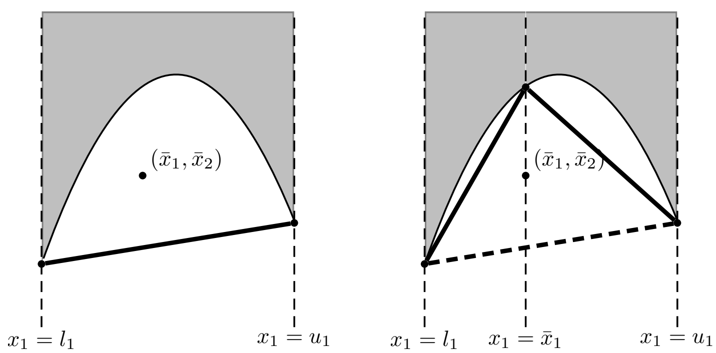
A common approach for solving nonconvex MINLP problems is to construct a convex relaxation C of the nonconvexities and to refine the relaxation using a so-called *spatial* branch-and-bound approach. Similar to the convex case, cutting planes can be used to strengthen the convex relaxation at any node of the branch-and-bound tree. Using the methods presented in Section 5, it is evident that one can strengthen this convex relaxation by using disjunctions based on the integrality requirements of the problem. However, integrality is not the only source for cutting planes. Indeed, as illustrated by Tuy's construction presented in Section 3, nonconvex constraints violated by the solution of the current relaxation can also be used to generate cutting planes.

In this section we will then concentrate on obtaining cuts from disjunctions that are *not* associated with the integrality requirements, but those that can be derived from other nonconvex constraints. Since we place ourselves in the context where we have at hand a convex relaxation C of the problem, the main question is how to identify appropriate disjunctions. Once a disjunction is found, cuts can be generated by applying the general theory of disjunctive programming presented in Section 4.

One general principle used to build disjunctions is simple and we start by illustrating it with one nonconvex constraint of a very simple form: $h(x_2) \le x_1$, where $h : \mathbb{R} \to \mathbb{R}$ is a concave function, and x_1 has finite lower and upper bounds l_1 and u_1, respectively. This setting might seem artificially simple but, to the best of our knowledge, all cutting plane methods for nonconvex MINLP problems consider one constraint of this form at a time to build a disjunction.

Figure 3 illustrates the construction of a disjunction. We consider the set $H \stackrel{\text{def}}{=} \{(x_1, x_2) \in \mathbb{R}^2 : h(x_2) \le x_1,\ l_1 \le x_1 \le u_1\}$. The convex hull of H is the set of points above the segment joining the two intersections of $h(x_2) = x_1$ with $x_1 = l_1$ and $x_1 = u_1$ respectively (see Figure 3(a)):

$$\operatorname{conv}(H) = \{(x_1, x_2) \in \mathbb{R}^2 : \frac{h(u_1) - h(l_1)}{u_1 - l_1} x_1 + \frac{h(u_1)l_1 - h(l_1)u_1}{u_1 - l_1} \le x_2,\ l_1 \le x_1 \le u_1\}.$$



(a) The set H and its convex hull. H is the gray shaded set of points above the parabola and its convex hull the set of points above the segment joining the two intersection of the parabola with $x_1 = l_1$ and $x_1 = u_1$.

(b) Construction of a disjunction that excludes the point $\bar{x}$. Any point of H is either on the left side of $x_1 = \bar{x}_1$ and above the left segment, or on the right side and above the right segment.

Figure 3: Construction of a disjunction from a constraint described by an univariate concave function.

Let $\bar{x}$ be the solution to be cut. Our goal is to build a disjunction that excludes $\bar{x}$. We suppose that (i) $h(\bar{x}_1) > \bar{x}_2$, since otherwise no cut can be generated using the constraint $h(x_1) \le x_2$, and (ii) $(\bar{x}_1, \bar{x}_2) \in \operatorname{conv}(H)$, otherwise $\bar{x}$ can be cut by the inequality $\frac{h(u_1)-h(l_1)}{u_1-l_1} x_1 + \frac{h(u_1)l_1-h(l_1)u_1}{u_1-l_1} \le x_2$ used to describe $\operatorname{conv}(H)$.

Consider the two sets $H^{\downarrow} = H \cap \{(x_1, x_2) : x_1 \le \bar{x}_1\}$ and $H^{\uparrow} = H \cap \{(x_1, x_2) : x_1 \ge \bar{x}_1\}$ (evidently, $H = H^{\downarrow} \cup H^{\uparrow}$). A valid two-term disjunction can be simply obtained by convexifying $H^{\downarrow}$ and $H^{\uparrow}$ independently. Indeed

$H \subseteq \text{conv}(H^{\downarrow}) \cup \text{conv}(H^{\uparrow})$ and $\bar{x} \notin \text{conv}(H^{\downarrow}) \cup \text{conv}(H^{\uparrow})$ (because $h(\bar{x}_1) > \bar{x}_2$). Therefore, $\big((\bar{x}_1, \bar{x}_2) \in H^{\downarrow}\big) \vee \big((\bar{x}_1, \bar{x}_2) \in H^{\uparrow}\big)$ is a valid disjunction excluding $\bar{x}$. It is important to note that no cut can be obtained by considering solely the disjunction $\big((\bar{x}_1, \bar{x}_2) \in H^{\downarrow}\big) \vee \big((\bar{x}_1, \bar{x}_2) \in H^{\uparrow}\big)$ because $(\bar{x}_1, \bar{x}_2) \in \text{conv}(\text{conv}(H^{\downarrow}) \cup \text{conv}(H^{\uparrow}))$, but one has to intersect the disjunction with the convex relaxation C of the complete problem. Even in this case, there is no guarantee that $\bar{x}$ can be cut unless $\bar{x}$ is an extreme point of C.

We now discuss two works that use this simple way of constructing disjunctions in a more general setting.

6.1 Indefinite Quadratic Constraints

Saxena, Bonami and Lee [44, 45] proposed two methods for separating disjunctive cuts for constraints represented by quadratic functions of the form $g_i(x) = x^\top A^i x + {a^i}^\top x + a_0^i \leq 0$, where A^i is an $n \times n$ indefinite symmetric matrix, a^i is an n-dimensional vector and a_0^i is a scalar. Furthermore, all the variables x_j appearing in products (i.e., such that $\exists i, k$ with $A^i_{kj} \neq 0$) are assumed to have finite lower and upper bounds l_j, u_j, respectively. In [44, 45], it is assumed that all constraints are quadratic, but it is straightforward to see that the approaches can also be applied if only a subset of constraints are quadratic.

The first method proposed in [44] is based on a standard reformulation-linearization technique applied to the quadratic constraints. An $n \times n$ matrix of auxiliary variables Y representing the products of variables is introduced: $Y = xx^\top$. The quadratic constraints are then reformulated as linear constraints: $\langle Y, A^k \rangle + {a^k}^\top x + a_0^k \leq 0$. The nonconvex constraint $Y = xx^\top$ can then be relaxed using two types of constraints: the convex semi-definite constraint $Y - xx^\top \succeq 0$ and the RLT inequalities (see [37, 46])

$$\max \left\{ \begin{array}{c} l_i x_j + l_j x_i - l_i l_j \\ u_i x_j + u_j x_i - u_i u_j \end{array} \right\} \leq y_{ij} \leq \min \left\{ \begin{array}{c} u_i x_j + l_j x_i - u_i l_j \\ l_i x_j + u_j x_i - l_i u_j \end{array} \right\}. \tag{4}$$

We call the convex relaxation using only the constraints (4) RLT and the one using both (4) and $Y - xx^\top \succeq 0$ SDP-RLT.

Disjunctive cuts can be used to strengthen the SDP-RLT relaxation in the following manner. Let $(\bar{x}, \bar{Y})$ be a solution to the SDP-RLT relaxation such that $\bar{Y} \neq \bar{x}\bar{x}^\top$. Because $\bar{Y} - \bar{x}\bar{x}^\top \succeq 0$ and $\bar{Y} \neq \bar{x}\bar{x}^\top$, the matrix $\bar{Y} - \bar{x}\bar{x}^\top$ certainly has at least one positive eigenvalue λ with corresponding eigenvector v. It follows that $(\bar{x}, \bar{Y})$ satisfies the inequality $v^\top \bar{Y} v - (v^\top \bar{x})^2 = \lambda v^\top v > 0$. However, the nonconvex constraint $Y - xx^\top = 0$ implies that $v^\top Y v - (v^\top x)^2 \leq 0$ is a valid (nonconvex) constraint for the problem. We can now use this last constraint to generate a valid disjunction of the same form as the one exhibited in Figure 3(b) in the space spanned by $v^\top x$ and $v^\top Y v$. Once this disjunction is obtained, it can be used to generate a cut using standard disjunctive programming techniques.

In [44], cuts are separated by using a linear outer-approximation of the convex relaxation. (Specifically, the constraint $Y - xx^\top \succeq 0$ is approximated with only a few supporting hyperplanes). The authors develop a cutting plane algorithm where at each iteration one cut is generated for each positive eigenvalue of the matrix $\bar{Y} - \bar{x}\bar{x}^\top$. Computational experiments show that the addition of disjunctive cuts to the SDP-RLT relaxation gives a substantial improvement of the bound obtained for many problems. On a test set of 129 moderate-size instances from MINLPlib, for 67 of the instances, the disjunctive approach closes more than an additional 98% of the gap of the RLT relaxation. The average gap closed by the disjunctive approach is 76%, while the gap closed by the SDP-RLT relaxation is 25%.

It is worth noting that directions v corresponding to eigenvectors are not the only ones that can be used to generate cuts. Indeed, any vector v such that $v^\top \bar{Y} v - (v^\top \bar{x})^2 > 0$ can be used to generate a disjunctive cut. Also in [44], an improved variant of the algorithm is proposed where, in addition to eigenvectors, cuts are generated using directions for which the error induced by convexifying the constraint $v^\top \bar{Y} v - (v^\top \bar{x})^2 \leq 0$ is greatest. The addition of these disjunctions allows to close about 80% of the gap of the RLT relaxation on the same test set as before.

The second type of cut proposed by Saxena, Bonami and Lee [45] results from using a different convex relaxation. The basic idea of the relaxation in [45] is to project the extended formulation with the RLT constraints (4) into the space of x variables. Disjunctive cuts are then generated using spatial disjunctions of the form

$$\left(x_j \leq \frac{l_j + u_j}{2} \right) \vee \left(x_j \geq \frac{l_j + u_j}{2} \right).$$

Noting that the projection of the RLT inequalities (4) gives rise to inequalities whose coefficients depend on the bounds on the variables l_j and u_j, the modified bound on x_j is *propagated* to the inequalities defining the projection on the two sides of the disjunction. A cut is then separated using the CGLP. Using this technique on the same testset as before, Saxena, Bonami and Lee [45] close almost the same gap as with the cuts in the extended space (76% of the gap of the RLT relaxation), but the computing times are almost two orders of magnitude faster.

6.2 Factorable MINLPs

Belotti [13] proposes to use disjunctive cuts in the more general setting of factorable MINLPs. A function is said to be factorable if it can be expressed as a finite recursion of finite sums and products of univariate functions. An MINLP is factorable if all the nonlinear functions g_i describing its feasible region are factorable [48].

In practice, solvers for factorable MINLP only consider a subset of factorable functions that can be obtained by using a pre-defined set of univariate functions (e.g., $\{\log, \exp, 1/x, \ldots\}$). Factorability is essential because it is used by the solver to automatically build convex relaxations of the problem. In a first step, the problem is reformulated by adding a number of artificial variables $x_{n+1}, \ldots, x_{n+K}$ into the problem. The new variables x_k are used to express the nonlinear expressions as $x_k = \psi(x_j)$ (where ψ is one of the predefined univariate functions) or $x_k = x_j x_l$. The factorability of the underlying problem guarantees that this reformulation can always be completed. In a second step the convex relaxation is formed by convexifying each of the nonlinear constraints independently using predefined convexifications (for example, $x_k = x_j x_l$ is convexified using the inequalities (4)). The convexification has two important properties: (i) the relaxation is exact when the variables involved in the functions to convexify attain their bounds; and (ii) the quality of the relaxation depends on these bounds.

An essential tool for solving factorable MINLP problems is bound tightening and bound propagation [30, 38, 42, 48, 14]. These techniques are used to tighten the convex relaxation when the bound on a variable is modified. The combination of the convexification method described above and bound propagation has led to the development of several branch-and-bound algorithms specifically designed for factorable MINLPs.

A weakness of the convexification method used in these algorithms is that each nonlinear function is convexified separately. One interest of using disjunctive cuts for solving nonconvex MINLP problems is that their use incoporates interactions between different nonlinear functions, resulting in a tighter relaxation.

Despite the more general setting than that described in Section 6.1, disjunctive cuts can be derived in a very similar way. Suppose that the constraint $x_k = \psi(x_j)$ is not satisfied by the solution to be cut $\bar{x}$. It is important to remember that ψ is a function that belongs to a finite library of simple functions known by the solver. In particular, by examining the second derivative, it is known on what portions of its domain ψ is concave or convex. If $\bar{x}_k < \psi(\bar{x}_j)$, and ψ is convex in $\bar{x}_j$, $\bar{x}$ can simply be cut by an outer approximation constraint. Therefore, we suppose that $\bar{x}_k < \psi(\bar{x}_j)$ and ψ is concave in $\bar{x}_j$. Then, we are in the case exhibited in Figure 3. The spatial disjunction

$$x_j \leq \theta \vee x_j \geq \theta$$

(where θ is any number in the interval $[l_j, u_j]$) is imposed, each side of the disjunction is intersected with the convex relaxation, and finally bound propagation algorithms are used to tighten each side of the disjunction. The result is a disjunction between two convex sets to which $\bar{x}$ does not belong. We note that in the simple case with only one constraint, the effect is similar to the disjunction exhibited in Figure 3(b), but in general it leads to a more involved disjunction that uses several constraints of the problem.

In Belotti's method θ does not necessarily have to coincide with the value of variable x_j in the solution to be cut. Several rules have been devised to select this value, the reader is referred to [14, 49].

Another difficulty is that there are usually too many disjunctions that can be used to generate cuts and one typically has to choose a subset of "good" candidates. Belotti [13] proposes to separate cuts for disjunctions corresponding to the 20 most violated non-linear constraints. The method is implemented and tested in the open-source solver Couenne [14] on a set of 84 publicly available instances. It is reported in [13] that the variant with cuts can solve more instances in a time limit of 3 hours (39 instances solved versus 26 without cuts).

References

[1] K. Abhishek, S. Leyffer, and J. Linderoth. FilMINT: An Outer Approximation-Based Solver for Convex Mixed-Integer Nonlinear Programs. *INFORMS Journal on Computing*, 22:555–567, 2010.

[2] A. Atamtürk and V. Narayanan. Conic mixed integer rounding cuts. *Mathematical Programming*, 122:1–20, 2010.

[3] E. Balas. Intersection cuts - a new type of cutting planes for integer programming. *Operations Research*, 19:19–39, 1971.

[4] E. Balas. Integer programming and convex analysis: Intersection cuts from outer polars. *Mathematical Programming*, 2:330–382, 1972.

[5] E. Balas. Disjunctive programming. *Annals of Discrete Mathematics*, 5:3–51, 1979.

[6] E. Balas. Disjunctive programming: Properties of the convex hull of feasible points. *Discrete Applied Mathematics*, 89:3–44, 1998. (originally MSRR # 348, Carnegie Mellon University, July 1974).

[7] E. Balas, S. Ceria, and G. Cornuéjols. A lift-and-project cutting plane algorithm for mixed 0-1 programs. *Mathematical Programming*, 58:295–324, 1993.

[8] E. Balas, S. Ceria, G. Cornuéjols, and N. Natraj. Gomory cuts revisited. *Operations Research Letters*, 19:1–9, 1996.

[9] E. Balas and M. Perregaard. Lift and project for mixed 0-1 programming: Recent progress. *Discrete Applied Mathematics*, 123:129–154, 2002.

[10] E. Balas and M. Perregaard. A precise correspondence between lift-and-project cuts, simple disjunctive cuts, and mixed integer Gomory cuts for 0-1 programming. *Mathematical Programming*, 94:221–245, 2003.

[11] E. Balas and A. Saxena. Optimizing over the split closure. *Mathematical Programming*, 113:219–240, 2008.

[12] E. Balas and E. Zemel. Facets of the knapsack polytope from minimal covers. *SIAM Journal on Applied Mathematics*, 34:119–148, 1978.

[13] P. Belotti. Disjunctive cuts for non-convex MINLP. Technical report, Lehigh University, 2009.

[14] P. Belotti, J. Lee, L. Liberti, F. Margot, and A. Wächter. Branching and bounds tightening techniques for non-convex MINLP. *Optimization Methods and Software*, 24:597–634, 2009.

[15] P. Bonami. Lift-and-project cuts for mixed integer convex programs. In O. Günlük and G. Woeginger, editors, *Integer Programming and Combinatoral Optimization*, volume 6655 of *Lecture Notes in Computer Science*, pages 52–64. Springer Berlin / Heidelberg, 2011.

[16] P. Bonami, L.T. Biegler, A.R. Conn, G. Cornuéjols, I.E. Grossmann, C.D. Laird, J. Lee, A. Lodi, F. Margot, N. Sawaya, and A. Wächter. An algorithmic framework for convex mixed integer nonlinear programs. *Discrete Optimization*, 5:186–204, 2008.

[17] P. Bonami, M. Kılınç, and J. Linderoth. Algorithms and software for convex mixed integer nonlinear programs. In Jon Lee and Sven Leyffer, editors, *Hot Topics in Mixed Integer Nonlinear Programming*, IMA Volumes. Springer Science and Business Media, LLC, to appear.

[18] S. Ceria and J. Soares. Convex programming for disjunctive optimization. *Mathematical Programming*, 86:595–614, 1999.

[19] M.T. Cezik and G. Iyengar. Cuts for mixed 0-1 conic programming. *Mathematical Programming*, 104:179–202, 2005.

[20] V. Chvátal. Edmonds polytopes and a hierarchy of combinatorial optimization. *Discrete Mathematics*, 4:305–337, 1973.

[21] M. Conforti, G. Cornuéjols, and G. Zambelli. Corner polyhedron and intersection cuts. *Surveys in Operations Research and Management Science*, 16:105–120, 2011.

[22] G. Cornuéjols, M. Conforti, and G. Zambelli. *50 Years of Integer Programming 1958-2008*, chapter Polyhedral Approaches to Mixed Integer Programming, pages 343–385. Springer-Verlag Berlin Heidelberg, 2009.

[23] S. Dash, O. Günlük, and J.P. Vielma. Computational experiments with cross and crooked cross cuts, 2011. Available at `http://www.optimization-online.org/DB_HTML/2011/06/3071.html`.

[24] S. Drewes. *Mixed Integer Second Order Cone Programming*. PhD thesis, Technische Universität Darmstadt, 2009.

[25] M.A. Duran and I. Grossmann. An outer-approximation algorithm for a class of mixed-integer nonlinear programs. *Mathematical Programming*, 36:307–339, 1986.

[26] M. Fischetti, A. Lodi, and A. Tramontani. On the separation of disjunctive cuts. *Mathematical Programming*, 128:205–230, 2011.

[27] R. Fletcher and S. Leyffer. Mixed integer nonlinear programming by outer approximation. Technical report, University of Dundee, 1996.

[28] R.E. Gomory. Outline of an algorithm for integer solutions to linear programs. *Bulletin of the American Mathematical Society*, 64:275–278, 1958.

[29] R.E. Gomory. An algorithm for integer solution solutions to linear programming. In R. L. Graves and P. Wolfe, editors, *Recent Advances in Mathematical Programming*, pages 269–302. McGraw-Hill, 1963.

[30] E. Hansen. *Global Optimization Using Interval Analysis*. M. Dekker, 1992.

[31] J.-B. Hiriart-Urruty and C. Lemarechal. *Fundamentals of convex analysis*. Springer-Verlag, Berlin, Germany, 2001.

[32] R. Horst and H. Tuy. *Global Optimization*. Springer-Verlag, New York, 1993.

[33] M. Kılınç, J. Linderoth, and J. Luedtke. Effective separation of disjunctive cuts for convex mixed integer nonlinear programs. Technical Report TR1681, Computer Sciences Department, University of Wisconsin-Madison, 2010.

[34] M. Kılınç, J. Linderoth, and J. Luedtke. Computationally effective disjunctive cuts for convex mixed integer nonlinear programs, May 2011. Presentation at SIAM Optimization Conference.

[35] M. Kılınç. *Disjunctive Cutting Planes and Algorithms for Convex Mixed Integer Nonlinear Programming*. PhD thesis, University of Wisconsin-Madison, 2011.

[36] A. Lodi. *50 Years of Integer Programming 1958-2008*, chapter Mixed Integer Programming Computation, pages 619–645. Springer-Verlag Berlin Heidelberg, 2009.

[37] G.P. McCormick. Computability of global solutions to factorable nonconvex programs: Part I—Convex underestimating problems. *Mathematical Programming*, 10:147–175, 1976.

[38] R.E. Moore. *Methods and Applications of Interval Analysis*. SIAM, Philadelphia, 1979.

[39] G.L. Nemhauser and L.A. Wolsey. A recursive procedure to generate all cuts for $0-1$ mixed integer programs. *Mathematical Programming*, 46:379–390, 1990.

[40] M. Padberg, T.J. Van Roy, and L.A. Wolsey. Valid linear inequalities for fixed charge problems. *Operations Research*, 33:842–861, 1985.

[41] M. Porembski. Finitely convergent cutting planes for concave minimization. *Journal of Global Optimization*, 20:109–132, 2001.

[42] H. Ratschek and J. Rokne. Interval methods. In R. Horst and P.M. Pardalos, editors, *Handbook of Global Optimization*, pages 751–828. Kluwer Academic Publishers, Dordrecht, 1995.

[43] T.J. Van Roy and L.A. Wolsey. Solving mixed integer programming problems using automatic reformulation. *Operations Research*, 35:45–57, 1987.

[44] A. Saxena, P. Bonami, and J. Lee. Convex relaxations of non-convex mixed integer quadratically constrained programs: extended formulations. *Mathematical Programming*, 124:383–411, 2010.

[45] A. Saxena, P. Bonami, and J. Lee. Convex relaxations of non-convex mixed integer quadratically constrained programs: projected formulations. *Mathematical Programming*, pages 1–55, On line first. 10.1007/s10107-010-0340-3.

[46] H.D. Sherali and W.P. Adams. *A reformulation-linearization technique for solving discrete and continuous nonconvex problems*. Kluwer Academic Publishers, 1999.

[47] R. Stubbs and S. Mehrotra. A branch-and-cut method for 0-1 mixed convex programming. *Mathematical Programming*, 86:515–532, 1999.

[48] M. Tawarmalani and N.V. Sahinidis. *Convexification and Global Optimization in Continuous and Mixed-Integer Nonlinear Programming: Theory, Algorithms, Software, and Applications*. Kluwer Academic Publishers, Boston MA, 2002.

[49] M. Tawarmalani and N.V. Sahinidis. Global optimization of mixed integer nonlinear programs: A theoretical and computational study. *Mathematical Programming*, 99:563–591, 2004.

[50] H. Tuy. Concave programming under linear constraints. *Doklady Akademii Nauk SSSR*, 1964. English translation: *Soviet Mathematics* (1964) 1437-1440.

[51] Y. Zhu and T. Kuno. A disjunctive cutting-plane-based branch-and-cut algorithm for 0-1 mixed-integer convex nonlinear programs. *Industrial and Engineering Chemistry Research*, 45:187–196, 2006.