J Glob Optim
DOI 10.1007/s10898-016-0401-0

Global optimization of non-convex generalized disjunctive programs: a review on reformulations and relaxation techniques

Juan P. Ruiz[2] · **Ignacio E. Grossmann**[1]

Received: 30 June 2015 / Accepted: 5 January 2016

Abstract In this paper we present a review on the latest advances in logic-based solution methods for the global optimization of non-convex generalized disjunctive programs. Considering that the performance of these methods relies on the quality of the relaxations that can be generated, our focus is on the discussion of a general framework to find strong relaxations. We identify two main sources of non-convexities that any methodology to find relaxations should account for. Namely, the one arising from the non-convex functions and the one arising from the disjunctive set. We review the work that has been done on these two fronts with special emphasis on the latter. We then describe different logic-based optimization techniques that make use of the relaxation framework and its impact through a set of numerical examples typically encountered in Process Systems Engineering. Finally, we outline challenges and future lines of work in this area.

Keywords Disjunctive programming · Non-convex optimization · Relaxations

1 Introduction

Mixed-integer nonlinear programming (MINLP) [10] is a well known framework to represent optimization problems that deal with discrete and continuous variables, where the model is mainly described by using algebraic equations defined on the discrete and continuous space. In order to represent accurately the behavior of complex systems, many nonlinear expressions are often used. In general, this leads to an MINLP where the solution space is non-convex, and hence, difficult to solve since this may give rise to local solutions that are suboptimal.

✉ Ignacio E. Grossmann
grossmann@cmu.edu

Juan P. Ruiz
juan.ruiz@soteicavisualmesa.com

1 Carnegie Mellon University, Pittsburgh, PA, USA

2 Soteica Visual Mesa LLC, Austin, TX, USA

Published online: 20 January 2016

In the last decades many global optimization algorithms for non-convex problems have been proposed [7,33]. To prove optimality of the solution, most of these methods rely on finding upper and lower bounds of the global optimum until their corresponding gap lies within a given tolerance. The lower bound prediction is often achieved by solving a continuous convex relaxation of the MINLP. The tighter the relaxation the closer the lower bound to the global optimum and that is why a large part of the research has been related to finding tighter relaxations. However, in general, finding the global optimum of large-scale non-convex MINLP models in reasonable computational time remains a largely unsolved problem.

Even though the MINLP framework has been successfully used in many different areas, it greatly relies on the expertise of the modeler to generate models that are tractable and effective to solve. With this in mind, in order to facilitate the generation of effective models, Raman and Grossmann [22] proposed the Generalized Disjunctive Programming framework, which can be regarded as an extension of Disjunctive Programming [3]. This alternative strategy not only considers algebraic expressions but also disjunctions and logic propositions, which allows the modeler to focus on the physical description of the problem rather than on the properties of the model from a mathematical perspective. This is particularly important when dealing with complex systems where a large number of different logic constructs is necessary to describe them and, hence, difficult to model effectively. Exploiting the underlying logic structure of this representation at a higher level of abstraction can help to obtain MINLP models with tighter relaxations and, hence, develop better solution methods [25].

This paper reviews the state of the art of global optimization techniques for non-convex GDPs. It is organized as follows, in Sect. 2 we introduce the general structure of a non-convex GDP and analyze the sources of non-convexity (i.e. arising from nonlinear terms and disjunctions). We review different techniques that have been proposed in the literature to handle them. In particular, we focus on the latest results by Ruiz and Grossmann [26] to find relaxations for convex GDPs. In Sect. 3 we show how the results in previous sections are used to develop a general framework to find convex continuous relaxations for non-convex GDPs as described in [29]. We validate the benefits of this strategy by using it within a set of optimization problems frequently arising in Process Systems Engineering. In Sect. 4 we briefly discuss the implementation of different solution methods that benefit from the use of the relaxation framework described in Sect. 3. Section 5 summarizes the paper and outlines challenges and future lines of work in this area.

2 Non-convex generalized disjunctive programs

The general structure of a non-convex GDP, which we denote as (GDP_{NC}), is as follows,

$$\begin{aligned}
&\min\ Z = f(x) \\
&s.t.\ \ g^l(x) \le 0 \qquad l \in L \\
&\qquad \bigvee_{i\in D_k} \begin{bmatrix} Y_{ik} \\ r^j_{ik}(x) \le 0 \quad j \in J_{ik} \end{bmatrix} \qquad k \in K \\
&\qquad \underline{\bigvee}_{i\in D_k} Y_{ik} \qquad k \in K \\
&\qquad \Omega(Y) = True \\
&\qquad x^{lo} \le x \le x^{up} \\
&\qquad x \in R^n,\ Y_{ik} \in \{True, False\}, i \in D_k, k \in K
\end{aligned} \tag{GDP_{NC}}$$

where $f : R^n \to R^1$ is a function of the continuous variables x in the objective function, $g^l : R^n \to R^1, l \in L$ belongs to the set of global constraints, the disjunctions $k \in K$, are composed of a number of terms $i \in D_k$, that are connected by the OR operator. In each term there is a Boolean variable Y_{ik} and a set of inequalities $r^j_{ik}(x) \leq 0$, $r^j_{ik} : R^n \rightarrow R^1$. If Y_{ik} is True, then $r^j_{ik}(x) \leq 0$ is enforced; otherwise, it is ignored. Note that a fixed cost term was associated to each disjunct in the original representation [22]. However, a more compact form was presented in [12] and it is used here. $\Omega(Y) = True$ are logic propositions for the Boolean variables expressed in the conjunctive normal form $\Omega(Y) = \underset{t=1,2,\ldots T}{\wedge}\left[\underset{(i,k)\in R_t}{\vee}(Y_{ik}) \underset{(i,k)\in Q_t}{\vee}(\neg Y_{ik})\right]$ where for each clause $t = 1, 2 \ldots T$, R_t is the subset of indices of Boolean variables that are non-negated, and Q_t is the subset of indices of Boolean variables that are negated. The logic constraints $\underset{i\in D_k}{\veebar} Y_{ik}$ ensure that only one Boolean variable is True in each disjunction.

It is important to note that the source of non-convexities in GDP_{NC} is twofold. On one hand, the regions that each disjunct defines in the disjunctions may be disconnected in the continuous space. In other words, for any $k \in K$ and $i \in D_k$ and $i' \in D_k$ the intersection of the disjunct i with i' may be empty. On the other hand, the region that is defined in a given disjunct may not be convex (i.e. for any $k \in K$, $i \in D_k$ the set $S = \{ x | r_{ik}(x) \leq 0, x \in R^n \}$ may not be convex). Notice that without loss of generality the global constraints $g^l(x) \leq 0$ can be considered as being part of a disjunction with one disjunct. Also, the nonlinear objective function could also be represented as part of the disjunctive set [29].

Any methodology that aims at finding a convex relaxation for non-convex GDPs must deal with these two sources.

2.1 Relaxation for non-convex regions arising in each disjunct

As proposed by Lee and Grossmann [17] a typical approach to find relaxations for this source of non-convexities consists in replacing the non-convex functions r^j_{ik}, g^l and f with suitable convex underestimators $\hat{r}^j_{ik}$, $\hat{g}^l$ and $\hat{f}$. It is important to note that if this is implemented in the original non-convex GDP (i.e. GDP_{NC}) a convex GDP is obtained. The structure of the resulting convex GDP can be seen in GDP_{CR}.

$$
\begin{aligned}
&\min\ Z = \hat{f}(x) \\
&s.t.\ \ \hat{g}^l(x) \leq 0 \qquad l \in L \\
&\qquad \underset{i\in D_k}{\vee}\begin{bmatrix} Y_{ik} \\ \hat{r}^j_{ik}(x) \leq 0 \quad j \in J_{ik} \end{bmatrix} \qquad k \in K \qquad\qquad (GDP_{CR}) \\
&\qquad \underset{i\in D_k}{\veebar} Y_{ik} \qquad k \in K \\
&\qquad \Omega(Y) = True \\
&\qquad x^{lo} \leq x \leq x^{up} \\
&\qquad x \in R^n,\ Y_{ik} \in \{True, False\}, i \in D_k, k \in K
\end{aligned}
$$

Then, the following relationship can be established,

$$F^{GDP_{CR}} \supseteq F^{GDP_{NC}},$$

where $F^{GDP_{CR}}$ denotes the defining disjunctive set of GDP_{CR} and $F^{GDP_{NC}}$ the defining disjunctive set of GDP_{NC}.

This is illustrated in Fig. 1 where $F^{GDP_{NC}} = \{(x_1, x_2) | [r_1(x) \leq 0] \vee [r_2(x) \leq 0], (x_1, x_2) \in R\}$ and $F^{GDP_{CR}} = \{(x_1, x_2) | [\widetilde{r_1}(x) \leq 0] \vee [\widetilde{r_2}(x) \leq 0], (x_1, x_2) \in R\}$

Considering that $\hat{f}(x)$ always underestimates $f(x)$ the solution of GDP_{CR} provides a lower bound of the global optimal solution of GDP_{NC}.

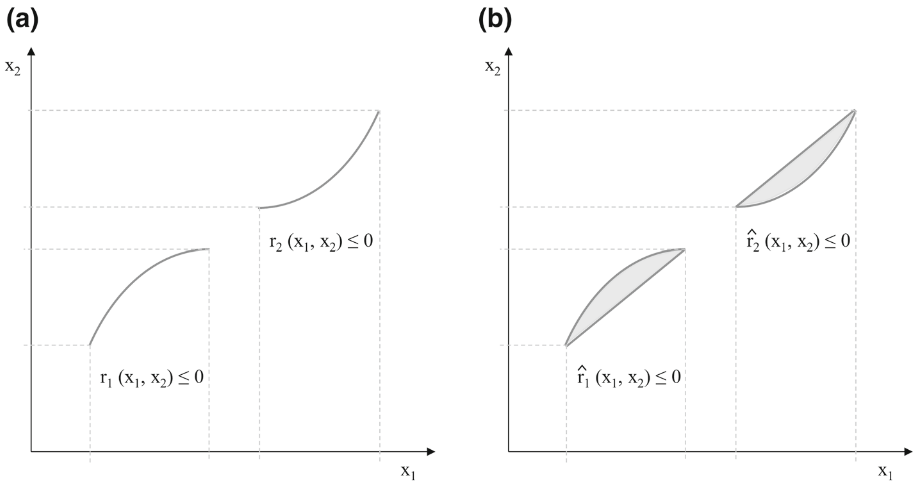

Fig. 1 **a** GDP_{NC}, **b** GDP_{CR}

Finding suitable convex underestimators $\hat{r}^j_{ik}$, $\hat{g}^l$ and $\hat{f}$ has been the purpose of research for many decades. However, most of the current approaches to obtain convex relaxations are based on replacing the non-convex functions with predefined convex envelopes [33]. Three of the most frequent functions that arise in non-convex programs are bilinear, concave and fractional. For the particular case of bilinear terms, tighter relaxations have been proposed. Note that the method to obtain the convex envelope for this special function, which was proposed by McCormick [1,19], is a particular case of the Reformulation-Linearization Technique (RLT) [31] in which cuts are constructed by multiplying constraints by appropriate variables and then linearizing the resulting bilinear terms. Efficient implementations of this approach for large scale problems were studied by Liberti and Pantelides [18]. Other techniques consider the convex envelope of the summation of the bilinear terms [33], the semidefinite relaxation of the whole set of bilinear terms [2] or the piecewise linear relaxations [39]. In [28] a methodology for finding tight convex relaxations for a special set of quadratic constraints given by bilinear and linear terms that frequently arise in the optimization of process networks was presented. The basic idea lies on exploiting the interaction between the vector spaces where the different set of variables are defined in order to generate cuts that will tighten the relaxation of traditional approaches. These cuts are not dominated by the McCormick convex envelopes and can be effectively used in conjunction with them.

In the last few years, techniques to find the convex envelopes for more general non-convex functions have been proposed [14]. Instead of relying on factorable programming techniques to iteratively decompose the non-convex factorable functions through the introduction of variables and then relaxing each intermediate expression, they consider the functions as a whole, leading to stronger relaxations. For a more thorough review on finding relaxations for non-convex functions please see [8].

In the last few years, an alternative way to find relaxations that relies on the physical meaning of the model rather than the mathematical constructs has been introduced [27]. The main idea consists in recognizing that each constraint or set of constraints has a meaning that comes from the physical interpretation of the problem. When these constraints are relaxed part of this meaning is lost. Adding redundant constraints that recover that physical meaning strengthens the relaxation. A methodology to find such redundant constraints based on engineering knowledge and physical insight was proposed.

It is important to note that depending on the strategy that is selected, linear or nonlinear relaxations can be developed. Even though the former is often preferred due to the maturity of linear programming techniques, the tightness of the latter may sometimes result in a significant improvement in the performance of the solution method that is chosen. With this in mind, in this work we consider both linear and nonlinear relaxations.

2.2 Dealing with non-convexities arising from the disjunctions

The discrete nature of the GDP may lead to a disjoint feasible set, even when the regions defined in each disjunct are convex. This leads to a second source of non-convexities for which a relaxation is necessary.

Typically, the continuous relaxation of convex GDPs (i.e. GDPs where the global constraints and the regions defined in each disjunct are convex) is achieved by first reformulating the GDP as a MINLP and then relaxing the integrality of the discrete variables.

GDPs are often reformulated as an MINLP/MIP by using either the big-M (BM) [20], or the Hull reformulation (HR) [16]. The former yields:

$$
\begin{aligned}
\min\ & Z = f(x) \\
s.t.\ & g(x) \leq 0 \\
& r_{ik}(x) \leq M(1 - y_{ik}) \qquad i \in D_k,\ k \in K \qquad (BM) \\
& \textstyle\sum_{i \in D_k} y_{ik} = 1 \qquad k \in K \\
& Ay \geq a \\
& x^{lo} \leq x \leq x^{up} \\
& x \in R^n,\ y_{ik} \in \{0, 1\},\quad i \in D_k,\ k \in K
\end{aligned}
$$

where the variable y_{ik} has a one to one correspondence with the Boolean variable Y_{ik}. Note that when $y_{ik} = 0$ and the parameter M is sufficiently large, the associated constraint becomes redundant; otherwise, it is enforced. Also, $Ay \geq a$ is the reformulation of the logic constraints in the discrete space, which can be easily implemented as described in the work by Williams [40] and discussed in the work by Raman and Grossmann [22]. The HR reformulation yields,

$$
\begin{aligned}
\min\ & Z = f(x) \\
s.t.\ & x = \textstyle\sum_{i \in D_K} \nu^{ik} \qquad k \in K \\
& g(x) \leq 0 \\
& y_{ik} r_{ik}(\nu^{ik}/y_{ik}) \leq 0 \qquad i \in D_k,\ k \in K \qquad (HR) \\
& y_{ik} x^{lo} \leq \nu^{ik} \leq y_{ik} x^{up} \qquad i \in D_k,\ k \in K \\
& \textstyle\sum_{i \in D_k} y_{ik} = 1 \qquad k \in K \\
& Ay \geq a \\
& x \in R^n,\ \nu^{ik} \in R^n,\ y_{ik} \in \{0, 1\},\ i \in D_k,\ k \in K
\end{aligned}
$$

As it can be seen, the HR reformulation is not as intuitive as the BM. However, there is also a one to one correspondence between (GDP) and (HR). Note that the size of the problem is increased by introducing a new set of disaggregated variables ν^{ik} and new constraints. On the other hand, as proved in Grossmann and Lee [11] and discussed by Vecchietti et al. [37], the HR formulation is at least as tight and generally tighter than the BM when the discrete domain is relaxed (i.e. $0 \leq y_{ik} \leq 1,\ k \in K,\ i \in D_k$). This is of great importance considering that the efficiency of the MINLP/MIP solvers heavily rely on the quality of these relaxations [10].

It is important to note that on the one hand the $y_{ik}\ r_{ik}\ (\nu^{ik}/y_{ik})$ is convex if $r_{ik}(x)$ is a convex function. On the other hand, if $r_{ik}(x)$ is nonlinear, the term requires the use of a suitable approximation to avoid singularities. Sawaya [30] proposed the following reformulation which yields an exact approximation at $y_{ik} = 0$ and $y_{ik} = 1$ for *any* value of ε in the interval (0,1), and the feasibility and convexity of the approximating problem are maintained:

$$y_{ik}r_{ik}(\nu^{ik}/y_{ik}) \approx \ ((1-\varepsilon)y_{ik}+\varepsilon)r_{ik}(\nu^{ik}/((1-\varepsilon)y_{ik}+\varepsilon)) - \varepsilon r_{ik}(0)(1-y_{ik})$$

Note that this approximation assumes that $r_{ik}(x)$ is defined at $x = 0$ and that the inequality $y_{ik}x^{lo} \le \nu^{ik} \le y_{ik}x^{up}$ is enforced. Clearly, if $r_{ik}(x)$ is linear (i.e. $r_{ik}(x) = A_{ik}x - b_{ik}$, with A_{ik} and b_{ik} a real matrix and vector, respectively), $y_{ik}r_{ik}(\nu^{ik}/y_{ik})$ does not need to be approximated. In this case $y_{ik}r_{ik}(\nu^{ik}/y_{ik}) = A_{ik}x - b_{ik}y_{ik}$.

One question that arises is whether the HR relaxation can be improved. This question was tackled by Sawaya and Grossmann [30] for the linear case and Ruiz and Grossmann [26] for the nonlinear case. In this paper we review the nonlinear case since it can be regarded as a generalization of the linear problem.

In [26] it was proved that any nonlinear convex GDP that involves Boolean and continuous variables can be equivalently formulated as a Disjunctive Convex Program (DCP) that only involves continuous variables. This transformation, which is equivalent to the one proposed by Sawaya and Grossmann [30] for linear GDP, consists in first replacing the Boolean variables $Y_{ik}, i \in D_k, k \in K$ inside the disjunctions by equalities $\lambda_{ik} = 1, i \in D_k, k \in K$, where λ is a vector of continuous variables whose domain is [0,1], and convert the logical relations $\underset{i \in D_k}{\vee}\ Y_{ik}$ and $\Omega(Y) = True$ into the algebraic equations $\sum_{i \in D_k} \lambda_{ik} = 1, k \in K$ and $A\lambda \ge a$, respectively. This yields the following disjunctive model:

$$\begin{aligned}
&\min\ Z = f(x) \\
&s.t.\ \ g(x) \le 0 \\
&\qquad \underset{i \in D_k}{\vee} \begin{bmatrix} \lambda_{ik} = 1 \\ r_{ik}(x) \le 0 \end{bmatrix} \quad k \in K \qquad\qquad (DCP) \\
&\qquad \textstyle\sum_{i \in D_k} \lambda_{ik} = 1 \quad k \in K \\
&\qquad A\lambda \ge a \\
&\qquad x^{lo} \le x \le x^{up} \\
&\qquad x \in R^n,\ \lambda_{ik} \in [0,1]
\end{aligned}$$

This equivalent representation means that the theory behind disjunctive convex programming [4,6] can be exploited to find relaxations for convex GDP.

One of the properties of disjunctive sets is that they can be expressed in many different equivalent forms. Among these forms, two extreme ones are the Conjunctive Normal Form (CNF), which is expressed as the intersection of elementary sets, and the Disjunctive Normal Form (DNF), which is expressed as the union of convex sets. One important result in disjunctive convex programming theory, as presented in [4,26], is that a set of equivalent disjunctive convex programs going from the CNF to the DNF can be systematically generated by performing an operation called "basic step" that preserves regularity. A Regular Form (RF) is defined as the form represented by the intersection of the union of convex sets. Hence, the regular form is:

$$F = \bigcap_{k \in K} S_k$$

where for $k \in K$, $S_k = \bigcup_{i \in D_k} P_i$ and P_i a convex set for $i \in D_k$.

The following theorem, as first stated in [4], defines a "basic step" as an operation that takes a disjunctive set to an equivalent disjunctive set with fewer conjuncts.

Theorem 1 *Let F be a disjunctive set in regular form. Then F can be brought to DNF by $|K| - 1$ recursive applications of the following basic step which preserves regularity:*
For some $r, s \in K$, bring $S_r \cap S_s$ to DNF by replacing it with:

$$S_{rs} = \bigcup_{i \in D_r, j \in D_s} (P_i \cap P_j)$$

Although the formulations obtained after the application of basic steps on the disjunctive sets are equivalent, their continuous relaxations are not. We denote the continuous relaxation of a disjunctive set $F = \bigcap_{j \in T} S_j$ in regular form, where each S_j is a union of convex sets, as the *hull-relaxation* of F (or $h - rel\ F$). Here $h - rel\ F := \bigcap_{j \in T} clconv\ S_j$ and $clconv\ S_j$ denotes the closure of the convex hull of S_j. That is, if $S_j = \bigcup_{i \in Q_j} P_i$, $P_i = \{x \in R^n, r_i(x) \leq 0\}$, then the $clconv S_j$ is given by:

$$\begin{aligned} & x = \sum_{i \in Q_j} \nu^i \\ & \lambda_i r_i(\nu^i/\lambda_i) \leq 0, \qquad i \in Q_j \\ & \sum_{i \in Q_j} \lambda_i = 1,\ \ \lambda_i \geq 0,\ i \in Q_j \qquad\qquad (DISJ_{rel}) \\ & |\nu^i| \leq L\lambda_i \qquad\qquad i \in Q_j \end{aligned}$$

where ν^i are disaggregated variables, λ_i are continuous variables between 0 and 1 and $\lambda_i r_i(\nu^i/\lambda_i)$ is the perspective function that is convex in ν and λ if the function $r(x)$ is also convex [32].

As shown in Theorem 2 [4,26] and illustrated in Fig. 2, the application of a basic step leads to a new disjunctive set whose hull relaxation is at least as tight, if not tighter, than the original one.

Theorem 2 *For $i = 1, 2....k$, let $F_i = \bigcap_{k \in K} S_k$ be a sequence of regular forms of a disjunctive set such that F_i is obtained from F_{i-1} by the application of a basic step, then:*

$$h\text{-}rel(F_i) \subseteq h\text{-}rel(F_{i-1})$$

It is important to note that every time a basic step is applied, the number of disjuncts generally increases, leading in principle to the need of a larger number of binary variables to represent them in the mixed-integer formulation. Based on the work on disjunctive linear programming [4], the following theorem establishes that no increase of the number of 0–1 variables is required [26].

Theorem 3 *Let $Z = \min\{f(x)|x \in F_d\}$ be a disjunctive convex program with the variables x bounded below and above by a large number L and such that F_d is a disjunctive set in regular form consisting of those $x \in R^n$ satisfying $\bigvee_{s \in Q_r} (r_s(x) \leq 0), r \in T_d$ and let F_n the disjunctive set obtained after the application of a number of basic steps on F_d , such that $x \in R^n$ satisfies $\bigvee_{t \in Q_j} (G^t(x) \leq 0), j \in T_n$. Then every $j \in T_n$ corresponds to a subset T_{dj} with $T_d = \bigcup_{j \in T_n} T_{dj}$ such that the disjunction in $\bigvee_{t \in Q_j} (G^t(x) \leq 0)$ for a given j is the*

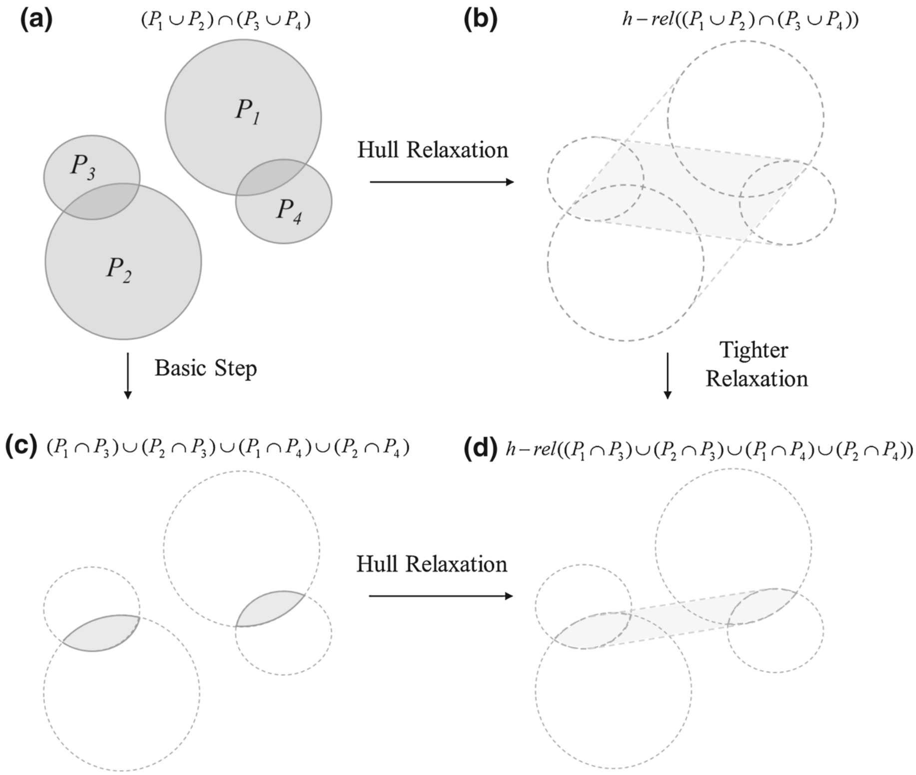

Fig. 2 Impact of the basic steps on the relaxation of an illustrative disjunctive set

disjunctive normal form of the set of disjunctions $\bigvee_{s \in Q_r} (r_s(x) \le 0), r \in T_{dj}$. *Furthermore, let* M_j^t *be the index set of the inequalities* $r_s(x) \le 0$ *making up the system* $G^t(x) \le 0$ *for a given* $j \in T_n$ *and* $t \in Q_j$. *Then, an equivalent mixed-integer nonlinear program can be described as:*

$$
\begin{array}{llr}
\min \; Z = f(x) & & \\
s.t. & & \\
\quad x = \sum_{t \in Q_j} \nu^t, & j \in T_n & \\
\quad \lambda_i G^t(\nu^t/\lambda_t) \le 0, & t \in Q_j,\ j \in T_n & \\
\quad \sum_{t \in Q_j} \lambda_t = 1, & t \in Q_j,\ j \in T_n & \\
\quad \lambda_t \ge 0, & t \in Q_j,\ j \in T_n & (HR_{CGDP}) \\
\quad \sum_{t \in Q_j | s \in M_j^t} \lambda_t = \delta_s^r, & s \in Q_r,\ r \in T_d,\ j \in T_n & \\
\quad \sum_{s \in Q_r} \delta_s^r = 1, & r \in T_d & \\
\quad |\nu^t| \le L\lambda_t, & t \in Q_j,\ j \in T_n & \\
\quad \delta_s^r \in \{0, 1\}, & t \in Q_j,\ j \in T_n &
\end{array}
$$

Even though the number of discrete variables does not increase, the number of constraints and continuous variables may increase. This is why it is important to apply the basic steps

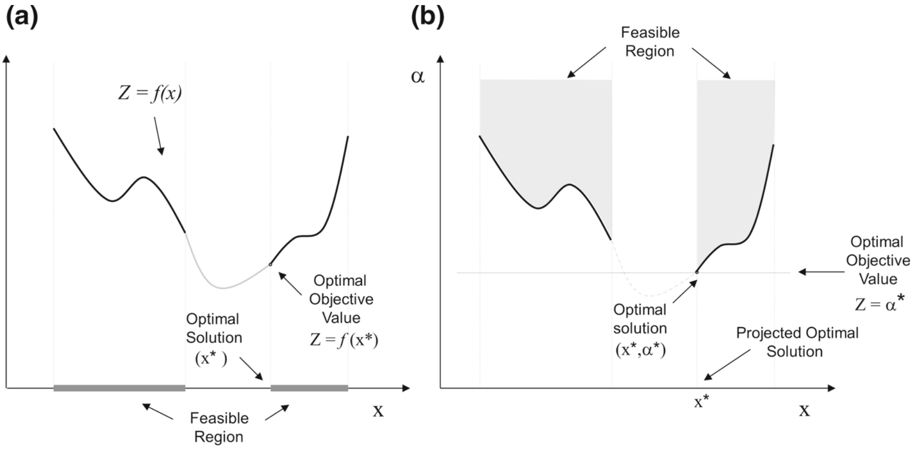

Fig. 3 Equivalence between **a** GDP_{NC} and **b** $GDP_{NC'}$

in an efficient way. Ruiz and Grossmann [24] developed a set of propositions that led to the development of rules to apply the basic steps. These are summarized as follows:
Rule 1 Apply basic steps between those disjunctions with at least one variable in common.
Rule 2 The more variables in common two disjunctions have, the more the tightening can be expected.
Rule 3 A basic step between a half space and a disjunctions with two disjuncts one of which is a point contained in the facet of the half space will not tighten the relaxation.
Rule 4 A smaller increase in the size of the formulation is expected when basic steps are applied between improper disjunctions and proper disjunctions.

A new rule developed in [26] consists in the inclusion of the objective function in the disjunctive set previous the application of basic steps. This has shown to be useful to strengthen the final relaxation of the disjunctive set. Note that this rule, differently from the previous ones, has an effect when the objective function is nonlinear. In the work by Ruiz and Grossmann [29] it is shown that this relaxation is still valid for the non-convex case. This is true considering that when the objective function of a non-convex GDP (GDP_{NC}) is represented as a constraint, it leads to an equivalent GDP ($GDP_{NC'}$). This is illustrated in Fig. 3.

An efficient and more systematic implementation of these rules is described in the work of Trespalacios and Grossmann [34].

3 Convex continuous relaxations of non-convex GDPs

Now we are ready to present one of the main results in the theory of non-convex GDPs, which is instrumental in the development of a relaxation framework. Namely, a hierarchy of relaxations for GDP_{NC}. Let us assume GDP_{CR0} is obtained by replacing the non-convex functions with suitable relaxations as presented in Sect. 2. Also, let us assume that GDP_{CRi} is the convex generalized disjunctive program whose defining disjunctive set is obtained after applying i basic steps on the disjunctive set of GDP_{CR0} and t is the number of basic steps required to achieve the DNF. Note that $i \leq t$. Then, from Theorem 2 and the main result in Sect. 2.1 the following relationship can be established,

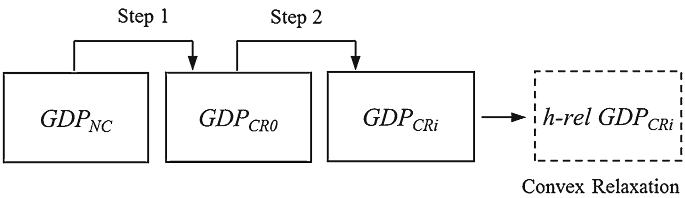

Fig. 4 Framework to obtain relaxations for GDP_{NC}

$$h\text{-}rel(F_0^{GDP_{CR}}) \supseteq h\text{-}rel(F_1^{GDP_{CR}}) \cdots \supseteq h\text{-}rel(F_i^{GDP_{CR}}) \cdots \supseteq \\ \cdots \supseteq h\text{-}rel(F_t^{GDP_{CR}}) \supseteq F_t^{GDP_{CR}} \sim F_0^{GDP_{CR}} \supseteq F^{GDP_{NC}},$$

where $F_i^{GDP_{CR}}$ denotes the defining disjunctive set of GDP_{CRi} and $F^{GDP_{NC}}$ the defining disjunctive set of GDP_{NC}. Also, the symbol $\sim$ denotes equivalence.

Based on the previous results, Fig. 4 shows a schematic of the general framework for finding strong relaxations for non-convex GDP. The framework consists of two steps. In Step 1 the non-convex GDP (GDP_{NC}) is relaxed as a convex GDP (GDP_{CR0}). In Step 2 of the framework the convex GDP is reformulated as an equivalent convex GDP (GDP_{CRi}) by using the basic steps. The hull-relaxation of the latter is a valid strong relaxation for the initial non-convex GDP and can be used to obtain tight lower bounds within any solution method.

Note that the hull relaxation of the GDP that is obtained after Step 1 (i.e. GDP_{CR0}) was initially proposed by Lee and Grossmann [17] as a valid relaxation for non-convex GDPs. This was later extended by Ruiz and Grossmann [29] by adding Step 2. In the following sections we will show the impact of the second step on the tightness of the relaxation.

3.1 Computational results for the relaxation framework

In this section we show through a set of numerical examples the benefits of the two step approach to find relaxations for non-convex GDPs with focus on describing the impact of the second step on the quality of the relaxations. In the first set of examples, the first step leads to a linear GDP, which in turn leads to a linear relaxation, whereas in the second set of examples the first step leads to a nonlinear GDP which in turn leads to a nonlinear relaxation. As it will be shown, the benefits of this approach to find strong relaxations is invariant to the linear properties of the system.

3.1.1 Linear relaxations

The first set of numerical examples consists of six problems that frequently arise in Process Systems [24] for which linear relaxations are proposed. The problems $Ex1_{Lin}$ and $Ex2_{Lin}$ deal with the optimal design and selection of a reactor. $Ex3_{Lin}$ and $Ex6_{Lin}$ are related to the optimization of a Heat Exchanger Network with discontinuous investment costs for the exchangers and can be represented by a non-convex GDP with bilinear and concave constraints [36]. $Ex4_{Lin}$ deals with the optimization of a Wastewater Treatment Network whose associated non-convex GDP formulation is a bilinear GDP [9]. Finally, $Ex5_{Lin}$ is a Pooling Design problem that can also be represented as a bilinear GDP [17].

Table 1 summarizes the characteristics and size of the examples, and Table 2 shows the lower bounds predicted by using only the first step [17] and using the first and second step.

Table 1 Size and characteristics of the example problems

	Boolean variables	Continuous variables	Bilinear terms	Concave terms
$Ex1_{Lin}$	2	3	1	0
$Ex2_{Lin}$	2	5	0	2
$Ex3_{Lin}$	9	8	4	9
$Ex4_{Lin}$	9	114	36	0
$Ex5_{Lin}$	9	76	24	0
$Ex6_{Lin}$	24	24	11	24

Table 2 Lower bounds of proposed framework

	Global optimum	Lower bound (Step I)	Lower bound (Step I + Step II)	Best lower bound DNF
$Ex1_{Lin}$	−1.01	−1.28	−1.10	−1.10
$Ex2_{Lin}$	5.56	4.90	5.33	5.33
$Ex3_{Lin}$	114, 384.78	91, 671.18	94, 925.77	97, 858.86
$Ex4_{Lin}$	1214.87	400.66	431.90	431.90
$Ex5_{Lin}$	−4640.00	−5515.00	−5468.00	−5241.00
$Ex6_{Lin}$	322, 122.09	260, 235.11	265, 361.46	281, 191.44

All the examples that were solved show an improvement in the lower bound prediction when the second step is used. For instance, in $Ex5_{Lin}$ it increased from -5515 to -5468 which is a direct indication of the reduction of the relaxed feasible region. The column "Best Lower Bound" represents the lower bound that would be obtained by the method if the second step takes the GDP to the DNF form. With this in mind, it can be used as an indicator of the performance of the proposed set of rules to apply basic steps. Note that in the $Ex1_{Lin}$, $Ex2_{Lin}$ and $Ex4_{Lin}$, the lower bound obtained using the two step approach is the same as the one obtained by solving the relaxed DNF, which is quite remarkable. A further indication of tightening is shown in Sect. 4 where numerical results of the solution methods are presented.

3.1.2 Nonlinear relaxations

The second set of numerical examples considers optimization problems that, as in the first set, are frequently found in Process Systems Engineering for which nonlinear relaxations are proposed [29]. $Ex1_{NonLin}$ and $Ex2_{NonLin}$ considers the optimization of a process network with fixed charges [15]. The problems are non-convex where the non-convexities arise from the nonlinear inequalities (given by exponential functions) defining the processes and from the disjunctive nature of the problem. $Ex3_{NonLin}$ and $Ex4_{NonLin}$ consider the optimization of a reactor networks with non-elementary kinetics described through posynomial functions. Finally, $Ex5_{NonLin}$ and $Ex6_{NonLin}$ consider the optimization of a heat exchanger network model with linear fractional terms.

In Table 3 we show the size and characteristics of the second set of instances and Table 4 shows the lower bounds predicted by using only the first step [17] and using the first and second step.

Table 3 Size and characteristics of the example problems

Example	Cont. vars.	Boolean vars.	Logic const.	Disj. const.	Global const.
$Ex1_{NonLin}$	5	2	1	1	3
$Ex2_{NonLin}$	5	2	1	1	3
$Ex3_{NonLin}$	4	2	1	1	6
$Ex4_{NonLin}$	4	2	1	1	6
$Ex5_{NonLin}$	18	2	2	2	21
$Ex6_{NonLin}$	18	2	2	2	21

Table 4 Lower bounds of proposed framework

	Global optimum	Lower bound (Step I)	Lower bound (Step I + Step II)	Best lower bound DNF
$Ex1_{NonLin}$	18.61	11.85	16.01	16.01
$Ex2_{NonLin}$	19.48	12.38	17.07	17.07
$Ex3_{NonLin}$	42.89	−337.50	−320.00	−320.00
$Ex4_{NonLin}$	76.47	22.50	40.00	40.00
$Ex5_{NonLin}$	48, 531.00	38, 729.27	48, 230.00	48, 531.00
$Ex6_{NonLin}$	45, 460.00	35, 460.00	45, 281.00	45, 281.00

Clearly, from Table 4 we observe a significant improvement in the predicted lower bound in all instances. For example, in $Ex2_{NonLin}$ the two step framework predicts 17.07 as a lower bound, whereas the approach based on using only the first step is only able to obtain a bound of 12.38. Moreover, the lower bounds obtained are close, if not the same as the one we would obtain if the relaxation of the DNF form is solved. For example, $Ex6_{NonLin}$, reaches a lower bound of 45,281, which is the same as the maximum attainable.

4 Logic-based global optimization algorithms for non-convex GDPs with improved relaxations

Several methods to solve non-convex GDPs to global optimality have been proposed [23]. However, we can organize them into two well defined groups. Namely, logic-based branch and bound methods [17,24] and logic-based outer-approximation methods [5].

The main steps in the implementation of the first group are described in Fig. 5. The algorithm starts by obtaining a local solution of the non-convex GDP problem by solving a MINLP reformulation with an optimizer that assumes convexity (e.g. DICOPT [38]), which provides an upper bound of the solution (Z^U). Then, a bound contraction procedure is performed as described by Zamora and Grossmann [41]. Finally, a partial branch and bound method is used on a relaxation of the non-convex GDP as described in Lee and Grossmann [17] that consists in only branching on the Boolean variables until a node with all the Boolean variables fixed (Y) is reached. At this point a spatial branch and bound procedure is performed as described in Quesada and Grossmann [13,21].

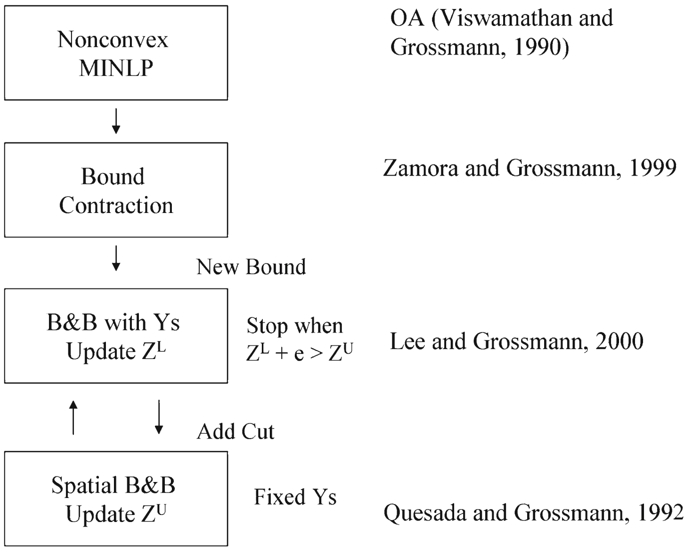

Fig. 5 Steps in logic-based branch and bound methods

It is important to note that the effectiveness of the bound contraction procedure as well as the spatial branch and bound step rely on the relaxation that is used. This is illustrated in Tables 5 and 6 where the performance of the method is tested on the set of instances described in Sects. 3.1.1 and 3.1.2. A clear indication of a tighter relaxation when using Step I in conjunction with Step II is observed in the columns *Bounding* and *Nodes*. The latter refers to the number of nodes the algorithm needs to visit to find the solution whereas the former refers to how much reduction in the upper and lower bounds of the variables can be predicted. More precisely, the column *Bounding %* refers to the upper/lower bound of the variable x_i before and after the bound contraction procedure, respectively. For instance, in $Ex1_{NonLin}$, the two step framework is able to reduce the bounds of the variables 35 % with respect to the original bounds, whereas by using only Step I the bounds are only contracted 13.8 %. Note that the strength of the relaxations of the non-convex functions heavily depend on the bounds of the variables on which they are defined, and that is why it is very important to count on an efficient procedure to find these bounds. Even though a modest reduction in the number of nodes required to find the solution when using nonlinear relaxations may be due to the fact that the problems are small in size, a significant reduction is observed in the instances that used linear relaxations. For example in the instance $Ex4_{Lin}$ only 130 nodes were needed when using the proposed framework as opposed to 408 nodes when only using the first step. Furthermore, the reduction in the number of nodes leads to a reduction in the time necessary to find the solution. For example $Ex4_{Lin}$ only requires 115 s when using the proposed relaxation as opposed to 176 s when using only the first step of the approach.

Motivated by the benefits of using the logic-based outer-approximation approach to solve convex GDP [36], an alternative group of methods was proposed [5]. The main idea is to solve iteratively reduced NLP subproblems to global optimality to obtain upper bounds of the global optimum and MILP master problems, which are valid outer-approximations of the original problem, to obtain lower bounds. Trespalacios and Grossmann [35] proposed an improvement of this method by generating cuts that strengthen the relaxation of the non-convex terms in the disjuncts. It is important to note that the quality of the lower bound

Table 5 Performance of the relaxations within a spatial B&B

	GO	Step I			Step I + Step II		
		Nds	Bounding %	Time (s)	Nds	Bounding %	Time (s)
$Ex1_{Lin}$	−1.01	5	35	2.1	1	38	1.4
$Ex2_{Lin}$	5.56	1	33	1.0	1	33	1.0
$Ex3_{Lin}$	114, 384.78	10	85	9.0	1	99	5.0
$Ex4_{Lin}$	1, 214.87	408	8	176	130	16	115
$Ex5_{Lin}$	−4, 640.00	162	1	89	140	1	93
$Ex6_{Lin}$	322, 122.09	18	98	24	5	99	18

Table 6 Performance of the relaxations within a spatial B&B

	GO	Step I			Step I + Step II		
		Nds	Bounding %	Time (s)	Nds	Bounding %	Time (s)
$Ex1_{NonLin}$	18.61	3	51.3	6	2	67.0	4
$Ex2_{NonLin}$	19.48	2	40.5	4	2	47.2	4
$Ex3_{NonLin}$	42.89	2	51.0	7	2	66.0	7
$Ex4_{NonLin}$	76.46	2	51.0	6	2	66.0	6
$Ex5_{NonLin}$	48, 531.00	3	13.8	15	1	35.0	14
$Ex6_{NonLin}$	45, 460.00	3	7.5	14	1	97	23

inferred by the MILP master problem heavily relies on the strength of the GDP relaxation and that is why the results presented in this paper will probably also have a great impact on the performance of the method.

5 Conclusions

In this paper we have presented a review on the state-of-the-art of solution methods for the global optimization of GDPs. Considering that the performance of these methods relies on the quality of the relaxations that can be generated, our focus has been on the discussion of a general framework to find strong relaxations. We identified two main sources of non-convexities that any methodology to find relaxations should account for. Firstly, the one arising from the non-convex functions, and secondly, the one arising from the disjunctive set. Research on both fronts have had and will have a great impact on the quality of these relaxations. We have described the use of these relaxations within logic-based global optimization methods. Even though there is a clear benefit of using stronger relaxations, one key challenge that remains to be solved is how to generate these relaxations without incurring in a significant increase in the size of the reformulation. With this in mind, new rules and techniques to apply basic steps or novel cutting plane strategies need to be developed. We have achieved significant progress in understanding the theory behind the generation of these relaxations. However, implementing them efficiently to fully exploit their potential is still an open problem.

Acknowledgments The authors would like to acknowledge financial support from the National Science Foundation under Grant 601 OCI-0750826.

References

1. Al-Khayyal, F.A., Falk, J.E.: Jointly constrained biconvex programming. Math. Oper. Res. **8**(2), 273–286 (1983)
2. Anstreicher, K.M.: Semidefinite programming versus the reformulation-linearization technique for nonconvex quadratically constrained quadratic programming. J. Glob. Optim. **43**(2–3), 471–484 (2009)
3. Balas, E.: Disjunctive Programming. Ann. Discret. Math. **5**, 3–51 (1979)
4. Balas, E.: Disjunctive programming and a hierarchy of relaxations for discrete optimization problems. SIAM J. Algebr. Discrete Methods **6**, 466–486 (1985)
5. Bergamini, M.L., Aguirre, P.A., Grossmann, I.E.: Logic-based outer approximation for globally optimal synthesis of process networks. Comput. Chem. Eng. **29**(9), 1914–1933 (2005)
6. Ceria, S., Soares, J.: Convex programming for disjunctive convex optimization. Math. Program. **86**, 595–614 (1999)
7. Floudas, C.A.: Deterministic Global Optimization: Theory Methods and Applications. Kluwer, Dordrecht (2000)
8. Floudas, C.A., Gounaris, C.E.: A review of recent advances in global optimization. J. Glob. Optim. **45**(1), 3–38 (2009)
9. Galan, B., Grossmann, I.E.: Optimal design of distributed wastewater treatment networks. Ind. Eng. Chem. Res. **37**, 4036–4048 (1998)
10. Grossmann, I.E.: Review of non-linear mixed integer and disjunctive programming techiques for process systems engineering. Optim. Eng. **3**, 227–252 (2002)
11. Grossmann, I.E., Lee, S.: Generalized convex disjunctive programming: nonlinear convex hull relaxation. Comput. Optim. Appl. **26**, 83–100 (2003)
12. Grossmann, I.E., Trespalacios, F.: Systematic modeling of discrete-continuous optimization models through generalized disjunctive programming. AIChE J. **59**, 3276–3295 (2013)
13. Horst, R., Tuy, H.: Global Optimization Deterministic Approaches, 3rd edn. Springer, Berlin (1996)
14. Khajavirad, A., Sahinidis, N.V.: Convex envelopes generated from finitely many compact convex sets. Math. Program. **137**(1), 371–408 (2013)
15. Kocis, G.R., Grossmann, I.E.: Relaxation strategy for the structural optimization of process flow sheets. Ind. Eng. Chem. Res. **26**, 1869 (1987)
16. Lee, S., Grossmann, I.E.: New algorithms for nonlinear generalized disjunctive programming. Comput. Chem. Eng. **24**, 2125–2141 (2000)
17. Lee, S., Grossmann, I.E.: Global optimization of nonlinear generalized disjunctive programming with bilinear inequality constraints: application to process networks. Comput. Chem. Eng. **27**, 1557–1575 (2003)
18. Liberti, L., Pantelides, C.C.: An exact reformulation algorithm for large non-convex NLPs involving bilinear terms. J. Glob. Optim. **36**(2), 161–189 (2006)
19. McCormick, G.P.: Computability of global solutions to factorable nonconvex programs. Part I. Convex underestimating problems. Math. Program. **10**, 146–175 (1976)
20. Nemhauser, G.L., Wolsey, L.A.: Integer and Combinatorial Optimization. Wiley-Interscience, New York (1988)
21. Quesada, I., Grossmann, I.E.: An LP/NLP based branch and bound algorithm for convex MINLP optimization problems. Comput. Chem. Eng. **16**, 937–947 (1992)
22. Raman, R., Grossmann, I.E.: Modelling and computational techniques for logic-based integer programming. Comput. Chem. Eng. **18**, 563 (1994)
23. Ruiz, J.P.: Thesis: Global optimization of nonconvex Generalized Disjunctive Programs. Carnegie Mellon University (2011)
24. Ruiz, J.P., Grossmann, I.E.: Strengthening the lower bounds for bilinear and concave GDP problems. Comput. Chem. Eng. **34**(3), 914–930 (2010)
25. Ruiz, J.P., Grossmann, I.E.: Generalized disjunctive programming: a framework for formulation and alternative algorithms for MINLP optimization. IMA Vol. Math. Its Appl. **218**, 93–115 (2012)
26. Ruiz, J.P., Grossmann, I.E.: A hierarchy of relaxations for nonlinear convex generalized disjunctive programming. Eur. J. Oper. Res. **218**(1), 38–47 (2012)
27. Ruiz, J.P., Grossmann, I.E.: Using redundancy to strengthen the relaxation for the global optimization of MINLP. Comput. Chem. Eng. **35**(12), 2729–2740 (2011)

28. Ruiz, J.P., Grossmann, I.E.: Exploiting vector space properties to strengthen the relaxation of bilinear programs arising in the global optimization of process networks. Optim. Lett. **5**, 1–11 (2012)
29. Ruiz, J.P., Grossmann, I.E.: Using convex nonlinear relaxations in the global optimization of nonconvex generalized disjunctive programs. Comput. Chem. Eng. **49**, 70–84 (2013)
30. Sawaya, N.: Thesis: Reformulations, relaxations and cutting planes for generalized disjunctive programming. Carnegie Mellon University (2006)
31. Sherali, H.D., Alameddine, A.: A new reformulation linearization technique for bilinear programming problems. J. Glob. Optim. **2**, 379–410 (1992)
32. Stubbs, R., Mehrotra, S.: A branch-and-cut method for 0–1 mixed convex programming. Math. Program. **86**(3), 515–532 (1999)
33. Tawarmalani, M., Sahinidis, N.: Convexification and Global Optimization in Continuous and Mixed-Integer Nonlinear Programming. Kluwer, Dordrecht (2002)
34. Trespalacios, F., Grossmann, I.E.: Algorithmic approach for improved mixed-integer reformulations of convex generalized disjunctive programs. INFORMS J. Comput. **27**(1), 59–74 (2014)
35. Trespalacios, F., Grossmann, I.E.: Cutting planes for improved global logic-based outer approximation for the synthesis of process networks. Comput. Chem. Eng., Submitted for publication (2015)
36. Turkay, M., Grossmann, I.E.: A logic-based outer-approximation algorithm for MINLP optimization of process flowsheets. Comput. Chem. Eng. **20**, 959–978 (1996)
37. Vecchietti, A., Lee, S., Grossmann, I.E.: Modeling of discrete/continuous optimization problems: characterization and formulation of disjunctions and their relaxations. Comput. Chem. Eng. **27**, 433–448 (2003)
38. Viswanathan, J., Grossmann, I.E.: A combined penalty function and outer-approximation method for MINLP optimization. Comput. Chem. Eng. **14**, 769–782 (1990)
39. Wicaksono, D.S., Karimi, I.A.: Piecewise MILP under- and overestimators for global optimization of bilinear programs. AICHE J. **54**, 991–1008 (2008)
40. Williams, H.P.: Mathematical Building in Mathematical Programming. Wiley, New York (1985)
41. Zamora, J.M., Grossmann, I.E.: A branch and bound algorithm for problems with concave univariate, bilinear and linear fractional terms. J. Glob. Optim. **14**(3), 217–249 (1999)