# A hierarchy of relaxations for nonlinear convex generalized disjunctive programming

Juan P. Ruiz, Ignacio E. Grossmann [1]

*Carnegie Mellon University - Department of Chemical Engineering*

*Pittsburgh, PA 15213*

**Abstract**

We propose a framework to generate alternative mixed-integer nonlinear programming formulations for disjunctive convex programs that lead to stronger relaxations. We extend the concept of "basic steps" defined for disjunctive linear programs to the nonlinear case. A basic step is an operation that takes a disjunctive set to another with fewer number of conjuncts. We show that the strength of the relaxations increases as the number of conjuncts decreases, leading to a hierarchy of relaxations. We prove that the tightest of these relaxations, allows in theory the solution of the disjunctive convex program as a nonlinear programming problem. We present a methodology to guide the generation of strong relaxations without incurring an exponential increase of the size of the reformulated mixed-integer program. Finally, we apply the theory developed to improve the computational efficiency of solution methods for nonlinear convex generalized disjunctive programs (GDP). This methodology is validated through a set of numerical examples.

*Keywords:*
Combinatorial Optimization, Convex Programming, Disjunctive Programming, Generalized Disjunctive Programming, Tight Relaxations

## 1. Introduction

Disjunctive convex programming deals with the optimization over the intersection of the union of closed convex sets (Ceria and Soares, 1999; Stubbs and Mehrotra, 1999). These problems arise frequently in many different areas, such as in process systems design (Duran and Grossmann, 1986; Turkay and Grossmann, 1996), scheduling of operations and layout problems (Sawaya, 2006). Even though formulations

[1] Corresponding Author. Tel.: +1-412-268-7139; e-mail: grossmann@cmu.edu (I.E. Grossmann)

that deal with the optimization over the intersection of the union of polyhedra were orginally proposed as an extension of the early work by Balas on disjunctive linear programming (see Balas (1979)), the solution methods have only borrowed a small portion of its wealth. As in mixed-integer programming, the efficiency of solution methods in disjunctive programming heavily depends on the way one formulates the problem, which in turn depends on the relaxations these formulations produce. For instance, Sherali and Adams proposed a method to generate formulations that lead to a hierarchy of convex relaxations for general convex mixed-integer programs (see Sherali and Adams (2009). Similarly, Balas showed that alternative formulations of disjunctive linear programs can be produced systematically leading to tighter relaxations (see Balas (1985)).

Generalized disjunctive programming (GDP) (Raman and Grossmann, 1994), is a generalization of disjunctive convex programming in the sense that it also allows the use of logic popositions that are expressed in terms of Boolean variables. In order to take advantage of current mixed-integer nonlinear programming solvers (e.g. DICOPT (Viswanathan and Grossmann, 1990), SBB (Brooke et al., 1998), $\alpha$-ECP (Westerlund and Pettersson, 1995), Bonmin (Bonami et al., 2008), FilMINT (Abhishek et al., 2006), BARON (Sahinidis, 1996), etc.), GDPs are often reformulated as MINLPs. The intuitive way to reformulate a GDP is by using the Big-M (BM) reformulation (Nemhauser and Wolsey, 1988). However, Lee and Grossmann showed that a less intuitive approach, given by the Hull Reformulation (HR), leads to tighter relaxations, and generally, significant improvements in the solution times (see Lee and Grossmann (2000)).

In this work we will extend the theory of disjunctive linear programming to generate better alternative formulations for disjunctive convex programs. Following the results in Sawaya and Grossmann (2006) for the case of Linear GDP, we will show that any convex GDP can be equivalently represented as a disjunctive convex program. As a result, we are able to exploit the new developments on disjunctive convex programming to improve the solution methods of generalized disjunctive programs.

This paper is organized as follows. In section 2 we present the general structure of a disjunctive convex program and we define an operation equivalent to a basic step for disjunctive linear programs that takes a disjunctive set to another with fewer number of conjuncts. The strength of relaxations increases as the number of conjuncts decreases. We prove that the tightest of these relaxations, allows in theory the solution of the disjunctive convex program as a nonlinear programming problem. At the end of this section we present a methodology to guide the generation of strong relaxations without incurring an exponential increase of the size of the reformulated mixed-integer program. In section 3 we introduce the general structure of a generalized disjunctive program and a transformation that takes it to a disjunctive convex program. We review traditional solutions methods and show how the theory developed for disjunctive convex programs can be used to improve their computational efficiency. In sections 4 and 5 we illustrate the application of the methodolgy through a set of numerical examples.

## 2. Disjunctive Convex Programming

We define disjunctive convex programming as the optimization over a disjunctive set. Each disjunctive set is defined by convex inequalities connected to each other by the operation of conjunction ($\wedge$, juxtaposition "and") or disjunction ($\vee$, "or"). Since each convex inequality defines a convex set, a disjunctive set can be viewed as a collection of convex inequalities joined together by the intersection ($\cap$) or union ($\cup$). A general representation of a disjunctive convex program is given by: $\min\{f(x)|x \in F\}$, where $F$ is a disjunctive set, $f(x) : R^n \to R^1$.

### *2.1. Disjunctive sets and their equivalent forms*

We denote the convex set defined by a convex inequality as $C = \{x \in R^n|\phi(x) \leq 0\}$, where $\phi(x) : R^n \to R^1$ is a convex function. Given a collection of sets such that $C_j = \{x \in R^n|\phi_j(x) \leq 0\}, j \in M$, we call their union, an elementary disjunctive set, which is defined as:

$$H = \bigcup_{j \in M} C_j = \{x \in R^n| \bigvee_{j \in M} \phi_j(x) \leq 0\}$$

and their intersection, a new convex set, which is defined as:

$$P = \bigcap_{j \in M} C_j = \{x \in R^n| \bigwedge_{j \in M} \phi_j(x) \leq 0\}$$

A disjunctive set can be expressed in many different forms that are logically equivalent and that can be obtained from each other by considering $F$ as a logical expression whose statements forms are inequalities, and applying rules of propositional calculus. Among these forms, the two extreme ones are the conjunctive normal form (CNF),

$$F_{CNF} = \bigcap_{i \in T} H_i$$

where each $H_i$ is an elementary disjunctive set, and the disjunctive normal form (DNF)

$$F_{DNF} = \bigcup_{i \in D} P_i$$

where each $P_i$ is a convex set.

In this paper we define Regular Form (RF) as the form between the (CNF) and (DNF) 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 $i \in D_k$.

In the remainder of the paper we assume $P_i = \{x \in R^n | g_i(x) \leq 0\}$ where $g_i(x) : R^n \to R^m$ and $g_i(x)$ is convex. We are now ready to define an operation, which when applied to a disjunctive set in regular form, results in another regular form with one less conjunct, i.e., an operation which brings the disjunctive set closer to the DNF. As in Balas (1985), we call this operation a **basic step** that is defined in the following theorem, which is an extension of Balas' work.

**Theorem 2.1.** *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:*

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

Theorem 2.1 provides a tool that allows the systematic generation of equivalent disjunctive sets. The following is an illustrative example of the theorem applied on a disjunctive set with two disjunctions.

Given the disjunctive set $F_1 = S_1 \cap S_2$ where,

$$S_1 = [g_{11}(x) \leq 0] \vee [g_{12}(x) \leq 0]$$

and,

$$S_2 = [g_{21}(x) \leq 0] \vee [g_{22}(x) \leq 0] \vee [g_{23}(x) \leq 0]$$

where $g_{ij}(x) : R^n \to R^m$ for $i = 1, 2$ and $j = 1, 2, 3$ are convex functions. The application of a basic step between $S_1$ and $S_2$ leads to $F_2 = S_{12}$ defined as:

$$\begin{aligned} S_{12} = \quad & [g_{11}(x) \leq 0 \wedge g_{21}(x) \leq 0] \vee [g_{11}(x) \leq 0 \wedge g_{22}(x) \leq 0] \\ & \vee [g_{11}(x) \leq 0 \wedge g_{23}(x) \leq 0] \vee [g_{12}(x) \leq 0 \wedge g_{21}(x) \leq 0] \\ & \vee [g_{12}(x) \leq 0 \wedge g_{22}(x) \leq 0] \vee [g_{12}(x) \leq 0 \wedge g_{23}(x) \leq 0] \end{aligned}$$

### *2.2. The convex hull of a disjunctive set*

The following characterization expresses the convex hull of a disjunctive set in DNF as the projection of a higher dimensional convex set onto $R^n$ as given in Ceria and Soares (1999).

**Theorem 2.2.** *Let W be given by $W = cl\,conv(S)$, $S = \bigcup_{i \in D} P_i$, and $D = \{i : P_i \neq \emptyset\}$. If each set $P_i$ is bounded, $x \in W$ if and only if there exist vectors $(\lambda_i, \nu^i)$, for every $i \in D$, such that the following nonlinear system is feasible*

$$\begin{array}{ll} x = \sum_{i \in D} \nu^i & \\ (cl\, g'_i)(\nu^i, \lambda_i) \leq 0, & i \in D \\ \sum_{i \in D} \lambda_i = 1,\ \lambda_i \geq 0, & i \in D \end{array} \qquad (1)$$

*where generically, $(cl\, g')(x, \lambda)$ denotes the closure of the perspective mapping of $g(x)$ at $(x, \lambda)$ and where $g'(x, \lambda) = \lambda g(x/\lambda)$ and its domain $D_{g'} = \{(x, \lambda) | x/\lambda \in Dom(g), \lambda > 0\}$.*

For the particular case in which $|x| \leq L$ , i.e. each continuous variable is bounded below as well as bounded above, where L is a sufficiently large number we can express the aforementioned system as (Stubbs and Mehrotra, 1999):

$$\begin{array}{ll} x = \sum_{i \in D} \nu^i & \\ (cl\, g'_i)(\nu^i, \lambda_i) \leq 0, & i \in D \\ \sum_{i \in D} \lambda_i = 1,\ \lambda_i \geq 0, & i \in D \\ |\nu^i| \leq L\lambda_i & i \in D \end{array} \qquad (2)$$

where $(cl\, g'_i)(\nu^i, \lambda_i) = \lambda_i g_i(\nu^i/\lambda_i)$ if $\lambda_i > 0$ and $(cl\, g'_i)(\nu^i, \lambda_i) = 0$ if $\lambda_i = 0$.

It is important to note that each variable $\lambda_i$ is associated to the disjunct $i$ in the disjunction through the corresponding perspective function. In particular, when $\lambda_i = 1$ all the constraints in the associated disjunct are enforced. As it is described in the next section, this observation is of main importance in order to find an equivalent mixed-integer nonlinear representation of the disjunctive convex program.

### *2.3. MINLP formulation for disjunctive convex programs*

The following theorem describes how to formulate a disjunctive program defined on a disjunctive set in DNF as a convex MINLP.

**Theorem 2.3.** *Let $Z = \min\{f(x) | x \in S\}$ be a disjunctive convex program where S is a disjunctive set in DNF such that $S = \bigcup_{i \in D} P_i$ and $P_i = \{x \in R^n, g_i(x) \leq 0\}, i \in D$*

*and $x$ is bounded below as well as bounded above by a large number $L$. Then, an equivalent mixed-integer nonlinear program can be described as:*

$$
\begin{array}{ll}
\min & f(x) \\
s.t. & \\
& x = \sum\limits_{i \in D} \nu^i \\
& (cl\, g_i')(\nu^i, \lambda_i) \leq 0, \quad i \in D \\
& \sum\limits_{i \in D} \lambda_i = 1,\ \lambda_i \geq 0, \quad i \in D \\
& |\nu^i| \leq L\lambda_i, \quad i \in D \\
& \lambda_i \in \{0,1\}, \quad i \in D
\end{array}
\tag{3}
$$

*where $(cl\, g_i')(\nu^i, \lambda_i) = \lambda_i g_i(\nu^i/\lambda_i)$ if $\lambda_i > 0$ and $(cl\, g_i')(\nu^i, \lambda_i) = 0$ if $\lambda_i = 0$.*

**Proof.** The proof trivially follows from Theorem 2.2. □

However, more often than not the disjunctive set is not in DNF but rather somewhere between CNF and DNF. A valid relaxation for this set is given by the hull relaxation, which is defined as the intersection of the convex hull of each disjunction. More precisely, given a disjunctive set $F = \bigcap\limits_{k \in K} S_k$ the relaxation of $(F)$, denoted as $h\text{-}rel(F)$ is defined as $h\text{-}rel(F) = \bigcap\limits_{k \in K} cl\, conv(S_k)$ where $cl\, conv(S_k)$ represents the closure of the convex hull of $S_k$.

Hence, Theorem 2.3 can be extended to represent a disjunctive program given in regular form as follows:

$$
\begin{array}{llll}
\min & f(x) & & \\
s.t. & & & \\
& x = \sum\limits_{i \in D_k} \nu^{ik}, & & k \in K \\
& (cl\, g_{ik}')(\nu^{ik}, \lambda_{ik}) \leq 0, & i \in D_k, & k \in K \\
& \sum\limits_{i \in D_k} \lambda_{ik} = 1,\ \lambda_{ik} \geq 0, & i \in D_k, & k \in K \\
& |\nu^{ik}| \leq L\lambda_{ik}, & i \in D_k & k \in K \\
& \lambda_{ik} \in \{0,1\}, & i \in D_k & k \in K
\end{array}
\tag{4}
$$

Applying the mixed-integer formulation presented above would lead to numerical difficulties to the NLP subproblems since the function $(cl\, g_{ik}')(\nu^{ik}, \lambda_{ik})$ is not differentiable at $\lambda_{ik} = 0$. In order to overcome the latter issue, we propose to use the approximation given by Sawaya (2006):

$$(cl\, g'_{ik})(\nu_{ik}, \lambda_{ik}) = ((1-\epsilon)\lambda_{ik} + \epsilon) g_{ik}(\nu^{ik}/(1-\epsilon)\lambda_{ik} + \epsilon) - \epsilon g_{ik}(0)(1-\lambda_{ik}) \leq 0 \quad (5)$$

The above approximation function, which can be shown to be convex provided $g_{ik}(x)$ is convex, is exact for $\lambda_{ik} = 1$ and $\lambda_{ik} = 0$, independently of $\epsilon$. This parameter, which represents a small non-negative tolerance, only has an impact on the accuracy of the relaxation. Clearly this approximation fails if $g_{ik}(x)$ is not defined at the zero point. However, we can overcome this issue by making a suitable variable transformation such that:

$$\lambda_i g_{ik}(\nu^{ik}/\lambda_{ik}) = ((1-\epsilon)\lambda_{ik} + \epsilon) g'_{ik}(\tau^{ik}/(1-\epsilon)\lambda_{ik} + \epsilon) - \epsilon g'_{ik}(0)(1-\lambda_{ik}) \leq 0 \quad (6)$$

where $g'_{ik}(y) = g_{ik}(y+\beta)$ and $y + \beta = x$. Note that $\tau^{ik}$ represents the disaggregated variable associated to $y$. Also note that convexity is preserved in this transformation provided the original function $g_{ik}$ is convex.

In the following section we show that a generally tighter relaxation can be obtained if the hull relaxation is calculated not on the orginal disjunctive set but rather on an equivalent set obtained after the application of basic steps. This in turn means that the continuous relaxation of the resulting MINLP will be tightened. Moreover, if the final form obtained after the application of basic steps is in DNF then its hull relaxation is the convex hull of the disjunctive set.

### *2.4. Hierarchy of relaxations*

The following theorem, which is an extension of Balas' work (see Balas (1985)) and we state without proof, shows that even though the disjunctive sets obtained from the application of basic steps are equivalent, their relaxations are not. Moreover, when a basic step is applied to a disjunctive set its hull relaxation is in general tightened.

**Theorem 2.4.** *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:*

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

#### *2.4.1. Illustrative example:*

Given the disjunctive set $F_1 = S_1 \cap S_2$ where $S_1 = P_{11} \cup P_{12}$ and $S_2 = P_{21} \cup P_{22}$, the hull relaxation is given in Figure 1 (a). After the application of a basic step, an equivalent disjunctive set is obtained $F_2 = (P_{11} \cap P_{21}) \cup (P_{11} \cap P_{22}) \cup (P_{12} \cap P_{21}) \cup (P_{12} \cap P_{22})$ whose hull relaxation can be seen in Figure 1 (b). Clearly $h\text{-}rel(F_2) \subseteq$

$h\text{-}rel(F_1)$

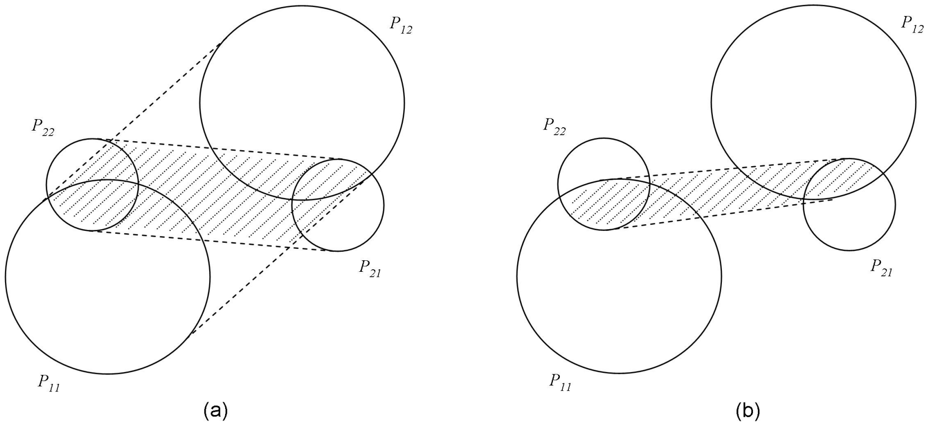

Figure 1: (a): $h\text{-}rel(F_1)$ (b): $h\text{-}rel(F_2)$

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. As in the case of disjunctive linear programming, we present the following theorem that establishes that no increase of the number of 0-1 variables is required.

**Theorem 2.5.** *Let $Z = \min\{f(x)|x \in F_d\}$ be a disjunctive convex program with the variables x bounded below as well as bounded 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 $\bigvee_{s\in Q_r} (g_s(x) \leq 0), r \in T_d$ and let $F_n$ be 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 disjunctive normal form of the set of disjunctions $\bigvee_{s\in Q_r} (g_s(x) \leq 0), r \in T_{dj}$. Furthermore, let $M_j^t$ be the index set of the inequalities $g_s(x) \leq 0$ making up the system $G^t(x) \leq 0$ for a given $j \in T_n$ and $t \in Q_j$. Then, an equivalent mixed-integer nonlinear program can be described as:*

$$
\begin{array}{lll}
\min & f(x) & \\
s.t. & & \\
& x = \sum\limits_{t \in Q_j} \nu^t, & j \in T_n \\
& (cl\, G'^t)(\nu^t, \lambda_t) \leq 0, & t \in Q_j, \quad j \in T_n \\
& \sum\limits_{t \in Q_j} \lambda_t = 1, & j \in T_n \\
& \lambda_t \geq 0, & t \in Q_j \\
& \sum\limits_{t \in Q_j | s \in M_j^t} \lambda_t = \delta_s^r, & s \in Q_r, \quad r \in T_d, j \in T_n \\
& \sum\limits_{s \in Q_r} \delta_s^r = 1, & r \in T_d \\
& |\nu^t| \leq L\lambda_t, & t \in Q_j, \quad j \in T_n \\
& \delta_s^r \in \{0, 1\}, & s \in Q_r, \quad r \in T_d
\end{array}
\qquad (7)
$$

where $(cl\, G'^t)(\nu^t, \lambda_t) = \lambda_t G^t(\nu^t/\lambda_t)$ if $\lambda_t > 0$ and $(cl\, G'^t)(\nu^t, \lambda_t) = 0$ if $\lambda_t = 0$

**Proof.** The proof follows from Theorem 4.4 of Balas work (see Balas (1985)).

To illustrate the above theorem, let us consider the disjunctive set $F_d$ with index sets $T_d = \{r_1, r_2\}$ such that $Q_{r_1} = \{s_1^{r1}, s_2^{r1}, s_3^{r1}\}$ and $Q_{r_2} = \{s_1^{r2}, s_2^{r2}, s_3^{r2}\}$. After the application of a basic step between the disjunctions $r_1$ and $r_2$ a new disjunctive set $F_n$ is obtained with the following index sets, $T_n = \{j_1\}$ such that $Q_{j_1} = \{t_1^{j_1}, t_2^{j_1}, t_3^{j_1}, t_4^{j_1}, t_5^{j_1}, t_6^{j_1}, t_7^{j_1}, t_8^{j_1}, t_9^{j_1}\}$. The direct application of Theorem 2.3 on $F_n$ would require the use of 9 binary variables as opposed to 6 binary variables needed to represent $F_d$. However, by following Theorem 2.5 $F_n$ can still be described through 6 binary variables. Note that for this example, $M_{j_1}^{t_1} = \{s_1^{r1}, s_1^{r2}\}$, $M_{j_1}^{t_2} = \{s_1^{r1}, s_2^{r2}\}$,$M_{j_1}^{t_3} = \{s_1^{r1}, s_3^{r2}\}$, $M_{j_1}^{t_4} = \{s_2^{r1}, s_1^{r2}\}$, $M_{j_1}^{t_5} = \{s_2^{r1}, s_2^{r2}\}$, $M_{j_1}^{t_6} = \{s_2^{r1}, s_3^{r2}\}$, $M_{j_1}^{t_7} = \{s_3^{r1}, s_1^{r2}\}$, $M_{j_1}^{t_8} = \{s_3^{r1}, s_2^{r2}\}$, $M_{j_1}^{t_9} = \{s_3^{r1}, s_3^{r2}\}$, hence, the equations that enforce the integrality of the continuous variables $\lambda$ that represent each term in the disjunctive set $F_n$ based on the integrality of the variables $\delta$ that represent each term in the disjunctive set $F_d$ are:

$$
\begin{array}{l}
\delta_{s_1}^{r_1} = \lambda_{t_1}^{j1} + \lambda_{t_2}^{j1} + \lambda_{t_3}^{j1} \\
\delta_{s_2}^{r_1} = \lambda_{t_4}^{j1} + \lambda_{t_5}^{j1} + \lambda_{t_6}^{j1} \\
\delta_{s_3}^{r_1} = \lambda_{t_7}^{j1} + \lambda_{t_8}^{j1} + \lambda_{t_9}^{j1} \\
\delta_{s_1}^{r_2} = \lambda_{t_1}^{j1} + \lambda_{t_4}^{j1} + \lambda_{t_7}^{j1} \\
\delta_{s_2}^{r_2} = \lambda_{t_2}^{j1} + \lambda_{t_5}^{j1} + \lambda_{t_8}^{j1} \\
\delta_{s_3}^{r_2} = \lambda_{t_3}^{j1} + \lambda_{t_6}^{j1} + \lambda_{t_9}^{j1}
\end{array}
$$

$$\lambda_{t_1}^{j1} + \lambda_{t_2}^{j1} + \lambda_{t_3}^{j1} + \lambda_{t_4}^{j1} + \lambda_{t_5}^{j1} + \lambda_{t_6}^{j1} + \lambda_{t_7}^{j1} + \lambda_{t_8}^{j1} + \lambda_{t_9}^{j1} = 1$$

$$\delta_{s_1}^{r_1} + \delta_{s_2}^{r_1} + \delta_{s_3}^{r_1} = 1$$
$$\delta_{s_1}^{r_2} + \delta_{s_2}^{r_2} + \delta_{s_3}^{r_2} = 1$$

$$0 \leq \lambda_{t_1}^{j1}, \lambda_{t_2}^{j1}, \lambda_{t_3}^{j1}, \lambda_{t_4}^{j1}, \lambda_{t_5}^{j1}, \lambda_{t_6}^{j1}, \lambda_{t_7}^{j1}, \lambda_{t_8}^{j1}, \lambda_{t_9}^{j1} \leq 1$$

$$\delta_{s_1}^{r_1}, \delta_{s_2}^{r_1}, \delta_{s_3}^{r_1}, \delta_{s_1}^{r_2}, \delta_{s_2}^{r_2}, \delta_{s_3}^{r_2} \in \{0, 1\}$$

Let us consider for example the case for which $\delta_{s_1}^{r_1} = 1$ and $\delta_{s_1}^{r_2} = 1$ then from the equations above it is clear that $\lambda_{t_1}^{j1} = 1$, as expected.

### *2.5. Convex nonlinear program equivalent to a disjunctive convex program*

In this section we show that any disjunctive convex program can be equivalently represented as a convex nonlinear program in the sense that at least one solution of both programs is the same.

**Theorem 2.6.** *Let $Z = \min\{f(x)|x \in S\}$ be a disjunctive convex program where $S$ is a disjunctive set in DNF such that $S = \bigcup_{i \in D} P_i$ and $P_i = \{x \in R^n, g_i(x) \leq 0\}$ where $P_i \neq \emptyset$ and that $x$ and $f(x)$ are bounded below as well as bounded above by a large number $L$. Then, the following convex nonlinear program has at least one solution that is also solution of the disjunctive program:*

$$\begin{array}{lll}
\min & \alpha & \\
s.t. & & \\
& \alpha = \sum\limits_{i \in D} \nu_\alpha^i & \\
& x = \sum\limits_{i \in D} \nu^i & \\
& (cl\, g_i')(\nu^i, \lambda_i) \leq 0, & i \in D \\
& (cl\, f')((\nu^i, \nu_\alpha^i), \lambda_i) \leq 0, & i \in D \\
& \sum\limits_{i \in D} \lambda_i = 1, \;\; \lambda_i \geq 0, & i \in D \\
& |\nu^i| \leq L\lambda_i, & i \in D \\
& |\nu_\alpha^i| \leq L\lambda_i, & i \in D
\end{array} \qquad (8)$$

*where $(cl\, g_i')(\nu^i, \lambda_i) = \lambda_i g_i(\nu^i/\lambda_i)$ if $\lambda_i > 0$ and $(cl\, g_i')(\nu^i, \lambda_i) = 0$ if $\lambda_i = 0$. Also, $(cl\, f')((\nu^i, \nu_\alpha^i), \lambda_i) = \lambda_i f(\nu^i/\lambda_i) - \nu_\alpha^i$ if $\lambda_i > 0$ and $(cl\, f')((\nu^i, \nu_\alpha^i), \lambda_i) = 0$ if $\lambda_i = 0$.*

**Proof.**

Clearly, the disjunctive problem $Z = \min\{f(x)|x \in S\}$ is equivalent to $Z' = \min\{\alpha|\alpha \geq f(x), x \in S\}$. Then, from Theorem 2.1 this program is, in turn, equivalent to $Z'' = \min\{\alpha|(x, \alpha) \in S''\}$ where $S''$ is the DNF form of the disjunctive set $S \cap S_f$ and

$S_f = \{(\alpha, x) \in R^{n+1}, \alpha \geq f(x)\}$. As a result, from Theorem 2.3, the optimization problem in (8) is nothing but the minimization of $\alpha$ over the convex hull of $S''$ and can be represented as $Z''' = \min\{\alpha | (x, \alpha) \in h\text{-}rel(S'')\}$. From Ceria and Soares' work on the optmization of a linear function over a disjunctive set in DNF (see (Ceria and Soares, 1999)), this nonlinear program has at least one solution that is also a solution of $Z'' = \min\{\alpha | (x, \alpha) \in S''\}$ and hence a solution of $Z = \min\{f(x) | x \in S\}$. □

Even though the value of the optimal objective in (8) is the same as the optimal objective value of $Z = \min\{f(x)|x \in S\}$, Theorem 2.6 does not guarantee the integrality of $\lambda$. In order to overcome this issue and generate an integral solution from the solution of (8) we state the following corollary.

**Corollary 2.7.** *Given an optimal solution of problem (8)* $(\alpha^*, x^*, (\nu_\alpha^{i*}, \nu^{i*}, \lambda_i^*)_{i \in D})$, *an optimal solution of the convex disjunctive problem* $Z = \min\{f(x)|x \in S\}$ *is given by* $\nu^{i*}/\lambda_i^*$ *for any* $i \in D$ *such that* $\lambda_i^* > 0$.

**Proof.**

From (Lee and Grossmann, 2000) the point $(\alpha^i, x^i) = (\nu_\alpha^{i*}/\lambda_i^*, \nu^{i*}/\lambda_i^*)$ belongs to the i-th term in the disjunction defined by $S''$. Clearly, $(\alpha^*, x^*) = \sum_{i \in D, \lambda_i^* > 0} \lambda_i^* (\alpha^i, x^i)$. Since any $(\alpha^i)$ cannot be less than $\alpha^*$ (i.e. $\alpha^*$ is a lower bound) then $(\alpha^i)$ must be equal to $(\alpha^*)$. Hence, any point $(\alpha^i, x^i)$ is a solution of $Z'' = \min\{\alpha | (x, \alpha) \in S''\}$ or similarly, any point $\nu^{i*}/\lambda_i^*, i \in D$ such that $\lambda_i^* > 0$ is solution of $Z = \min\{f(x)|x \in S\}$. □

*2.5.1. Illustrative example:*

This example is adapted from Lee and Grossmann (2000), and consists of the optimization of a convex function on a disjunctive set in DNF described as follows:

$$\begin{aligned}
\min \quad & Z = (x_1 - 3)^2 + (x_2 - 2)^2 + 1 \\
s.t. \quad & \\
& [x_1^2 + x_2^2 \leq 1] \vee [(x_1 - 4)^2 + (x_2 - 1)^2 \leq 1] \qquad (CIRC_1) \\
& \vee [(x_1 - 2)^2 + (x_2 - 4)^2 \leq 1] \\
& |x_i| \leq 5 \quad i \in \{1, 2\}
\end{aligned}$$

In Figure 2, the optimal solution $Z^*$ of the disjunctive convex program is shown. Also $Z^{rel}$ denotes the solution of the problem on the hull relaxation of the disjunctive set (i.e. $Z^{rel} = 1$ at $x_1 = 3$ and $x_2 = 2$) which can be obtained by solving the following

NLP problem,

$$\begin{array}{ll} \min & Z = (x_1 - 3)^2 + (x_2 - 2)^2 + 1 \\ s.t. & \\ & \begin{array}{ll} x_1 = \sum\limits_{i \in \{1,2,3\}} \nu_1^i & \\ x_2 = \sum\limits_{i \in \{1,2,3\}} \nu_2^i & (RCIRC_1) \\ (cl\, g'^1)(\nu^1, \lambda_1) \leq 0 & \\ (cl\, g'^2)(\nu^2, \lambda_2) \leq 0 & \\ (cl\, g'^3)(\nu^3, \lambda_3) \leq 0 & \\ \sum\limits_{i \in \{1,2,3\}} \lambda_i = 1, \;\; \lambda_i \geq 0, & i \in \{1,2,3\} \\ |\nu_1^i| \leq 5\lambda_i, & i \in \{1,2,3\} \\ |\nu_2^i| \leq 5\lambda_i, & i \in \{1,2,3\} \end{array} \end{array}$$

where,
$(cl\, g'^1)(\nu^1, \lambda_1) = \lambda_1[(\nu_1^1/\lambda_1)^2 + (\nu_2^1/\lambda_1)^2 - 1]$ if $\lambda_1 > 0$ and $(cl\, g'^1)(\nu^1, \lambda_1) = 0$ if $\lambda_1 = 0$; $(cl\, g'^2)(\nu^2, \lambda_2) = \lambda_2[(\nu_1^2/\lambda_2 - 4)^2 + (\nu_2^2/\lambda_2 - 1)^2 - 1]$ if $\lambda_2 > 0$ and $(cl\, g'^2)(\nu^2, \lambda_2) = 0$ if $\lambda_2 = 0$; $(cl\, g'^3)(\nu^3, \lambda_3) = \lambda_3[(\nu_1^3/\lambda_2 - 2)^2 + (\nu_2^3/\lambda_3 - 4)^2 - 1]$if $\lambda_3 > 0$ and $(cl\, g'^3)(\nu^3, \lambda_3) = 0$ if $\lambda_3 = 0$.

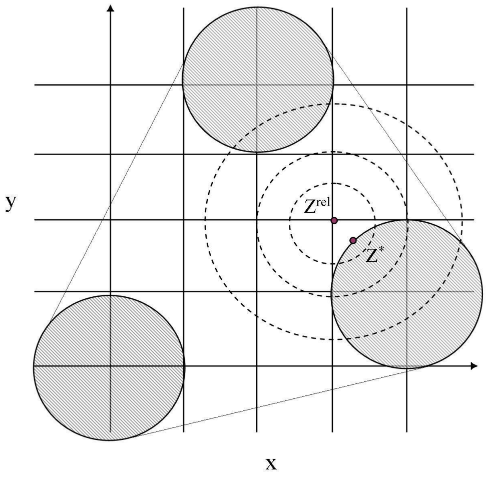

Figure 2: Optimal solution of illustrative example

An alternative NLP formulation for the above problem can be obtained by following Theorem 2.6, which is nothing but the relaxed mixed-integer formulation of the following disjunctive convex program:

$$
\begin{array}{ll}
\min & Z = \alpha \qquad\qquad\qquad\qquad\qquad\qquad (CIRC_2) \\
s.t. & \\
& \left[\begin{array}{c} x_1^2 + x_2^2 \leq 1 \\ (x_1-3)^2 + (x_2-2)^2 + 1 \leq \alpha \end{array}\right] \vee \left[\begin{array}{c} (x_1-4)^2 + (x_2-1)^2 \leq 1 \\ (x_1-3)^2 + (x_2-2)^2 + 1 \leq \alpha \end{array}\right] \\
& \vee \left[\begin{array}{c} (x_1-2)^2 + (x_2-4)^2 \leq 1 \\ (x_1-3)^2 + (x_2-2)^2 + 1 \leq \alpha \end{array}\right] \\
& |x_i| \leq 5 \quad i \in \{1,2\}
\end{array}
$$

Namely,

$$
\begin{array}{lll}
\min & Z = \alpha & \\
s.t. & & \\
& x_1 = \sum\limits_{i\in\{1,2,3\}} \nu_1^i & \\
& x_2 = \sum\limits_{i\in\{1,2,3\}} \nu_2^i & (RCIRC_2) \\
& \alpha = \sum\limits_{i\in\{1,2,3\}} \nu_\alpha^i & \\
& (cl\, g'^1)(\nu^1, \lambda_1) \leq 0 & \\
& (cl\, g'^2)(\nu^2, \lambda_2) \leq 0 & \\
& (cl\, g'^3)(\nu^3, \lambda_3) \leq 0 & \\
& (cl\, f')((\nu^i, \nu_\alpha^i), \lambda_i) \leq 0 \quad i \in \{1,2,3\} & \\
& \sum\limits_{i\in\{1,2,3\}} \lambda_i = 1,\ \lambda_i \geq 0, \quad i \in \{1,2,3\} & \\
& |\nu_1^i| \leq 5\lambda_i, \qquad\qquad i \in \{1,2,3\} & \\
& |\nu_2^i| \leq 5\lambda_i, \qquad\qquad i \in \{1,2,3\} & \\
& |\nu_\alpha^i| \leq L\lambda_i, \qquad\qquad i \in \{1,2,3\} &
\end{array}
$$

where, $(cl\, g'^1)(\nu^1, \lambda_1) = \lambda_1[(\nu_1^1/\lambda_1)^2 + (\nu_2^1/\lambda_1)^2 - 1]$ if $\lambda_1 > 0$ and $(cl\, g'^1)(\nu^1, \lambda_1) = 0$ if $\lambda_1 = 0$; $(cl\, g'^2)(\nu^2, \lambda_2) = \lambda_2[(\nu_1^2/\lambda_2 - 4)^2 + (\nu_2^2/\lambda_2 - 1)^2 - 1]$ if $\lambda_2 > 0$ and $(cl\, g'^2)(\nu^2, \lambda_2) = 0$ if $\lambda_2 = 0$; $(cl\, g'^3)(\nu^3, \lambda_3) = \lambda_3[(\nu_1^3/\lambda_2 - 2)^2 + (\nu_2^3/\lambda_3 - 4)^2 - 1]$if $\lambda_3 > 0$ and $(cl\, g'^3)(\nu^3, \lambda_3) = 0$ if $\lambda_3 = 0$; $(cl\, f')((\nu^i, \nu_\alpha^i), \lambda_i) = \lambda_i[(\nu_1^i/\lambda_i - 3)^2 + (\nu_2^i/\lambda_i -$

$2)^2 - \nu^i_\alpha/\lambda_i + 1]$ if $\lambda_i > 0$ and $(cl\, f')((\nu^i, \nu^i_\alpha), \lambda_i) = 0$ if $\lambda_i = 0$.

Note that $(CIRC_2)$ can be obtained from $(CIRC_1)$ by writing the objective function as a constraint and applying a basic step with the proper disjunction. Figure 3 shows the representation of the above disjunctive convex program and its relaxation. Note that the optimal solution $Z^* = 1.17$ projected on the $x$ space (i.e. $x_1 = 3.29$ and $x_2 = 1.71$) is the same as the original disjunctive program, which is the same as the solution when the disjunctive set is relaxed. Thus, the solution of the disjunctive program $(CIRC_1)$ is the same as the solution of the nonlinear program $(RCIRC_2)$!

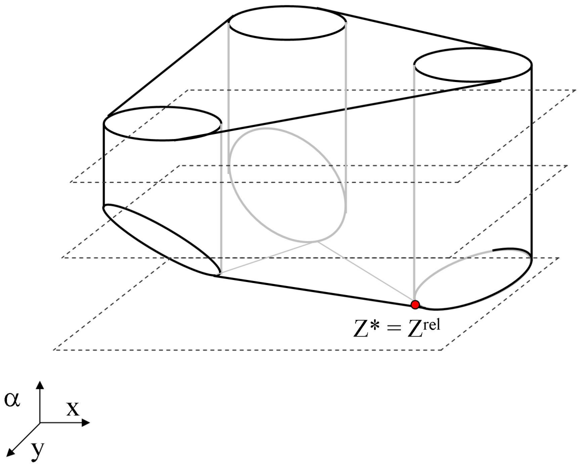

Figure 3: Optimal solution of illustrative example

In summary, using Theorem 2.3 in combination with Theorem 2.6 and Theorem 2.4, and bringing the disjunctive set to DNF, we can find the value of the objective function of any disjunctive convex program by solving a single nonlinear program. Moreover, by following Corollary 2.7, a simple arithmetic operation on the solution of the relaxed NLP leads to a solution of the disjunctive convex program.

### *2.6. Rules for applying basic steps*

Although ideally one would like to convert a convex disjuntive program into DNF form to solve it as continuous nonlinear program as stated in Theorem 2.6, this is clearly not practical given the exponential number of terms that might arise in the

DNF form. Therfore, in order to apply only a subset of basic steps, and in order to make good use of the hierarchy described in section 2.4., one important aspect is to understand which basic steps will lead to an improvement in the tightness of the relaxation, and hence in a potential increase in the lower bound of the global optimum. In other words, we need to be able to differentiate among the basic steps that will lead to a strict inclusion with those that will keep the relaxation unchanged. As described in the work of Ruiz and Grossmann (2010) the following propositions provide sufficient conditions for a particular disjunctive set for which a basic step will not lead to a tighter relaxation.

**Proposition 2.8.** *Let $S_1$ and $S_2$ be two disjunctive sets defined in $R^n$. If the set of variables constrained in $S_1$ are not constrained in $S_2$ , and the set of variables constrained in $S_2$ are not constrained in $S_1$, then h-rel$(S_1 \cap S_2)$ = h-rel$(S_1) \cap$ h-rel$(S_2)$ .*

**Proposition 2.9.** *Let $S = \bigcup_i P_i$ , where $P_i, i = \{1, 2\}$ are convex sets defined in the x space, $H$ is a half space defined by $ax + b \leq 0$ and $H^*$ is a facet of $H$. If $P_1$ is a point such that $P_1 \subseteq H^*$ then h-rel$(S \cap H)$ = h-rel$(S) \cap$ h-rel$(H)$.*

An illustration of Proposition 2.9 is shown in Figure 4.

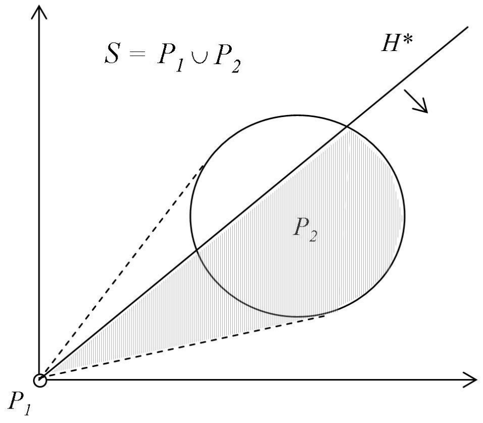

Figure 4: h-rel$(S \cap H)$ = h-rel$(S) \cap$ h-rel$(H)$

Another important aspect to consider is the effect that a particular basic step has in the increase of the size of the formulation. In this respect we can differentiate two types of basic steps. Firstly, the ones that are implemented between two proper disjunctions, and second, the ones that are implemented between a proper and an improper disjunction. In this work we propose to use the latter approach. Note that in this case, parallel basic steps (i.e. simultaneous intersection of each improper disjunction with all proper disjunctions) will not lead to an increase in the number of convex sets in the disjunctive set, keeping the size of the formulation smaller. To illustrate this, let us consider the disjunctive set given by $F_1 = S_1 \cap S_2$ where $S_1 = P_1 \cup P_2$ and $S_2 = P_3$ are a proper and improper disjunction respectively. Clearly, the application of a basic step between $S_1$ and $S_2$ leads to the following disjunctive set $F_2 = S_{12} = (P_1 \cap P_3) \cup (P_2 \cap P_3)$ with no more convex sets than in $F_1$.

We are now ready to present an improved strategy based on the theory developed for disjunctive convex programing to solve generalized disjunctive programs more efficiently. In the following section we introduce the general structure of GDP and review the traditional approaches to solve them.

## 3. Convex Generalized Disjunctive Programs

The general structure of a GDP, which facilitates the formulation of discrete/continuous optimization problems, can be represented as follows (Raman & Grossmann, 1994):

$$\begin{array}{lll}
\min & Z = f(x) + \sum_{k\in K} c_k & \\
s.t. & r(x) \leq 0 & \\
 & \bigvee_{i\in D_k} \left[\begin{array}{c} Y_{ik} \\ g_{ik}(x) \leq 0 \\ c_k = \gamma_{ik} \end{array}\right] & k \in K \qquad (GDP) \\
 & \underline{\vee}_{i\in D_k}\ Y_{ik} & k \in K \\
 & \Omega(Y) = True & \\
 & x^{lo} \leq x \leq x^{up} & \\
 & x \in R^n \,,\ c_k \in R^1,\ Y_{ik} \in \{True, False\} &
\end{array}$$

where $f : R^n \to R^1$ is a function of the continuous variables $x$ in the objective function, $r : R^n \to R^l$ 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 $Y_{ik}$, a set of inequalities $g_{ik}(x) \leq 0,\ \ g_{ik} : R^n \to R^j$, and a cost variable $c_k$. If $Y_{ik}$ is true, then $g_{ik}(x) \leq 0$ and $c_k = \gamma_{ik}$ are enforced; otherwise they are ignored. Also, $\Omega(Y) = True$ are logic propositions for the Boolean variables expressed in the conjunctive normal form $\Omega(Y) = \bigwedge_{t=1,2,..T} \left[ \bigvee_{(i,k)\in R_t} (Y_{ik}) \bigvee_{(i,k)\in Q_t} (\neg Y_{ik}) \right]$ where for each clause t=1,2 … 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 expression $\underset{i\in D_k}{\veebar} Y_{ik}$ ensures that only one term in each disjunction is True.

*3.1. Equivalency between disjunctive convex programs and convex GDP*

In order to make use of the theory developed for disjunctive convex programs defined in the continuous space in order to solve convex GDP defined in the Boolean and continuous space, we will propose a transformation that takes a convex GDP and represents it as a disjunctive convex program.

This transformation, which is equivalent to the one proposed by Sawaya and Grossmann for linear GDP (see Sawaya (2006)), 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 $\lambda$ is a vector of continuous variables whose domain is [0,1], and convert the logical relations $\underset{i\in D_k}{\veebar} Y_{ik}$ and $\Omega(Y) = True$ into the algebraic equations $\sum_{i\in D_k} \lambda_{ik} = 1$, $k \in K$ and $A\lambda \geq a$, respectively. This yields the following disjunctive model:

$$
\begin{array}{lll}
\min & Z = f(x) + \sum_{k\in K} c_k & \\
s.t. & r(x) \leq 0 & \\
 & \underset{i\in D_k}{\vee} \left[ \begin{array}{c} \lambda_{ik} = 1 \\ g_{ik}(x) \leq 0 \\ c_k = \gamma_{ik} \end{array} \right] & k \in K \qquad (CGDP) \\
 & \sum_{i\in D_k} \lambda_{ik} = 1 & k \in K \\
 & A\lambda \geq a & \\
 & x^{lo} \leq x \leq x^{up} & \\
 & x \in R^n \, , \; c_k \in R^1, \; \lambda_{ik} \in \{0,1\} &
\end{array}
$$

In the following proposition we state the equivalence between (GDP) and (CGDP), which can be easily proved by considering the above-mentioned mapping between the Boolean variables Y and continuous variables $\lambda$.

**Proposition 3.1.** *The model (GDP) is equivalent to (CGDP) in the sense that every point (x,c,Y) in the feasible region of (GDP) has a one to one correspondence with every point (x,c,$\lambda$) in (CGDP) and that the optimal value of the objective function in (GDP) is the same as the optimal value of the objective function in (CGDP).*

*3.2. MINLP Reformulations*

In order to take advantage of the existing MINLP solvers, GDPs are often reformulated as an MINLP by using either the big-M (BM) (Nemhauser and Wolsey, 1988), or the Hull Relaxation (HR) (Lee and Grossmann, 2000). The former yields:

$$
\begin{array}{ll}
\min & Z = f(x) + \sum_{i\in D_k}\sum_{k\in K}\gamma_{ik}\lambda_{ik} \\
s.t. & r(x) \le 0 \\
 & g_{ik}(x) \le M(1-\lambda_{ik}) \quad i\in D_k,\ k\in K \qquad (BM) \\
 & \sum_{i\in D_k}\lambda_{ik} = 1 \qquad k\in K \\
 & A\lambda \ge a \\
 & x^{lo} \le x \le x^{up} \\
 & x\in R^n, \lambda_{ik}\in\{0,1\}\,,\, i\in D_k\ , k\in K
\end{array}
$$

where the variable $\lambda_{ik}$ has a one-to-one correspondence with the Boolean variable $Y_{ik}$. Note that when $\lambda_{ik}=0$ and the parameter $M$ is sufficently large, the associated constraint becomes redundant; otherwise, it is enforced. Also, $A\lambda\ge a$ is the reformulation of the logic constraints in the discrete space, which can be easily accomplished as described in the work by Williams (1985) and discussed in the work by Raman and Grossmann (1994). The convex hull reformulation yields,

$$
\begin{array}{ll}
\min & Z = f(x) + \sum_{i\in D_k}\sum_{k\in K}\gamma_{ik}\lambda_{ik} \\
s.t. & x = \sum_{i\in D_K}\nu^{ik} \qquad k\in K \\
 & r(x) \le 0 \\
 & (cl\, g'_{ik})(\nu^{ik},\lambda_{ik}) \le 0 \quad i\in D_k,\ k\in K \qquad (HR) \\
 & \lambda_{ik}x^{lo} \le \nu^{ik} \le \lambda_{ik}x^{up} \qquad i\in D_k,\ k\in K \\
 & \sum_{i\in D_k}\lambda_{ik} = 1 \qquad k\in K \\
 & A\lambda \ge a \\
 & x\in R^n,\, \nu^{ik}\in R^n, c_k\in R^1, \lambda_{ik}\in\{0,1\}\,,\, i\in D_k\,,\, k\in K
\end{array}
$$

where $(cl\, g'_{ik}) = \lambda_{ik}g_{ik}(\nu^{ik}/\lambda_{ik})$ if $\lambda_{ik}>0$ and $(cl\, g'_{ik}) = 0$ if $\lambda_{ik}=0$.

As it can be seen, the HR reformulation is less intuitive than 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 $\nu^{ik}$ and new constraints. On the other hand, as proved in Grossmann and Lee (2003) and discussed in Vecchietti, Lee and Grossmann (2003), the HR formulation is at least as tight and generally tighter than the BM when the discrete domain is relaxed (i.e. $0\le\lambda_{ik}\le 1,\ k\in K, i\in D_k$). This is of great importance considering that the efficiency of the MINLP solvers heavily relies on the strength of these relaxations.

Methods that have addressed the solution of problem (MINLP) include the branch and bound method (BB) (Gupta and Ravindran, 1985; Borchers and Mitchell, 1994), Generalized Benders Decomposition (GBD) (Geoffrion, 1972), Outer-Approximation (OA) (Duran and Grossmann, 1986; Fletcher and Leyffer, 1994), LP/NLP based branch and bound (Quesada and Grossmann, 1992; Bonami et al., 2008), and Extended Cutting Plane Method (ECP) (Westerlund and Pettersson, 1995; Westerlund and Poern, 2002).

REMARK 1.
In order to fully exploit the logic structure of GDP problems, two other solution methods have been proposed for the case of convex nonlinear GDP, namely, the Disjunctive Branch and Bound method (Lee and Grossmann, 2000), which builds on the Branch and Bound procedure by Beaumont (1991) and the Logic-Based Outer-Approximation method (Turkay and Grossmann, 1996). For an extensive comparison we refer the reader to Ruiz and Grossmann (2011).

REMARK 2.
In the MINLP reformulation method, it is clear that the (HR) reformulation proposed by Lee and Grossmann (2000) is equivalent to the result in the Theorem 2.3 extended for disjunctive sets not in DNF. As we pointed out, the relaxation of this formulation can be strengthened by applying a set of basic steps on the disjunctive set. In particular, if the objective function is nonlinear, we propose to add this as a constraint becoming part of the disjunctive set before the application of basic steps. In order to prevent the exponential increase of the size of the formulation, we propose to use the rules in section (2.6). Regarding the Logic Based Methods, these can be improved by considering the use of alternative generalized disjunctive programs that arise from the application of basic steps on the disjunctive set. Clearly, the master problem will be strengthened by the generation of stronger cuts.

## 4. Examples

### *4.1. Example 1: Simple Case (Circles2D3, Circles2D36, Circles3D36)*

This set of problems deals with the optimization of a convex quadratic function over a set of disjoint hyperspheres. The particular case Circles2D3 is taken from Lee and Grossmann (2000), which consists of the optimization of a two dimensional quadratic function over three disjoint circles. Similarly, the problem Circles2D36 is defined with 36 circles in 2-D and Circles3D36 with 36 circles in 3-D. In general these problems can be described through the following GDP formulation, where the disjunction corresponds to the selection of circles:

$$
\begin{aligned}
&\min Z = \textstyle\sum_i (x_i - a_i)^2 + c \\
&s.t. \\
&\bigvee_{j \in J} \begin{bmatrix} Y_j \\ \sum_i (x_i - b_i)^2 \leq d_i \\ c = \gamma_j \end{bmatrix} \qquad\qquad\qquad\qquad (CIRC) \\
&\bigvee_{j \in J} Y_j = True \\
&x_i \in R
\end{aligned}
$$

### 4.2. Example 2 : Process Network (Proc8, Proc10, Proc12)

In general, the underlying goal of a classical process synthesis problem consists in selecting the process (i.e. process units and mass flow network), that maximizes the profit when selling a product or set of products considering the cost of raw materials and the cost of the process units.

The model in the form of the GDP problem involves disjunctions for the selection of units, and propositional logic for the relationship of these units. Each disjunction contains the equation for each unit (these relax as convex inequalities). Proc8, Proc10, Proc12 are three instances with eight, ten and twelve unit operations, respectively. A general GDP model can be described as follows:

$$
\begin{aligned}
&Min\, Z = \textstyle\sum_{i \in I} c_i + \sum_{j \in J} p_j x_j + \gamma \\
&\text{s.t.} \\
&\textstyle\sum_{j \in J} r_{jn} x_j \leq 0 \quad \forall n \in N \\
&\begin{bmatrix} Y_i \\ \sum_j d_{ij}(e^{x_j/t_{ij}} - 1) - \sum_j s_{ij} x_j \leq 0 \\ c_i = \gamma_i \end{bmatrix} \vee \begin{bmatrix} \neg Y_i \\ x_j = 0, j \in J^i \\ c_i = 0 \end{bmatrix} \qquad\qquad (PROC) \\
&\Omega(Y) = \text{True} \\
&x_j,\, c_i \geq 0, Y_i = \{True, False\}
\end{aligned}
$$

### 4.3. Example 3: Farm land layout (Flay02, Flay03, Flay04)

These problems concern farm land layout, where one would like to determine the optimal length and width of a number of rectangular patches of land with fixed area, such that the perimeter of the set of patches is minimized. The nonlinearities in this set of problems arise from a set of hyperbolic constraints. A GDP formulation is given as follows (Sawaya, 2006), where the disjunctions are used to represent the non-overlapping of the rectangles:

$$
\begin{aligned}
&\min Z = 2(Length + Width) \\
&s.t. \\
&Length \geq x_i + L_i && i \in N \\
&Width \geq y_i + W_i && i \in N \\
&A_i/W_i - L_i \leq 0 && i \in N && (FLAY)
\end{aligned}
$$

$$
\begin{bmatrix} Z_{ij}^1 \\ x_i + L_i/2 \leq x_j - L_j/2 \end{bmatrix} \vee \begin{bmatrix} Z_{ij}^2 \\ x_j + L_j/2 \leq x_i - L_i/2 \end{bmatrix}
$$

$$
\vee \begin{bmatrix} Z_{ij}^3 \\ y_i + H_i/2 \leq y_j - H_j/2 \end{bmatrix} \vee \begin{bmatrix} Z_{ij}^4 \\ y_j + H_j/2 \leq y_i - H_i/2 \end{bmatrix} \forall i, j \in N, i < j
$$

$$
\begin{aligned}
&0 \leq Length \leq Length^U, 0 \leq Width \leq Width^U \\
&L_i^L \leq L_i \leq A_i/L_i^U, W_i^L \leq w_i \leq A_i/W_i^U \\
&0 \leq x_i \leq Length^U - L_i^L, 0 \leq y_i \leq Width^U - W_i^L \\
&Z_{ij}^1, Z_{ij}^2, Z_{ij}^3, Z_{ij}^4 \in \{True, False\} && \forall i, j \in N, i < j
\end{aligned}
$$

### *4.4. Example 4: Constrained Layout Problem (Clay0203, Clay0303, Clay0204)*

This problem was reported in Sawaya and Grossmann (2006). Non-overlapping units represented by rectangles must be placed within the confines of certain designated circular areas represented by non-linear constraints, such that the cost of connecting these units is minimized. The GDP model is as follows:

$$
\begin{aligned}
&\min Z = \sum_i \sum_j c_{ij}(delx_{ij} + dely_{ij}) \\
&s.t. \\
&delx_{ij} \geq x_i - x_j && \forall i, j \in N, i < j && (CLAY) \\
&delx_{ij} \geq x_j - x_i && \forall i, j \in N, i < j \\
&dely_{ij} \geq y_i - y_j && \forall i, j \in N, i < j \\
&dely_{ij} \geq y_j - y_i && \forall i, j \in N, i < j
\end{aligned}
$$

$$
\begin{bmatrix} Y_{ij}^1 \\ x_i + L_i/2 \leq x_j - L_j/2 \end{bmatrix} \vee \begin{bmatrix} Y_{ij}^2 \\ x_j + L_j/2 \leq x_i - L_i/2 \end{bmatrix}
$$

$$
\vee \begin{bmatrix} Y_{ij}^3 \\ y_i + H_i/2 \leq y_j - H_j/2 \end{bmatrix} \vee \begin{bmatrix} Y_{ij}^4 \\ y_j + H_j/2 \leq y_i - H_i/2 \end{bmatrix} \forall i, j \in N, i < j
$$

$$\bigvee_{k \in K} \begin{bmatrix} W_{ik} \\ (x_i - L_i/2 - x_k)^2 + (y_i + H_i/2 - y_k)^2 \leq r_k^2 \\ (x_i - L_i/2 - x_k)^2 + (y_i - H_i/2 - y_k)^2 \leq r_k^2 \\ (x_i + L_i/2 - x_k)^2 + (y_i + H_i/2 - y_k)^2 \leq r_k^2 \\ (x_i + L_i/2 - x_k)^2 + (y_i - H_i/2 - y_k)^2 \leq r_k^2 \end{bmatrix}$$

$$\begin{aligned} & x_i \leq UB_i^1 && \forall i \in N \\ & x_i \geq LB_i^1 && \forall i \in N \\ & y_i \leq UB_i^2 && \forall i \in N \\ & y_i \geq LB_i^2 && \forall i \in N \end{aligned}$$

$$delx_{ij}, dely_{ij} \in R_+^1, Y_{ij}^1, Y_{ij}^2, Y_{ij}^3, Y_{ij}^4, W_{ik} \in \{True, False\} \;\; \forall i, j \in N, i < j$$

## 5. Numerical Results

In this section we present a comparison of the solution methods applied to traditional formulations of GDP problems (i.e. BM and HR formulations) and the ones applied with the proposed formulations that involve the application of basic steps. Since the proposed framework allows the generation of many different relaxations, even when only using the rules presented in section 2.5 to guide the application of basic steps, for reproducibility purposes, in all the examples we intersect only the improper disjunctions given by the global constraints with the proper disjunctions. Note that every global constraint is an improper disjunction. For the case of a program with a nonlinear objective function we first rewrite it as an equivalent disjunctive program with a linear objective before applying the basic steps. We finally analyze the effect of the solution methods by using a branch and bound solver (SBB) using CONOPT (v 3.14) as the NLP subsolver. In all cases we used a value of $\epsilon = 1 \times 10^{-8}$ and a constant large Big-M parameter, $1 \times 10^5$. The models were implemented in GAMS (Brooke et al., 1998) and solved on a Pentium(R) CPU 3.40GHz and 1GB of RAM. In Table 1 we describe the characteristics of the GDP formulations and in Table 2 we show the size of the MINLP reformulations using the big-M, hull relaxation and proposed approach.

Table 3 presents a comparison of the performance of SBB using different formulations. As it can be seen from Table 3 the HR approach predicts tighter lower bounds than the BM approach for problems Proc8, Proc10, Proc12, Circles2D3, Circles2D36 and Circles3D36. This is in agreement with the results in the work of Lee and Grossmann (2000) where they showed that HR formulations are at least as tight as, if not tighter, than BM formulations. However, only Proc8, Proc10 and Proc12 benefit from faster computational times. On the other hand the proposed approach finds tighter lower bounds than the HR and BM approaches in all instances leading in most cases to significant improvements in the computational time required. The proposed approach finds the solution faster in 10 out of 12 problems when compared with the BM and in 8 out of 12 problems when compared with

Table 1: Size and characteristics of the examples formulated as GDP

| Example | Cont. Vars. | Boolean Vars. | Logic Const. | Disj. Const. | Global. Const. |
|---|---|---|---|---|---|
| Circles2D3 | 8 | 3 | 1 | 1 | 0 |
| Circles2D36 | 39 | 36 | 1 | 1 | 0 |
| Circles3D36 | 40 | 36 | 1 | 1 | 0 |
| Proc8 | 42 | 8 | 22 | 8 | 11 |
| Proc10 | 51 | 10 | 10 | 10 | 12 |
| Proc12 | 57 | 12 | 12 | 12 | 12 |
| Flay02 | 15 | 4 | 1 | 1 | 6 |
| Flay03 | 27 | 12 | 3 | 3 | 9 |
| Flay04 | 43 | 24 | 6 | 6 | 12 |
| Clay0203 | 31 | 18 | 6 | 6 | 12 |
| Clay0303 | 34 | 21 | 6 | 6 | 12 |
| Clay0204 | 53 | 32 | 10 | 10 | 24 |

Table 2: Size of different reformulations

| | BM Approach | | | HR Approach | | | Proposed Approach | | |
|---|---|---|---|---|---|---|---|---|---|
| Example | Bin | Con | Const | Bin | Con | Const | Bin | Con | Const |
| Circles2D3 | 3 | 8 | 12 | 3 | 16 | 20 | 3 | 20 | 27 |
| Circles2D36 | 36 | 39 | 38 | 36 | 111 | 112 | 36 | 147 | 184 |
| Circles3D36 | 36 | 40 | 38 | 36 | 148 | 149 | 36 | 184 | 221 |
| Proc8 | 8 | 42 | 97 | 8 | 98 | 152 | 8 | 444 | 843 |
| Proc10 | 10 | 51 | 98 | 10 | 124 | 158 | 10 | 638 | 1181 |
| Proc12 | 12 | 57 | 114 | 12 | 137 | 184 | 12 | 805 | 1462 |
| Flay02 | 4 | 15 | 12 | 4 | 47 | 52 | 4 | 55 | 80 |
| Flay03 | 12 | 27 | 25 | 12 | 123 | 145 | 12 | 195 | 334 |
| Flay04 | 24 | 43 | 43 | 24 | 235 | 283 | 24 | 511 | 865 |
| Clay0203 | 18 | 31 | 55 | 15 | 88 | 130 | 15 | 160 | 316 |
| Clay0303 | 21 | 34 | 67 | 21 | 100 | 424 | 21 | 268 | 571 |
| Clay0204 | 32 | 53 | 91 | 32 | 165 | 235 | 32 | 641 | 1503 |

Table 3: Performance of SBB using different formulation strategies

| | | BM Approach | | | HR Approach | | | Proposed Approach | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Example | Opt. | LB | Nds | T(s) | LB | Nds | T(s) | LB | Nds | T(s) |
| Circles2D3 | 1.17 | 0.00 | 4 | 1.0 | 1.15 | 4 | 1.04 | 1.17 | 0 | 0.7 |
| Circles2D36 | 2.25 | 0.00 | 70 | 7.8 | 0.00 | 70 | 15.40 | 2.24 | 29 | 4.9 |
| Circles3D36 | 15.77 | 0.44 | 70 | 7.3 | 12.04 | 70 | 18.50 | 15.72 | 34 | 10.8 |
| Proc8 | 68.01 | -829.0 | 34 | 4.6 | 67.12 | 2 | 1.0 | 68.01 | 0 | 1.7 |
| Proc10 | -73.51 | -1,108.88 | 197 | 21.7 | -78.81 | 4 | 1.0 | -73.56 | 2 | 1.9 |
| Proc12 | -69.51 | -1,108.88 | 234 | 27.7 | -74.81 | 8 | 1.0 | -69.51 | 2 | 2.9 |
| Flay02 | 37.95 | 28.28 | 6 | 1.0 | 28.28 | 6 | 1.7 | 37.95 | 3 | 1.0 |
| Flay03 | 48.99 | 30.98 | 104 | 10.7 | 30.98 | 108 | 12.1 | 41.94 | 30 | 9.0 |
| Flay04 | 54.40 | 30.98 | 2,415 | 234.0 | 30.98 | 2,887 | 288.0 | 41.69 | 52 | 48.0 |
| Clay0203 | 41,573.30 | 0.00 | 323 | 32.7 | 0.00 | 216 | 22.0 | 3,010.00 | 206 | 28.7 |
| Clay0303 | 26,670.00 | 0.00 | 380 | 42.0 | 0.00 | 879 | 99.0 | 3,103.00 | 331 | 69.0 |
| Clay0204 | 6,545.00 | 0.00 | 2,265 | 229.0 | 0.00 | 2,835 | 507.0 | 4,760.00 | 546 | 157.0 |

the HR. A reduction in the number of nodes and an increase in the lower bounds in all the problems are a direct indication that the proposed approach leads to tighter formulations. Also, it is important to note that the lower bounds obtained by following the proposed rules are close to the ones that we could obtain from the tightest relaxation (i.e. relaxation of disjunctive set in DNF), but avoiding the exponential increase in the size of the formulation. Evidence of this can be found in the cases of Proc8, Proc10, Proc12 and Flay02 where the lower bounds predicted are exactly the lower bounds that we would obtain in the relaxation of the DNF form. Finally, it is important to note that for the case of Circle2D3 the solution is found at the root node as predicted by Theorem 2.6. Also note that for the similar cases, Circle2D36 and Circle3D36, the solver has to branch a few nodes to compensate the error that arises from the approximation parameter $\epsilon$. However, for $\epsilon \to 0$, the number of nodes tends to zero.

## 6. Conclusions

We have proposed a framework for generating alternative mixed-integer nonlinear formulations for convex disjunctive programs that lead to stronger relaxations. We have defined an operation equivalent to a basic step for linear disjunctive programs that takes a disjunctive set to another with fewer number of conjuncts. We have shown that the strength of the relaxations increases as the number of conjuncts decreases, leading to a hierarchy of relaxations. We proved that the tightest of these relaxations, when coupled with reformulating the objective function as a constraint, allows in theory the solution of the convex disjunctive program by solving a single continuous nonlinear programming problem. We have presented a methodology to guide the generation of strong relaxations without incurring in an exponential increase of the size of the reformulated mixed-integer program. Finally, we have applied the proposed theory to improve the computational efficiency of solution methods for convex generalized disjunctive programs. This methodology was validated through a set of numerical examples showing an average of 75% improvements in the prediction of the lower bound. It is important to note that the set of proposed rules does not lead to a unique implementation in finding relaxations. Even though the implementation proposed in this paper has shown promising results, further improvements in the relaxations may be expected in future works (e.g. by considering basic steps between proper disjunctions).

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

## References

Abhishek, K., Leyffer, S., and Linderoth, J. (2006). *FilMINT: An Outer-Approximation-Based Solver for Nonlinear Mixed Integer Programs.* ANL/MCS-P1374-0906, Argonne National Laboratory.

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

Balas, E. (1985). Disjunctive programming and a hierarchy of relaxations for discrete optimization problems. *SIAM J. Alg. Disc. Meth.*, 6:466–486.

Beaumont, N. (1991). An algorithm for disjunctive programs. *European Journal of Operations Research*, 48:362–371.

Bonami, P., Biegler, L., Conn, A., Cornuejols, G., Grossmann, I., Laird, C., Lee, J., Lodi, A., Margot, F., Sawaya, N., and A., W. (2008). An algorithmic framework for convex mixed integer nonlinear programs. *Discrete Optimization*, 5:186–204.

Borchers, B. and Mitchell, J. (1994). An improved branch and bound algorithm for mixed integer nonlinear programming. *Computers and Operations Research*, 21:359–367.

Brooke, A., Kendrick, D., Meeraus, A., and Raman, R. (1998). *GAMS, a User's Guide*. IMA, In press.

Ceria, S. and Soares, J. (1999). Convex programming for disjunctive convex optimization. *Math Programming*, 86:595–614.

Duran, M. and Grossmann, I. (1986). An outer-approximation algorithm for a class of mixed-integer nonlinear programs. *Math Programming*, 36:307.

Fletcher, R. and Leyffer, S. (1994). Solving mixed integer nonlinear programs by outer-approximation. *Math Programming*, 66:327.

Geoffrion, A. (1972). Generalized benders decomposition. *JOTA*, 10:237–260.

Gupta, O. and Ravindran, V. (1985). Branch and bound experiments in convex nonlinear integer programming. *Management Science*, 31(12):1533–1546.

Lee, S. and Grossmann, I. (2000). New algorithms for nonlinear generalized disjunctive programming. *Computers and Chemical Engineering*, 24:2125–2141.

Nemhauser, G. and Wolsey, L. (1988). *Integer and Combinatorial Optimization*. Wiley-Interscience, New York.

Quesada, I. and Grossmann, I. (1992). An lp/nlp based branch and bound algorithm for convex minlp optimization problems. *Computers and Chemical Engineering*, 16:937–947.

Raman, R. and Grossmann, I. (1994). Modelling and computational techniques for logic-based integer programming. *Computers and Chemical Engineering*, 18:563.

Ruiz, J. and Grossmann, I. (2010). Strengthening the lower bounds for bilinear and concave gdp problems. *Computers and Chemical Engineering*, 34(3):914–930.

Ruiz, J. and Grossmann, I. (2011). *Generalized Disjunctive Programming: A Framework for formulation and alternative algorithms for MINLP optimization*. IMA, In press.

Sahinidis, N. (1996). Baron: A general purpose global optimization software package. *Journal of Global Optimization*, 8(2):201–205.

Sawaya, N. (2006). *Thesis: Reformulations, relaxations and cutting planes for generalized disjunctive programming.* Carnegie Mellon University, Pittsburgh, US.

Stubbs, R. and Mehrotra, S. (1999). A branch-and-cut method for 0-1 mixed convex programming. *Math Programming*, 86(3):515–532.

Sherali and Adams, (2009). A Reformulation-Linearization Technique (RLT) for semi-infinite and convex programs under mixed 0-1 and general discrete restrictions. *Discrete Applied Mathematics*, 157(6):1319–1333.

Turkay, M. and Grossmann, I. (1996). A logic-based outer-approximation algorithm for minlp optimization of process flowsheets. *Computers and Chemical Enginering*, 20:959–978.

Viswanathan and Grossmann, I. (1990). A combined penalty function and outer-approximation method for minlp optimization. *Computers and Chemical Enginering*, 14:769–782.

Westerlund, T. and Pettersson, F. (1995). A cutting plane method for solving convex minlp problems. *Computers and Chemical Enginering*, 19:S131–S136.

Westerlund, T. and Poern, R. (2002). Solving pseudo-convex mixed integer optimization problems by cutting plane techniques. *Optimization and Engineering*, 3:253–280.

Williams, H. (1985). *Mathematical Building in Mathematical Programming.* John Wiley.