SIMPLE EXTENDED FORMULATION FOR THE DOMINATING SET POLYTOPE VIA FACILITY LOCATION

MOURAD BAİOU AND FRANCISCO BARAHONA

ABSTRACT. In this paper we present an extended formulation for the dominating set polytope via facility location. We show that with this formulation we may describe the dominating set polytope for some class of graphs as cacti graphs, though its description in the natural node variables dimension has been only partially obtained. Moreover, the inequalities describing this polytope have coefficients in \{-1, 0, 1\}. This is not the case for the dominating set polytope in the node-variables dimension. It is known from [9] that for any integer \(p\), there exists a facet defining inequality having coefficients in \(\{1, \ldots, p\}\). We also show a decomposition theorem by means of 1-sums. Again this decomposition is much simpler with the extended formulation than with the node-variables formulation given in [10]. We also give a linear time algorithm to solve the minimum dominating set problem in cacti graphs.

1. Introduction

Let \(G = (V, A)\) be a directed graph, not necessarily connected, where each arc and each node has a cost (or a profit) associated with it. Consider the following version of the uncapacitated facility location problem (UFLP), where each location \(v \in V\) has a weight \(w(v)\) that corresponds to the revenue obtained by opening a facility at that location, minus the cost of building this facility. Each arc \((u, v) \in A\) has a weight \(w(u, v)\) that represents the revenue obtained by assigning the customer \(u\) to the opened facility at location \(v\), minus the cost originated by this assignment. The goal is to select some nodes where facilities are opened, and the non selected nodes might be assigned in such a way that the overall profit is maximized. This version of the UFLP is called the prize-collecting uncapacitated facility location problem (pc-UFLP). The following is a linear programming relaxation of the pc-UFLP.

\[
\begin{align*}
& \max \sum_{(u, v) \in A} w(u, v)x(u, v) + \sum_{v \in V} w(v)y(v) \\
& \sum_{(u, v) \in A} x(u, v) + y(u) \leq 1 \quad \forall u \in V, \\
& x(u, v) \leq y(v) \quad \forall (u, v) \in A, \\
& x(u, v) \geq 0 \quad \forall (u, v) \in A, \\
& y(v) \geq 0 \quad \forall v \in V,
\end{align*}
\]

For each node \(u\), the variable \(y(u)\) takes the value 1 if the node \(u\) is selected and 0 otherwise. For each arc \((u, v)\) the variable \(x(u, v)\) takes the value 1 if \(u\) is assigned to \(v\) and 0 otherwise. Inequalities (2) express the fact that either node \(u\) can be selected or

\textbf{Date}: November 7, 2012.

\textbf{Key words and phrases}. uncapacitated facility location problem, dominating set problem, lifted g-odd cycle inequalities, stable set problem.
it can be assigned to another node. Inequalities (3) indicate that if a node \( u \) is assigned to a node \( v \) then this last node should be selected.

Let \( P(G) \) be the polytope defined by (2)-(4), and let \( UFLP'(G) \) be the convex hull of \( P(G) \cap \{0,1\}^{|V|+|A|} \). Clearly

\[
UFLP'(G) \subseteq P(G).
\]

Given a directed graph \( G = (V, A) \), a subgraph induced by the nodes \( v_1, \ldots, v_r \) of \( G \) is called a \textit{bidirected cycle} if the only arcs in this induced subgraph are \((v_i, v_{i+1})\) and \((v_{i+1}, v_i)\), for \( i = 1, \ldots, r \), with \( v_{r+1} = v_1 \). We denote it by \( BIC_r \). The first part of this paper is devoted to the study of \( UFLP'(G) \). First we assume that \( G \) itself is a bidirected cycle. At first sight, the description of \( UFLP'(BIC_n) \) seems easy because of the simple structure of \( BIC_n \). We will show that we need to add the so-called \textit{lifted g-odd cycle inequalities}, to complete its description. These inequalities define facets of \( UFLP'(BIC_n) \), and are valid for \( UFLP'(G) \) for any graph \( G \). We also give a linear time algorithm to separate these inequalities.

To complete the description of \( UFLP'(G) \) in a more general class of graphs, we consider the graphs \( G = (V, A) \) that decompose by means of 1-sum, that is \( G \) may be decomposed into two graphs \( G_1 = (V_1, A_1) \) and \( G_2 = (V_2, A_2) \), with \( V = V_1 \cup V_2 \), \( V_1 \cap V_2 = \{u\} \) and \( A_1 \cup A_2 = A \) and \( A_1 \cap A_1 = \emptyset \). We describe \( UFLP'(G) \) using the descriptions of \( UFLP'(G_1') \) and \( UFLP'(G_2') \), where \( G_1' \) and \( G_2' \) are two auxiliary graphs defined, respectively, from \( G_1 \) and \( G_2 \). As a consequence we obtain a complete description of \( UFLP'(G) \) when \( G \) may be decomposed as 1-sums of bidirected cycles.

In the second part of this paper we discuss the consequences of these results when applied to the dominating set problem. More precisely, let \( G = (V, E) \) be an undirected graph. A subset \( D \cap V \) is called a \textit{dominating set} if every node of \( V \cap D \) is adjacent to a node of \( D \). The \textit{minimum weight dominating set problem} (MDWSP) is to find a dominating set \( D \) that minimizes \( \sum_{v \in V} w(v) \), where \( w(v) \) is a weight associated with each node \( v \in V \). The natural linear relaxation of the MDWSP is defined by the linear program below

\[
\begin{align*}
\text{min} & \quad \sum_{v \in V} w(v)x(v) \\
\text{subject to} & \quad x(N[v]) \geq 1 \quad \forall v \in V, \\
& \quad x(v) \geq 0 \quad \forall v \in V, \\
& \quad x(v) \leq 1 \quad \forall v \in V,
\end{align*}
\]

where \( N[v] \) denotes the set of neighbors of \( v \) including it. Define \( DSP(G) \) to be the convex hull of the integer vectors satisfying (7)-(9).

The MDWSP is a special case of the set covering problem. It is NP-hard even when all the weights are equal to 1, this may be shown using a simple reduction from the vertex cover problem. A large literature is devoted to this case and many of its variants, for a deep understanding of the subject we refer to [26, 25]. It has been shown that when the weights are all equal to 1, the MDWSP is solvable in many classes of graphs, a non-exhaustive list is cactus graphs [30], trees [30], series-parallel graphs [27], permutation graphs [11, 12, 16, 24], cocomparability graphs [28], (see chapter 2 in [25] for more classes). For the weighted case of the MDWDSP we have a short list of graphs where this problem may be solved in polynomial time, for cycles [8] and for strongly chordal graphs [23]. Little is known from the point of view of polyhedral approach and particularly few complete characterizations of the polytope associated with the MDWSP are known. For
the case of strongly chordal graph Farber [23] gives a primal-dual algorithm to solve the MWDSP this shows that $DSP(G)$ is defined by (7)-(9). The polytope $DSP(G)$ has been, first, characterized for cycle graphs in [8] and later published in [9]. This result has also been established in [34] using a different approach. One may also use the results related to the set covering polytope [18, 5, 17, 33], to cite a few, to establish new results for the MWDSP. The set covering polytope is the convex hull of \( \{ x \in \mathbb{R}^n : Ax \geq 1, x \in \{0,1\}^n \} \), where $A$ is an $m \times n$ matrix with 0,1 entries. For example, the polytope $DSP(G)$ when $G$ is a cycle with $n$ nodes coincide with the set covering polytope when $A$ is the $C_n^3$ circulant matrix. Recently in [7] a complete description of the set covering polytope is established when $A$ is the circulant matrix $C_{2k}^k$ or $C_{3k}^k$, $k \geq 3$.

We give an extended formulation via facility location to completely characterize the $DSP(G)$ when $G$ is a cactus. This description has been studied in the original dimension that is $\mathbb{R}^{\vert V\vert}$ in [8, 9]. They developed several facet defining inequalities for this case, and showed that this polytope has a more complicated structure than the case when $G$ is a cycle. Even with the 1-sum composition developed in [10], the complete characterization of $DSP(G)$ in cactus graphs has not been found. The main difficulty reported in [8, 9] is the description of the polytope when restricted to the auxiliary graphs obtained after the decomposition. In our work we show that with the extended formulation this task is easy and allows us to completely describe this polytope in a higher dimension. Moreover in [8, 9], it has been shown that for any fixed integer $p$, there exist a cactus $G$ such that $DSP(G)$ has a facet defining inequality with coefficients $1, \ldots, p$. In our description all the facets defining inequalities have coefficients in \{0, −1, +1\}.

This paper is organized as follows. In Section 2 we give some definitions and notations. In Section 3 we address some of the facets and valid inequalities of the facility location polytope. In Section 4 we study bidirected cycles. The dominating set polytope is studied in Section 5. The algorithmic consequences are studied in Section 6. Section 7 contains some concluding remarks.

2. Definitions and Notations

The nodes of a bidirected cycle $BIC_n$ will be denoted by $1, \ldots, n$, and the arcs will be $(i, i + 1)$ and $(i + 1, i)$, for $i = 1, \ldots, n$. When we use numbers $i + j$ or $i − j$, $i, j \in \{1, n\}$, the positive numbers are taken modulo $n$ and the negative ones are taken modulo $−n$. The number zero represents the node $n$. A bidirected path $P$ of a graph $G = (V, A)$ is a sequence of nodes $\{1, 2, \ldots, k\}$, with $(i, i + 1)$ and $(i + 1, i)$ both in $A$, for $i = 1, \ldots, k − 1$. The size of $P$ is $k − 1$.

Given a directed graph $G = (V, A)$ its intersection graph, denoted by $I(G)$, is defined as follows. A node in $I(G)$ is associated for each arc of $G$, and for $(u, v)$ and $(w, x)$ in $A$, their corresponding nodes are adjacent if $u = w$, or $v = w$, or $x = u$. It is easy to see that $I(BIC_n)$ is a circulant graph $G = (W, E)$, where $W = \{a_1, \ldots, a_{2n}\}$ and $E$ consists of the edges $\{a_i, a_{i+1}\}$ and $\{a_i, a_{i+2}\}$, for $i = 1, \ldots, 2n$, the indices are taken modulo $2n$.

Given an undirected graph $G = (V, E)$. A subset $S \subseteq V$ is called stable if there is no edge between any pair of nodes of $S$. The convex hull of the incidence vectors of the stable sets in $G$ is called the stable set polytope and is denoted by $SSP(G)$. When each node $v \in V$ is associated with a weight $w(v)$, the maximum weight stable set problem (MWSSP) is to find a stable set $S \subseteq V$ maximizing $\sum_{v \in S} w(v)$. A set $K \subseteq V$ is called a clique if there is an edge between every pair of nodes in $K$. 


Let $G = (V, A)$ be a directed graph. For $S \subseteq U$, we denote by $\delta^+(S)$ the set of arcs $(u, v) \in A$ with $u \in S$ and $v \in V \setminus S$. For a node $v \in V$ we write $\delta^+(v)$ instead of $\delta^+(\{v\})$. If there is a risk of confusion we use $\delta^+_C$.

For a polyhedron $P$ we say that an inequality $ax \leq b$ is valid for $P$, if it is satisfied by every vector in $P$. If $P$ is a polytope with extreme points in $\{0, 1\}^n$, and $ax \leq b$ is an inequality valid for $P$, we call it a rank inequality if all the coefficients of $a$ are in $\{0, 1\}$.

If $P \subseteq \mathbb{R}^n$ is a polyhedron and it contains $n + 1$ affinely independent points, we say that $P$ is full dimensional. If $ax \leq b$ is valid for $P$, and there are $n$ affinely independent vectors of $P$ that satisfy it with equation, we have that $ax \leq b$ defines a facet of $P$. If $P = \{x|Ax \leq b\}$ is a full dimensional polyhedron, and each inequality of $Ax \leq b$ defines a different facet, then this is a minimal system that defines $P$.

For a ground set $X$, a set $S \subseteq X$ and a function $f$ from $X$ to $\mathbb{R}$, we use $f(S)$ to denote $f(S) = \sum_{a \in S} f(a)$.

3. Facets and valid inequalities for $UFLP^r(G)$

In this section we give a family of valid inequalities for $UFLP^r(G)$, and we show their Gomory-Chvátal derivation, cf. [15]. We say that an inequality has Gomory-Chvátal rank one, if it can be derived with just one application of this procedure. We will also show that they define facets for $UFLP^r(G)$ when $G = BIC_n$. All these inequalities will be useful later when characterizing $UFLP^r(G)$ for cactus graphs. Before let us give the following remark that will be used implicitly along the paper.

Remark 1. $UFLP^r(G)$ is full dimensional.

3.1. The bidirected cycle inequalities. Let $G = (V, A)$ be any directed graph. Let $BIC_r$ a bidirected cycle in $G$ of size $r$. It may be easily seen that the inequality

$$(10) \quad \sum_{a \in A(BIC_r)} x(a) \leq \left\lfloor \frac{2|r|}{3} \right\rfloor,$$

is valid for $UFLP^r(G)$. This inequality is called the bidirected cycle inequality. It has been introduced in [1] and has an analogue for the stable set polytope [36], which is a rank inequality. The Gomory-Chvátal derivation is as follows. We multiply each inequality (2) by $\frac{2}{3}$, each inequality (3), with respect to the arcs in $A(BIC_r)$, by $\frac{1}{3}$, and finally combine with the trivial inequalities (4) to remove the variables with respect to the arcs not in $A(BIC_r)$. Then if we sum these inequalities we obtain an inequality with left-hand side $x(A(BIC_r))$ and right-hand side $\frac{2|r|}{3}$. After rounding down the right-hand side we obtain (10). So it has Gomory-Chvátal rank one.

Theorem 2. Assume that $G = BIC_n$. The bidirected cycle inequality (10) defines a facet of $UFLP^r(G)$ if and only if $n = 3k + 1$.

Proof. The proof consists of three cases.

- If $n = 3k$, then there are only three vectors that satisfy (10) with equation. Since this is a full dimensional polytope, we need at least $3n$ vectors.
- If $n = 3k + 2$. There are $2n$ vectors that satisfy (10) with equation.
- Consider now the case when $n = 3k + 1$. Assume that any vector that satisfies (10) with equation also satisfies

$$(11) \quad \alpha x + \beta y \leq \gamma,$$
with equation, where (11) defines a facet of $UFLP'(G)$.

There is a vector $(\tilde{x}, \tilde{y})$ that satisfies (10) with equation, $\tilde{x}(i, i+1) = \tilde{x}(i+1, i) = \tilde{x}(i+1, i+2) = \tilde{x}(i+2, i+1) = \tilde{y}(i+1) = 0$, and $\tilde{x}(i, i-1) = 1$. We can set $\tilde{y}(i+1) = 1$ and the new vector also satisfies (10) with equation. This implies

$$\beta(i+1) = 0.$$  \hspace{1cm} (12)

If we set $\tilde{x}(i, i+1) = 1$ and $\tilde{x}(i, i-1) = 0$, this new vector also satisfies (10) with equation. This implies

$$\alpha(i, i+1) = \alpha(i, i-1).$$  \hspace{1cm} (13)

There is a vector $(\tilde{x}, \tilde{y})$ that satisfies (10) with equation, and $\tilde{x}(i-1, i) = \tilde{y}(i) = 1$ and $\tilde{x}(i-2, i-1) = \tilde{x}(i-1, i-2) = \tilde{x}(i, i+1) = \tilde{x}(i+1, i) = 0$. We can set $\tilde{x}(i-1, i) = \tilde{y}(i) = 0$ and $\tilde{x}(i, i-1) = \tilde{y}(i-1) = 1$, and this new vector also satisfies (10) with equation. This implies

$$\alpha(i-1, i) = \alpha(i, i-1).$$  \hspace{1cm} (14)

From (12), (13) and (14), we obtain

$$\beta(i) = 0,$$

$$\alpha(i, i+1) = \alpha(i+1, i) = \alpha(i+1, i+2) = \rho,$$

for all $i \mod n$, and

$$\gamma = \rho \left\lfloor \frac{2n}{3} \right\rfloor.$$  \hspace{1cm} \square

3.2. The lifted g-odd cycle inequalities. A simple cycle $C$ is an ordered sequence $v_0, a_0, v_1, a_1, \ldots, a_{p-1}, v_p,$

where

- $v_i$, $0 \leq i \leq p-1$, are distinct nodes,
- $a_i$, $0 \leq i \leq p-1$, are distinct arcs,
- either $v_i$ is the tail of $a_i$ and $v_{i+1}$ is the head of $a_i$, or $v_i$ is the head of $a_i$ and $v_{i+1}$ is the tail of $a_i$, for $0 \leq i \leq p-1$, and
- $v_0 = v_p$.

By setting $a_p = a_0$, we associate with $C$ three more sets as below.

- We denote by $\hat{C}$ the set of nodes $v_i$, such that $v_i$ is the head of $a_{i-1}$ and also the head of $a_i$, $1 \leq i \leq p$.
- We denote by $\hat{C}$ the set of nodes $v_i$, such that $v_i$ is the tail of $a_{i-1}$ and also the tail of $a_i$, $1 \leq i \leq p$.
- We denote by $\hat{C}$ the set of nodes $v_i$, such that either $v_i$ is the head of $a_{i-1}$ and also the tail of $a_i$, or $v_i$ is the tail of $a_{i-1}$ and also the head of $a_i$, $1 \leq i \leq p$.

Notice that $|\hat{C}| = |\hat{C}|$. A cycle will be called $g$-odd (generalized odd) if $p + |\hat{C}|$ (or $|\hat{C}| + |\hat{C}|$) is odd, otherwise it will be called $g$-even. A cycle $C$ with $\hat{C} = \emptyset$ is a directed cycle. The set of arcs in $C$ is denoted by $A(C)$.

Let $C$ be a g-odd cycle. Now we define the lifting set $\hat{A}(C)$ as follows. For each node $i \in \hat{C}$ we have two cases:
Figure 1. A g-odd cycle $C$. The black nodes are those of $\hat{C}$.

- If $i - 1$ and $i + 1$ are in $\hat{C}$, we pick arbitrarily one arc from $\{(i - 1, i), (i + 1, i)\}$ and add it to $\bar{A}(C)$.
- If only one of the neighbors of $i$ is in $\hat{C}$, say the node $j$. We add $(j, i)$ to $\bar{A}(C)$.

Once the set $\bar{A}(C)$ has been defined, a lifted g-odd cycle inequality has the form

$$\sum_{a \in A(C)} x(a) + \sum_{a \in \bar{A}(C)} x(a) - \sum_{v \in \hat{C}} y(v) \leq |\hat{C}| + |\hat{C}| - 1.$$

Notice that given a g-odd cycle $C$, we might have several lifting sets $\bar{A}(C)$, therefore we might have several lifted g-odd cycle inequalities.

Similar inequalities called lifted odd cycle inequalities have been studied in [32, 19, 13, 14].

Figure 2. The two lifted g-odd cycles that may be obtained from $C$ of Figure 1. The lifting set is represented by the bold arcs.

Lemma 3. The lifted g-odd cycle inequalities (15) are Gomory-Chvátal cuts of rank one for $UFLP'(G)$, for any graph.

Proof. Let $C$ a g-odd cycle and $\bar{A}(C)$ the lifting set. From inequalities (2)-(5) we obtain

$$\sum_{(u, v) \in A} x(u, v) + y(u) \leq 1, \text{ for every node } u \in \hat{C} \cup \check{C},$$

$$-x(u, v) \leq 0, \text{ for every arc } (u, v) \notin A(C) \cup \bar{A}(C), \text{ } u \in \hat{C} \cup \check{C},$$

$$-y(u) \leq 0, \text{ for every node } u \in \hat{C} \text{ and } \bar{A}(v, u) \in \bar{A}(C),$$

$$x(u, v) - y(v) \leq 0, \text{ for every arc } (u, v) \in A(C) \cup \bar{A}(C).$$
Their sum gives
\[ 2 \sum_{a \in A(C)} x(a) + 2 \sum_{a \in \tilde{A}(C)} x(a) - 2 \sum_{v \in \tilde{C}} y(v) \leq |\tilde{C}| + |\tilde{C}|, \]
dividing by 2 and rounding down the right-hand side we obtain inequality (15).

To visualize the feasible 0–1 vectors that satisfy (15) with equation, we build an undirected graph \( H = (U, E) \) as follows. For each node \( i \in \tilde{C} \cup \tilde{C} \) we add a node \( i' \) to \( U \). For each node \( i \in \tilde{C} \) we add the nodes \( i' \) and \( i'' \). For each arc \((i, j) \in A(C)\) with \( j \in \tilde{C} \), we add an edge \( i'j' \) to \( E \). If \( j \in \tilde{C} \) and the arcs \((i, j) \) and \((k, j) \) are in \( A(C) \) we add the edges \( i'j', j'j'' \) and \( j''k' \) to \( E \). Finally for each arc \((i, j) \in A(C)\) we add a parallel edge between \( i' \) and \( j' \). This last edge is parallel to the edge associated with \((j, i) \in A(C)\).

We have that \(|U| = |C| + |\tilde{C}|\), and \(|U|\) is odd. For each edge \( e \) associated with an arc \((i, j)\) we define \( z(e) = x(i, j) \). For each edge \( i'i'' \) associated with \( i \in \tilde{C} \), we define \( z(i'i'') = 1 - y(i) \).

Let \((\bar{x}, \bar{y})\) be a feasible 0–1 vector that satisfies (15) with equation and let \( \bar{z} \) be its associated vector defined as above. For each node \( u \in U \) we have
\[ \bar{z}(\delta(u)) \leq 1. \]
We also have
\[ \sum_{e \in E} \bar{z}(e) = \frac{|U| - 1}{2}. \]
Thus any feasible 0–1 vector that satisfies (15) with equation corresponds to a maximum matching in the graph \( H \). This interpretation is useful in the proof of the theorem below.

Now suppose that for each pair of parallel edges we keep one of them. Let \( M \) be the matrix whose rows are the incidence vectors of all maximal matchings. We have the following Lemma.

**Lemma 4.** The matrix \( M \) is nonsingular. If \( \gamma \) is a vector with all its components equal to a constant \( \theta \), then the solution of the system
\[ M\mu = \gamma \]
is \( \mu_i = \theta / k \) for all \( i \), where \( k = \frac{|U| - 1}{2} \).

**Proof.** We have an odd cycle with nodes \( v_1, \ldots, v_{2k+1} \) and edges \( v_i v_{i+1} \); here the indices are taken modulo \( 2k + 1 \). We order the rows and columns of \( M \) as follows.

- Let \( \{v_1v_2, v_3v_4, \ldots, v_{2k-1}v_{2k}\} \) be the first matching. These edges give the first set of columns, and this matching gives the first row.
- Next we shift the matching and take \( \{v_3v_4, \ldots, v_{2k-1}v_{2k}, v_{2k+1}v_1\} \). Then the next column corresponds to \( v_{2k+1}v_1 \), and this matching gives the second row.
- We repeat this shifting until the \( 2k + 1 \) matchings have been produced.

Then \( M \) is a circular matrix of size \( 2k + 1 \), with \( k \) consecutive ones per row. If we take the difference between the first and the second equation be have
\[ \mu_1 = \mu_{k+1}. \]
In a similar way we have
\[ \mu_i = \mu_{k+1}, \tag{16} \]
where the indices are taken modulo \( 2k + 1 \).
Since \( k \) and \( 2k + 1 \) are relatively prime, their least common multiple is \( k(2k + 1) \). Thus after using \( 2k + 1 \) times equation (16) we obtain

\[
\mu_i = \mu_1,
\]

for all \( i \). And from any of the equations we have

\[
\mu_i = \frac{\theta}{k}.
\]

This also shows that \( M \) is nonsingular. \( \Box \)

Now we can treat lifted \( g \)-odd cycle inequalities.

**Theorem 5.** All lifted \( g \)-odd cycle inequalities define facets of \( UFLP'(BIC_n) \).

**Proof.** Let us assume that any vector that satisfies (15) with equation also satisfies

\[
(17) \quad \alpha x + \beta y = \gamma,
\]

with equation, where (17) defines a facet.

First consider a node \( i \in \mathcal{C} \cup \bar{\mathcal{C}} \). There is a \( 0-1 \) vector \((\bar{x}, \bar{y})\) that satisfies (15) with equation and \( \bar{x}(i, i - 1) = \bar{x}(i, i + 1) = \bar{y}(i) = 0 \). We can set \( \bar{y}(i) = 1 \) and the new vector also satisfies (15) with equation. This implies

\[
(18) \quad \beta(i) = 0.
\]

Now consider an arc \((i, i + 1) \notin A(C) \cup \bar{A}(C) \). There are four cases.

- **i \in \mathcal{C} and i + 1 \in \bar{C}**. Here \((i + 2, i + 1) \in \bar{A}(C) \). There is a \( 0-1 \) feasible vector \((\bar{x}, \bar{y})\) that satisfies (15) with equation and \( \bar{x}(i + 2, i + 1) = \bar{y}(i + 1) = 1, \bar{y}(i) + \bar{x}(i, i + 1) + \bar{x}(i, i - 1) = 0 \). We can set \( \bar{x}(i, i + 1) = 1 \) and the new vector also satisfies (15) with equation. This implies \( \alpha(i, i + 1) = 0 \).
- **i \in \bar{C} and i + 1 \in \mathcal{C}**. There is a \( 0-1 \) feasible vector \((\bar{x}, \bar{y})\) that satisfies (15) with equation and \( \bar{y}(i + 1) + \bar{x}(i + 1, i) + \bar{x}(i + 1, i + 2) = 0, \bar{y}(i) + \bar{x}(i, i + 1) + \bar{x}(i, i - 1) = 0 \). We can set \( \bar{x}(i, i + 1) = \bar{y}(i + 1) = 1 \) and the new vector also satisfies (15) with equation. This implies \( \alpha(i, i + 1) = 0 \).
- **i \in \mathcal{C} and i + 1 \in \bar{C}**. There is a \( 0-1 \) feasible vector \((\bar{x}, \bar{y})\) that satisfies (15) with equation and \( \bar{y}(i) + \bar{x}(i, i + 1) + \bar{x}(i, i - 1) = 0, \bar{x}(i + 2, i + 1) = \bar{y}(i + 1) = 1 \). We can set \( \bar{x}(i, i + 1) = \bar{y}(i + 1) = 1 \) and the new vector also satisfies (15) with equation. This implies \( \alpha(i, i + 1) = 1 \).

Thus from all the preceding cases we have that \((i, i + 1) \notin A(C) \cup \bar{A}(C) \) implies

\[
(19) \quad \alpha(i, i + 1) = 0.
\]

Next consider an arc \((i, i + 1) \in \bar{A}(C) \). There is a \( 0-1 \) vector \((\bar{x}, \bar{y})\) that satisfies (15) with equation and \( \bar{x}(i, i + 1) = 1 \). We can set \( \bar{x}(i, i + 1) = \bar{y}(i + 1) = 0 \) and \( \bar{x}(i + 1, i) = \bar{y}(i) = 1 \) and the new vector also satisfies (15) with equation. This implies

\[
(20) \quad \alpha(i, i + 1) = \alpha(i + 1, i).
\]
Now consider all 0-1 vectors that satisfy
\[ \sum_{a \in A(C)} x(a) - \sum_{v \in \hat{C}} y(v) = \frac{|\hat{C}| + |\hat{C}| - 1}{2}. \]
Consider their restriction to \( A(C) \cup \hat{C} \). For each of these vectors \((x, y)\), define \( y'(i) = 1 - y(i) \) for all \( i \in \hat{C} \). The new vectors satisfy
\[ \sum_{a \in A(C)} x(a) + \sum_{v \in \hat{C}} y'(v) = \frac{|\tilde{C}| + 3|\hat{C}| - 1}{2} = \frac{|C| + |\hat{C}| - 1}{2}, \]
and
\[ \sum_{a \in A(C)} \alpha(a) x(a) + \sum_{v \in \hat{C}} \beta'(v) y'(v) = \gamma + \sum_{v \in \hat{C}} \beta'(v), \]
where \( \beta' = -\beta \). Now we look at \((\alpha, \beta')\) as variables, and from (21) we extract a system of equations like
\[ M \begin{bmatrix} \alpha \\ \beta' \end{bmatrix} = \gamma', \]
Since all components of \( \gamma' \) are \( \gamma + \sum_{v \in \hat{C}} \beta'(v) \), it follows from Lemma 4 that
\[ \alpha(a) = \rho \text{ for all } a \in A(C), \]
\[ -\beta(v) = \beta'(v) = \rho \text{ for all } v \in \hat{C}, \]
\[ \gamma = \rho \frac{|\tilde{C}| + |\hat{C}| - 1}{2}. \]
We should have \( \rho > 0 \), because (17) is valid. Therefore (17) is a positive multiple of (15).

\[ \square \]

4. The characterization of \( UFLP'(BIC_n) \)

In this section we give a minimal system of inequalities that defines \( UFLP'(BIC_n) \), namely the main result of this section is the following.

**Theorem 6.** \( UFLP'(BIC_n) \) is described by the constraints (2)-(5), the bidirected cycle inequality (10) with respect to \( BIC_n \) and the lifted g-odd cycle inequalities (15). Moreover, these inequalities describe a minimal system for \( UFLP'(BIC_n) \).

In order to prove this theorem, we need first several lemmas. Assume that
\[ \alpha x + \beta y \leq \rho \]
is a valid inequality defining a facet of \( UFLP'(BIC_n) \). Let
\[ F_{\alpha,\beta} = \{(x, y) \in UFLP'(BIC_n) \cap \{0, 1\}^{V+A} : \alpha x + \beta y = \rho \}. \]
We will show that (22) is one of the inequalities (2)-(5), (10) or (15). We assume in this section that (22) is not a positive multiple of (2)-(5) and (10). We will recall this when needed. In the proof we will implicitly use the following remark.

**Remark 7.** There exist always a 0-1 vector in \( F_{\alpha,\beta} \) that satisfies inequalities (2)-(5) as a strict inequalities (not necessarily at the same time). Otherwise (22) is one of the inequalities (2)-(5).
Also it easy easy to show the following remark, which hold for any graph and not specifically for \( BIC_n \).

**Remark 8.** \( \alpha(u, v) \geq 0 \) for each \( (u, v) \in A \); \( \beta(u) \leq 0 \) for each \( u \in V \); \( \rho \geq 0 \).

We have the following key lemma, its proof is the subject of the following sub-section.

**Lemma 9.** We have \( \alpha(u, v) \in \{0, 1\} \) for each \( (u, v) \in A(BIC_n) \), and \( \beta(u) \in \{0, -1\} \) for each \( u \in V(BIC_n) \).

4.1. **The proof of Lemma 9.** Let \( G(V, A) = BIC_n \). From \( G \) define the graph \( G' = (V', A') \) by adding a new node \( u' \) for each node \( u \in V(BIC_n) \), and the arc \( (u, u') \).

Consider the following linear system with respect to \( G' \)

\[
\begin{align*}
(23) & \quad \sum_{(u,v) \in A} x(u,v) + y(u) + x(u,u') = 1 \quad \forall u \in V, \\
(24) & \quad x(u,v) \leq y(v) \quad \forall (u,v) \in A, \\
(25) & \quad x(u,v) \geq 0 \quad \forall (u,v) \in A, \\
(26) & \quad y(v) \geq 0 \quad \forall v \in V, \\
(27) & \quad x(u,u') \geq 0 \quad \forall u \in V.
\end{align*}
\]

This system is the linear relaxation of the classical uncapacitated facility location problem (UFLP). We denote by \( UFLP(G') \) the convex hull of the 0-1 vectors satisfying (23)-(27). Notice the following remark.

**Remark 10.** Any valid inequality of \( UFLP(G') \) is also a valid inequality of \( UFLP(G') \).

And any valid inequality of \( UFLP(G') \), except the equations (23), may be written in the form of a valid inequality of \( UFLP(G') \) by eliminating the variables \( x(u,u') \) using equations (23).

Now consider the intersection graph of \( G' \), \( I(G') \). The figure below it shows the case when \( G = BIC_4 \) and \( I(G') \).

For convenience we rename the variables \( x(u,u') \) to \( s(u) \) and call them slack variables. The following linear system is obtained from (23)-(27) by eliminating the variables \( y(u) \) for \( u \in V \) with the use of inequalities (24). This is the projection of (23)-(27) onto the \( x \) and the \( s \) variables,

\[
\begin{align*}
(28) & \quad x(w, u) + \sum_{(u,v) \in A} x(u,v) + s(u) \leq 1 \quad \forall (w,u) \in A(BIC_n), \\
(29) & \quad x(u,v) \geq 0 \quad \forall (u,v) \in A(BIC_n), \\
(30) & \quad s(u) \geq 0, \quad \forall u \in V(BIC_n).
\end{align*}
\]

Notice that each inequality (28) corresponds to a maximal clique inequality with respect to \( I(G') \). The \( x \) variables are associated with the nodes of \( I(G') \) that correspond to the arcs of \( BIC_n \) and the slack variables are associated with the nodes of \( I(G') \) that correspond to the new arcs \( (u, u') \) for each \( u \in V(BIC_n) \).

The convex hull of the 0-1 solutions satisfying (28)-(30) is the stable set polytope \( SSP(I(G')) \). It is easy to check that \( I(G') \) is a quasi-line graph, that is the neighbors of each node may be decomposed into two cliques. We are going to use results about the stable set polytope of a quasi-line graph. We have a similar remark as above.
Remark 11. Any valid inequality of \( SSP(I(G')) \) is also a valid inequality of \( UFLP(G') \). And any valid inequality of \( UFLP(G') \), except the equations (23), may be written in the form of a valid inequality of \( SSP(I(G')) \), by eliminating the variables \( y(u) \) using equations (23).

For the stable set polytope of quasi-line graphs, Oriolo [31] proposed the valid inequalities below called *clique family inequalities*. Later in [21] it has been proved that these inequalities together with the maximal clique and nonnegativity, are sufficient to describe \( SSP(G) \) when \( G \) is a quasi-line graph.

Let \( \mathcal{F} \) be a family of \( n \) maximal cliques. Let \( 1 < p < n \) be an integer. Consider the following sets.

\[
I(p) = \{ v : |\{ C \in \mathcal{F} : v \in C \}| \geq p \}
\]

\[
O(p) = \{ v : |\{ C \in \mathcal{F} : v \in C \}| = p - 1 \}.
\]

Let \( r = n - [n/p]p \) and \( \lambda = (p - r - 1)/(p - r) \). Then a clique-family inequality is

\[
\sum_{v \in I(p)} x_v + \lambda \sum_{v \in O(p)} x_v \leq [n/p].
\]

If we study \( SSP(I(G')) \), the nodes of \( I(G') \) associated with the arcs of \( BIC_n \) appear in three maximal cliques, and the nodes associated to the slack variables appear in two maximal cliques. Thus in our case \( p \) takes the values 2 or 3.

If \( p = 2 \) we should have \( r = 1 \) and \( \lambda = 0 \), so we obtain an inequality with \( 0 - 1 \) coefficients. If \( p = 3 \) and \( r = 2 \) again \( \lambda = 0 \) and we obtain an inequality with \( 0 - 1 \) coefficients.

If \( p = 3 \) and \( r = 1 \), then \( \lambda = 1/2 \). If we multiply by 2 we obtain an inequality with coefficients \( 0, 1, 2 \). If a node \( v \) appears in three cliques, they should be the ones shown in Figure 4. Then the node \( w \) (it corresponds to the slack variable \( s(w) \)) appears in two cliques.
Figure 4. The three cliques are \{v, s, r, f\}, \{v, s, t, w\}, \{v, t, u, w\}. The nodes \(w\) and \(r\) are those associated with the slack variables \(s(w)\) and \(s(r)\).

If \(v\) represents the arc \((i, j)\) then the inequality has a coefficient 2 for \(x(i, j)\) and a coefficient 1 for \(s(i)\). This is the slack variable associated with \(i\). From Remark 11 this inequality is also valid for UFPLP\((G')\). From Remark 10, we may rewrite this inequality to obtain a valid inequality for UFPLP'\((G)\), we use the equation

\[
y(i) + x(i, j) + x(i, k) + s(i) = 1
\]

to eliminate the variable \(s(i)\). We obtain an inequality with coefficient \(-1\) for \(y(i)\) and coefficient 1 for \(x(i, j)\). The variable \(x(i, k)\) can have coefficient 0 or 1. By Remarks 10 and 11, any valid inequality of UFPLP'\((G)\) is also valid for SSP\((I(G'))\). This completes the proof Lemma 9.

4.2. The proof of Theorem 6. First we give a list of useful lemmas. They will be used in the discussion that completes the proof at the end of this subsection.

**Lemma 12.** We cannot have \(\alpha(u, v) = 1\) for all \((u, v) \in A(BIC_n)\) and \(\beta(u) = -1\) for all \(u \in V(BIC_n)\).

**Proof.** Assume that \(\alpha(u, v) = 1\) for all \((u, v) \in A(BIC_n)\) and \(\beta(u) = -1\) for all \(u \in V(BIC_n)\). Notice that

\[
\text{Max}_{(x,y) \in UFPLP'(BIC_n)} \left\{ \sum_{(u,v) \in A(BIC_n)} x(u,v) - \sum_{u \in V(BIC_n)} y(u) \right\} = \left\lfloor \frac{n}{3} \right\rfloor.
\]

In fact, notice that if \(y(i) = 1\), to have a positive contribution to the objective, we need \(x(i-1, i) = 1 = x(i+1, i)\). So we should have a configuration like in Figure 5.

Figure 5. The black nodes and bold arcs correspond to variables with value 1, the other variables are zero, except the nodes in the ends.
The maximum number of such configurations is $\left\lfloor \frac{n}{3} \right\rfloor$. It follows that

\begin{equation}
\left\lfloor \frac{n}{3} \right\rfloor \leq \rho.
\end{equation}

We will study three cases. If $n = 3k$, there are only 3 feasible 0-1 vectors that give the maximum of (31). If $n = 3k + 1$, there are only $n$ feasible 0-1 vectors that give this maximum of (31). Since we have a full dimensional polytope we need at least $3n$ vectors that satisfy the inequality as equation, to have a facet. Therefore, to finish the proof we need to study the case $n = 3k + 2$ differently, since in his case we have $3n$ vectors that give the maximum of (31).

Let $n = 3k + 2$. The nodes of $BIC_n$ are $1, \ldots, 3k, 3k + 1, 3k + 2$.

We define a cycle $C$ as follows. The set of nodes in $\hat{C}$ are $i = 2 + 3l$, for $l = 0, \ldots, k - 1$ and the node $3k + 1$. For each node $i \in \hat{C}$ let $(i - 1, i)$ and $(i + 1, i)$ in $A(C)$. To complete the cycle we add the arcs $(3k + 1, 2)$ and $(2, 1)$ for $i = 3l, l = 1, \ldots, k - 1$. It results that the nodes $i = (3k + 2) + 3l$ for $l = 0, \ldots, k$, are in $\hat{C}$ and the nodes $i = (3k + 2) + 3l + 1$ for $l = 0, \ldots, k - 1$, are in $\hat{C}$. Hence $|\hat{C}| = |\hat{C}| = k + 1$ and $|\hat{C}| = k$, so $C$ is a $g$-odd cycle. Define a lifting set as follows: with $\hat{A}(C) = \{(i + 1, i) : i = (3k + 2) + 3l, l = 0, \ldots, k - 1\}$. We have the following lifted g-odd cycle inequality,

\begin{equation}
\sum_{a \in \hat{A}(C)} x(a) + \sum_{a \in \hat{A}(C)} x(a) - \sum_{v \in \hat{C}} y(v) \leq \frac{|\hat{C}| + |\hat{C}| - 1}{2} = k = \left\lfloor \frac{n}{3} \right\rfloor.
\end{equation}

Inequalities (3) imply,

\begin{align}
\sum_{a \in A(C)} x(a) + \sum_{a \in A(C)} x(a) - \sum_{v \in C} y(v) & \leq \frac{|\hat{C}| + |\hat{C}| - 1}{2} = k = \left\lfloor \frac{n}{3} \right\rfloor,
\end{align}

\begin{align}
\sum_{a \in A(C)} x(a) + \sum_{a \in A(C)} x(a) - \sum_{v \in C} y(v) & \leq \frac{|\hat{C}| + |\hat{C}| - 1}{2} = k = \left\lfloor \frac{n}{3} \right\rfloor.
\end{align}

The sum of (33), (34) and (35) shows that (22) cannot define a facet of $UFLP'(BIC_n)$.

\begin{lemma}
Lemma 13. Let $i$ be a node of $BIC_n$ with $\beta(i) = -1$. Then $\alpha(i + 1, i) = \alpha(i - 1, i) = 1$.
\end{lemma}

\begin{proof}
Suppose that $\alpha(i - 1, i) = 0$ (the case when $\alpha(i + 1, i) = 0$ is analogue). We must have a solution $(x, y) \in F_{\alpha, \beta}$ with $x(i + 1, i) = 0$ and $y(i) = 1$. Since $\beta(i) = -1$, then the solution obtained from $(x, y)$ by replacing the values of $y(i)$ and $x(i - 1, i)$ by 0 would violate (22), which contradicts the fact that (22) is valid for $UFLP'(G)$.
\end{proof}

\begin{lemma}
Lemma 14. Let $i$ be a node of $BIC_n$ with $\beta(i) = -1$. If $\alpha(i, i - 1) = \alpha(i - 1, i) = 1$, then $\beta(i - 1) = -1$.
\end{lemma}

\begin{proof}
Assume this is not true, then $\beta(i - 1) = 0$. We must have $(x, y) \in F_{\alpha, \beta}$ with $x(i + 1, i) = 0$, $y(i) = 1$. This implies that $x(i - 1, i) = 1$, otherwise setting $y(i) = 0$ would yield a feasible solution that violates (22). Let $(x', y')$ the solution obtained from $(x, y)$ by setting $x'(i, i - 1) = 1$, $y'(i - 1) = 1$ and $x'(i - 1, i) = y'(i) = 0$, and $x'$ and $y'$ have the same values as $x$ and $y$ for the other variables. Then it is easy to check that $(x', y') \in UFLP'(BIC_n)$ and does not verify (22).
\end{proof}

\begin{lemma}
Lemma 15. Let $i$ be a node of $BIC_n$ with $\beta(i) = -1$. If $\alpha(i, i - 1) = 1$, then $\alpha(i, i + 1) = 1$.
\end{lemma}

\begin{proof}
Assume this is not true, then $\beta(i + 1) = 0$. We must have $(x, y) \in F_{\alpha, \beta}$ with $x(i + 1, i) = 0$, $y(i) = 1$. This implies that $x(i - 1, i) = 1$, otherwise setting $y(i) = 0$ would yield a feasible solution that violates (22). Let $(x', y')$ the solution obtained from $(x, y)$ by setting $x'(i, i - 1) = 1$, $y'(i - 1) = 1$ and $x'(i - 1, i) = y'(i) = 0$, and $x'$ and $y'$ have the same values as $x$ and $y$ for the other variables. Then it is easy to check that $(x', y') \in UFLP'(BIC_n)$ and does not verify (22).
\end{proof}
Lemma 16. Let \( \alpha(i, i - 1) = 1 \). From Lemma 13 we know that \( \alpha(i + 1, i) = \alpha(i - 1, i) = 1 \) and now using Lemma 14 we have \( \beta(i - 1) = -1 \).

Assume that we cannot have a solution \( (x, y) \in F_{\alpha, \beta} \) with \( x(i, i - 1) = 0 \), \( y(i - 1) = 1 \) and \( x(i - 2, i - 1) = 1 \). If \( (x, y) \in F_{\alpha, \beta} \), we know that we cannot have \( x(i, i - 1) = x(i - 2, i) = 0 \) and \( y(i - 1) = 1 \), since \( \beta(i - 1) = -1 \) and so setting \( y(i - 1) \) to zero would violate (22). It follows that any solution \( (x, y) \in F_{\alpha, \beta} \) must satisfy \( x(i, i - 1) = y(i - 1) \). But then (22) is implied by (4).

From the above discussion we may assume that there exists a solution \( (x, y) \in F_{\alpha, \beta} \) with \( x(i, i - 1) = 0 \), \( y(i - 1) = 1 \) and \( x(i - 2, i - 1) = 1 \).

If \( y(i) = 1 \), then we must have \( x(i + 1, i) = 1 \) since \( \beta(i) = -1 \). Now if we set \( x(i + 1, i) \) and \( y(i) \) to zero and \( x(i, i - 1) \) to one, we obtain a feasible solution that violates (22). Also if we have \( y(i) = 0 \) and \( x(i, i + 1) = 0 \), then setting \( x(i, i - 1) \) to one would violate (22). It follows that \( x(i, i + 1) = 1 \) and in this case we must have \( \alpha(i, i + 1) = 1 \). Otherwise, setting \( x(i, i + 1) \) to zero and \( x(i, i - 1) \) to one would violate (22). \( \square \)

The following lemma summarizes the implications of Lemmas 12, 13, 14 and 15.

**Lemma 16.** Let \( i \) be a node of \( BIC_n \) with \( \beta(i) = -1 \). Then the following holds

1. \( \alpha(i + 1, i) = \alpha(i - 1, i) = 1 \), and
2. \( \alpha(i, i - 1) = \alpha(i, i + 1) = 0 \).

**Proof.** (a1) is obtained from Lemma 13. Now if we suppose that (a2) is not true, then Lemma 14 and Lemma 15 imply that \( \alpha(u, v) = 1 \) for each \( (u, v) \in A(BIC_n) \) and \( \beta(u) = -1 \) for each \( u \in V(BIC_n) \). But this contradicts Lemma 12. \( \square \)

**Lemma 17.** If \( \alpha(i - 1, i) = 1 \) and \( \beta(i) = 0 \), then \( \alpha(i, i + 1) = 1 \).

**Proof.** There is a vector \( x \in F_{\alpha, \beta} \) with \( y(i - 1) + x(i - 1, i) + x(i - 1, i - 2) = 0 \).

- If \( y(i) = 1 \), we set \( x(i - 1, i) = 1 \) and violate the inequality; so \( y(i) = 0 \).
- If \( x(i, i + 1) = 0 \), then we can set \( y(i) = 1 \) and proceed as before; so \( x(i, i + 1) = 1 \).
- If \( \alpha(i, i + 1) = 0 \), we set \( x(i, i + 1) = 0 \) and proceed as before; so \( \alpha(i, i + 1) = 1 \). \( \square \)

**Lemma 18.** Suppose that we are not dealing with the bidirected cycle inequality. If \( \alpha(i, i + 1) = \alpha(i + 1, i) = 1 \) then \( \alpha(i + 2, i + 1) = \alpha(i - 1, i) = 0 \).

**Proof.** Assume \( i = 1 \). The proof is based on the statements below.

- It follows from Lemma 16 that \( \beta(1) = \beta(2) = 0 \).
- It follows from Lemma 17 that \( \alpha(2, 3) = \alpha(1, n) = 1 \).
- As illustrated in Figure 6, and since this is not a bidirected cycle inequality, we assume that there is an index \( k \geq 2 \) such that:
  - \( \beta(j) = \beta(j + 1) = 0 \), \( \alpha(j, j + 1) = \alpha(j + 1, j) = 1 \), for \( 1 \leq j \leq k \).
  - \( \alpha(n, 1) = \alpha(k + 2, k + 1) = 0 \).
  - \( \alpha(1, n) = \alpha(k + 1, k + 2) = 1 \).
- There is a vector \( x \in F_{\alpha, \beta} \) with \( y(k - 1) + x(\delta^+(k - 1)) = 0 \). We modify \( x \) as below to obtain a vector that violates the inequality.
  - If \( y(k) = 1 \) we just set \( x(k - 1, k) = 1 \).
  - If \( y(k) = 0 \) and \( x(k, k + 1) = 0 \), we set \( y(k) = 1 \) and proceed as above.
If \( y(k) = 0 \) and \( x(k, k+1) = 1 \), we set \( y(k+1) = x(k, k+1) = x(k+2, k+1) = 0 \), and \( y(k) = x(k-1, k) = x(k+1, k) = 1 \).

\[ \square \]

Lemma 19. If \( \alpha(i-1, i) = \alpha(i+1, i) = 1 \), then \( \beta(i) = -1 \).

Proof. Suppose \( \beta(i) = 0 \). It follows from Lemma 17 that \( \alpha(i-1, i) = \alpha(i+1, i) = 1 \). This contradicts Lemma 18.

Lemma 20. We have at least one of the values \( \alpha(i, i+1) \) or \( \alpha(i+1, i) \) equal to 1, for each \( i = 1, \ldots, n \).

Proof. Assume that \( \alpha(i, i+1) = \alpha(i+1, i) = 0 \). From Lemma 16, we have \( \beta(i) = \beta(i+1) = 0 \).

Let us show that \( \alpha(i+2, i+1) = 0 \). Suppose this is not the case. Let \((x, y) \in F_{\alpha, \beta}\) with \( x(i+2, i+3) + x(i+2, i+1) + y(i+2) = 0 \). If we set \( x(i+2, i+1) \) to 1 and if necessary we set \( y(i+1) \) to 1 and \( x(i+1, i) \) to 0, we obtain a feasible solution that violates (22).

Also \( \alpha(i+1, i+2) = 0 \). In fact, if this not the case take a solution \((x, y) \in F_{\alpha, \beta}\) with \( x(i+1, i+2) = 0 \) and \( y(i+2) = 1 \). Now if we set \( x(i+1, i+2) \) to 1 and possibly the variables \( x(i, i+1), x(i+1, i) \) and \( y(i+1) \) to zero we obtain a valid solution that violates (22). Again Lemma 16 implies that \( \beta(i+2) = 0 \).

If we continue this for the arcs \((i+2, i+3)\) and \((i+3, i+2)\) and so on, we obtain that \( \alpha(u, v) = 0 \) for each arc \((u, v) \in A(BIC_n)\), and \( \beta(v) = 0 \) for all \( v \in BIC_n \). This contradicts the fact that (22) defines a facet of \( UFLP'(G) \).

Let \( G_\alpha \) be the graph induced by the arcs \((i, j) \in A(BIC_n)\) with \( \alpha(i, j) = 1 \), we call this graph the support graph of (22). Recall that a bidirected path \( P \) of a graph \( G = (V, A) \) is a sequence of nodes \( P = \{1, 2, \ldots, k\} \) with \( (i, i+1) \) and \( (i+1, i) \) are both in \( A \), for \( i = 1, \ldots, k-1 \). The size of \( P \) is \( k-1 \). We say that \( P \) is maximal if we cannot extend it to a bidirected path from one of its ends.

Notice that by definition, the support graph of any g-odd lifted cycle inequality satisfies the following three properties:

- it contains a cycle as a subgraph,
- each maximal bidirected path is of size 1. Moreover, if \( P = \{i, i+1\} \) is such a path, then \((i-1, i)\) and \((i+2, i+1)\) do not appear in the support graph of the inequality, and
- if \( C \) is the g-odd cycle that defines the inequality, and \( i \) a node in \( \hat{C} \), then the support graph must contain exactly one of the arcs \((i-1, i)\) or \((i+1, i)\) when both nodes \( i-1 \) and \( i+1 \) are in \( \hat{C} \), it contains none of the arcs if both of these
nodes are in \( \hat{C} \), and finally if only one of these nodes, say \( i + 1 \) is in \( \hat{C} \), we must have the arc \((i + 1, i)\) in the support graph.

Now we enumerate some properties of \( G_\alpha \).

- Lemma 20 implies that \( G_\alpha \) contains at least one cycle as a subgraph. Choose any such a cycle and call it \( C \).
- Lemma 18 implies each maximal bidirected path is of size 1, and that for any such bidirected path \( P = \{i, i + 1\} \) the arcs \((i - 1, i)\) and \((i + 2, i + 1)\) are not in \( G_\alpha \). Lemma 20 implies that \((i, i - 1)\) and \((i + 1, i + 2)\) belong to \( G_\alpha \).
- Let \( i \in \hat{C} \). We have that \( G_\alpha \) must contain at most one of the arcs \((i - 1, i)\) and \((i + 1, i)\), since the size of maximal bidirected path is 1.
  
  - If both \( i - 1 \) and \( i + 1 \) are in \( \hat{C} \), then Lemma 19 implies that \( \beta(i - 1) = -1 = \beta(i + 1) \). Lemma 16 implies that \( \alpha(i - 1, i) = 0 = \alpha(i + 1, i) \). So in this case the arcs \((i - 1, i)\) and \((i + 1, i)\) are not in \( G_\alpha \).
  
  - Assume that \( i + 1 \) is in \( \hat{C} \) then Lemma 21 below implies that exactly one of the arcs \((i - 1, i)\) or \((i + 1, i)\) is in \( G_\alpha \).
  
  - If the node \( i - 1 \) is in \( \hat{C} \), Lemma 19 implies that \( \beta(i - 1) = -1 \) and Lemma 16 implies that \( \alpha(i - 1, i) = 0 \), so \((i - 1, i)\) is not an arc of \( G_\alpha \).

**Lemma 21.** If \( i \in \hat{C} \) and \( i + 1 \in \hat{C} \), then exactly one of the arcs \((i - 1, i)\) or \((i + 1, i)\) is in \( G_\alpha \).

**Proof.** Suppose that \( G_\alpha \) contains none of the arcs \((i - 1, i)\) or \((i + 1, i)\), that is \( \alpha(i - 1, i) = \alpha(i + 1, i) = 0 \). We have that \( \alpha(i, i - 1) = \alpha(i, i + 1) = 1 \). Lemma 16 implies that \( \beta(i) = 0 \). And since \( i + 1 \) is in \( \hat{C} \), we must have \( \alpha(i + 1, i + 2) = 1 \). Again Lemma 16 implies that \( \beta(i + 1) = 0 \).

We may assume that there is a solution \((x, y) \in F_{\alpha, \beta}\) with \( x(i + 1, i) = 1 \), otherwise (22) is the trivial inequality \( x(i + 1, i) \geq 0 \). Now if we set \( x(i, i + 1) \) and \( y(i + 1) \) to 1; \( x(i + 1, i) \) and \( y(i) \) to 0 and possibly \( x(i - 1, i) \), to 0, we obtain a feasible solution that violates (22). Therefore, we must have exactly one of the arcs \((i - 1, i)\) or \((i + 1, i)\) in \( G_\alpha \).

The above discussion shows that the support graph \( G_\alpha \) coincides with the support graph of the lifted g-odd cycle inequality defined from \( C \). Moreover, from Lemma 16, each node \( i \) with \( \beta(i) = -1 \) must be in \( \hat{C} \). And from Lemma 19, for each node \( i \in \hat{C} \) we have \( \beta(i) = -1 \). Since the g-odd cycle inequalities define facets for \( UFLP'(BIC_n) \), we know that there is a solution that satisfies it as equality, and hence \( \rho = \frac{|\hat{C}| + |\hat{C}| - 1}{2} \).

It is easy to check that inequalities (2)-(5) define facets for \( UFLP'(BIC_n) \), and from Theorems 5 and 2, we conclude that we have a minimal system of inequalities that defines \( UFLP'(BIC_n) \). This completes the proof of Theorem 6.

5. The dominating set polytope

Let \( G = (V, E) \) be an undirected connected graph. The graph \( G \) is a cactus if each edge of \( G \) is contained in at most one cycle of \( G \). For example every tree is a cactus. The main result of this section is a complete description of the dominating set polytope \( DSP(G) \) in \( \mathbb{R}^{|V| + |E|} \) when \( G \) is a cactus. This description may be seen as an extended formulation of \( DSP(G) \).
The characterization of $DSP(G)$ when $G$ is a cactus has been investigated in [8, 10, 9]. In these papers the authors characterize the dominating set polytope when $G$ is a chord less cycle. They also study the polytope of $DSP(G)$ when $G$ is obtained as a 1-sum of two graphs $G_1$ and $G_2$. They build $DSP(G)$ from $DSP(G'_1)$ and $DSP(G'_2)$, where $G'_1$ (resp. $G'_2$) is obtained as a 1-sum of $G_1$ (resp. $G_2$) and a 5-cycle (chord less cycle of size five). From the structure of a cactus graph, it follows that one may obtain the description of $DSP(G)$ in this class if we know the $DSP(G)$ for the class $\Gamma$ of graphs that are obtained by 1-sum of a cycle (edge) and a family of 5-cycles. But unfortunately they fail to characterize the polytope for the graphs in $\Gamma$. They show that this polytope may require some facets defining inequalities with coefficients $1, \ldots, p$ for any integer $p$. In our work we will give an extended formulation for $DSP(G)$ having coefficients in $\{-1, 0, 1\}$, and with this formulation we succeed to characterize the $DSP(G)$ in the class $\Gamma$. As a consequence we obtain an extended formulation for $DSP(G)$ when $G$ is a cactus graph.

For an undirected $G = (V, E)$, recall that $DSP(G)$ is the convex hull of the feasible 0-1 vectors satisfying the following linear system $D(G)$,

$$y(N[v]) \geq 1 \quad \forall v \in V, \quad (36)$$
$$y(v) \geq 0 \quad \forall v \in V, \quad (37)$$
$$y(v) \leq 1 \quad \forall v \in V, \quad (38)$$

Let $H = (V, A)$ any directed graph and define the following linear system $U(H)$,

$$\sum_{(u,v) \in A} x(u,v) + y(u) = 1 \quad \forall u \in V, \quad (39)$$
$$x(u,v) \leq y(v) \quad \forall (u,v) \in A, \quad (40)$$
$$x(u,v) \geq 0 \quad \forall (u,v) \in A, \quad (41)$$
$$y(v) \geq 0 \quad \forall v \in V, \quad (42)$$

this is the uncapacitated facility location linear relaxation. Here we have equalities instead of the inequalities (2). Define $UFLP(H)$ to be the convex hull of the feasible 0-1 vectors satisfying (39)-(42).

Now given an undirected graph $G = (V, E)$, define the directed graph $\vec{G} = (V, A)$ that have the same node-set as $G$, and its arc-set $A$ is defined from $E$ by replacing each edge $uv \in E$ by two arcs $(u, v)$ and $(v, u)$.

**Remark 22.** For any undirected graph $G = (V, E)$, the projection of $UFLP(\vec{G})$ over the $y$'s variables is exactly $DSP(G)$.

A key result is the following.

**Theorem 23.** Let $G = (V, E)$ an undirected cactus graph. The polytope $UFLP(\vec{G})$ is completely described by inequalities (10), (15) and (39)-(42).

The next subsection gives a composition theorem for $UFLP(G)$ for general directed graphs. This is useful for the proof of Theorem 23 that will be given in subsection 5.3.
5.1. The 1-sum composition for $UFLP(G)$. We consider a directed graph $G = (V, A)$ that decomposes into two graphs $G_1 = (V_1, A_1)$ and $G_2 = (V_2, A_2)$, with $V = V_1 \cup V_2$, $V_1 \cap V_2 = \{u\}$, $A = A_1 \cup A_2$, $A_1 \cap A_2 = \emptyset$.

In [3] the same decomposition has been studied for $UFLP'(G)$ (that is when we have inequalities instead of the equalities (39)). It has been shown that the 1-sum suffices to describe $UFLP'(G)$. That is we do not need auxiliary graphs defined from $G_1$ and $G_2$, only the descriptions of $UFLP'(G_1)$ and $UFLP'(G_2)$ suffice to get $UFLP'(G)$. This is not the case for $UFLP(G)$ as shown by the example of Figure 7, see [2].

\begin{figure}[h]
\centering
\includegraphics[width=0.5\textwidth]{figure7.png}
\caption{UFLP(G1) and UFLP(G2) are described by (39)-(42), but this not the case for UFLP(G).}
\end{figure}

We define $G'_1$ that is obtained from $G_1$ after replacing $u$ by $u'$ and adding the node $t'$ and the arc $(u', t')$. We also define $G'_2$, obtained from $G_2$ after replacing $u$ by $u''$ and adding the node $t''$ and the arc $(u'', t'')$, see Figure 8.

\begin{figure}[h]
\centering
\includegraphics[width=0.5\textwidth]{figure8.png}
\caption{Figure 8}
\end{figure}

We denote by $z_1$ the vector associated with the variables of the arcs and nodes of $G_1$, and we denote by $z_2$ the vector associated with the variables of the arcs and the nodes of $G_2$. Let $\alpha$ be the variable associated with $(u', t')$, and let $\beta$ be the variable associated with $(u'', t'')$. The following theorem shows that $UFLP(G)$ is obtained from $UFLP(G'_1)$ and $UFLP(G'_2)$ without the addition of a new family of inequalities.
Theorem 24. Suppose that the system
\begin{equation}
A\begin{bmatrix} \bar{z}_1 \\ \alpha \end{bmatrix} \leq b
\end{equation}
describes \text{UF LP}(G'_1). Similarly suppose that
\begin{equation}
C\begin{bmatrix} \bar{z}_2 \\ \beta \end{bmatrix} \leq d
\end{equation}
describes \text{UF LP}(G'_2). Then a system defining \text{UF LP}(G) is obtained by putting together (43) and (44) and
- replacing \(\alpha\) by \(z_2(\delta^+_{G_2}(u''))\),
- replacing \(\beta\) by \(z_1(\delta^+_{G_1}(u'))\),
- identifying \(z_1(u')\) and \(z_2(u'')\).

This system is denoted by
\begin{equation}
E\begin{bmatrix} \bar{z}_1 \\ \bar{z}_2 \end{bmatrix} \leq f.
\end{equation}

Proof. Let \((\bar{z}_1, \bar{z}_2)\) be an extreme point of the polytope defined by (45). Define \(\bar{\alpha} = \bar{z}_2(\delta^+_{G_2}(u''))\) and \(\bar{\beta} = \bar{z}_1(\delta^+_{G_1}(u'))\).

Case 1: \(\bar{z}_1(u') = 0\).

We have that \((\bar{z}_1, \bar{\alpha}) \in \text{UF LP}(G'_1)\) and \((\bar{z}_2, \bar{\beta}) \in \text{UF LP}(G'_2)\). If \((\bar{z}_1, \bar{\alpha})\) is an extreme point of \(\text{UF LP}(G'_1)\), we have to consider two sub-cases:

- \(\bar{z}_1(\delta^+_{G'_1}(u')) = 0\).
  If \((\bar{z}_2, 0)\) is not an extreme point of \(\text{UF LP}(G'_2)\), \(\bar{z}_2 = 1/2\lambda_1 + 1/2\lambda_2\), with \((\lambda_1, 0), (\lambda_2, 0)\) in \(\text{UF LP}(G'_2)\), \(\lambda_1 \neq \lambda_2\). Since \(\lambda_1(\delta^+_{G_2}(u'')) = 1, \lambda_2(\delta^+_{G_2}(u'')) = 1\), we have that \((\bar{z}_1, \bar{z}_2) = 1/2(\bar{z}_1, \lambda_1) + 1/2(\bar{z}_1, \lambda_2)\), with \((\bar{z}_1, \lambda_1)\) and \((\bar{z}_2, \lambda_2)\) satisfying (45), a contradiction. Thus \((\bar{z}_2, 0)\) is an extreme point and \((\bar{z}_1, \bar{z}_2)\) is an integral vector.

- \(\bar{z}_1(\delta^+_{G'_1}(u')) = 1\).
  This implies \(\bar{z}_2(\delta^+_{G'_2}(u'')) = 0\). If \((\bar{z}_2, 1)\) is not an extreme point, \(\bar{z}_2 = 1/2\lambda_1 + 1/2\lambda_2\), with \((\lambda_1, 1), (\lambda_2, 1)\) in \(\text{UF LP}(G'_2)\), \(\lambda_1 \neq \lambda_2\). Since \(\lambda_1(\delta^+_{G_2}(u'')) = 0 = \lambda_2(\delta^+_{G_2}(u''))\), we have that \((\bar{z}_1, \bar{z}_2) = 1/2(\bar{z}_1, \lambda_1) + 1/2(\bar{z}_1, \lambda_2)\), with \((\bar{z}_1, \lambda_1)\) and \((\bar{z}_2, \lambda_2)\) satisfying (45), a contradiction. Thus \((\bar{z}_2, 1)\) is an extreme point and \((\bar{z}_1, \bar{z}_2)\) is an integral vector.

Now we should study the situation in which \((\bar{z}_1, \bar{\alpha})\) and \((\bar{z}_2, \bar{\beta})\) are not extreme points.

We should have \((\bar{z}_1, \bar{\alpha}) = 1/2(\omega_1, \alpha_1) + 1/2(\omega_2, \alpha_2)\), with \((\omega_1, \alpha_1), (\omega_2, \alpha_2)\) in \(\text{UF LP}(G'_1)\), \((\omega_1, \alpha_1) \neq (\omega_2, \alpha_2)\). If \(\omega_1(\delta^+_{G'_1}(u')) = \omega_2(\delta^+_{G'_1}(u')) = \bar{z}_1(\delta^+_{G'_1}(u'))\), we have \((\bar{z}_1, \bar{z}_2) = 1/2(\omega_1, \bar{z}_2) + 1/2(\omega_2, \bar{z}_2)\), with \((\omega_1, \bar{z}_2)\) and \((\omega_2, \bar{z}_2)\) satisfying (45). A contradiction.

Now we assume that
\[
\omega_1(\delta^+_{G'_1}(u')) = \bar{z}_1(\delta^+_{G'_1}(u')) - \epsilon
\]
\[
\omega_2(\delta^+_{G'_1}(u')) = \bar{z}_1(\delta^+_{G'_1}(u')) + \epsilon,
\]
with \(\epsilon > 0\).
We also have \((\bar{z}_2, \bar{\beta}) = 1/2(\lambda_1, \beta_1) + 1/2(\lambda_2, \beta_2)\), with \((\lambda_1, \beta_1), (\lambda_2, \beta_2)\) in \(UFLP(G'_2)\), \((\lambda_1, \beta_1) \neq (\lambda_2, \beta_2)\). If \(\lambda_1 \delta^+_{G_2}(u'') = \lambda_2 \delta^+_{G_2}(u'') = \bar{z}_2 \delta^+_{G_2}(u'')\), we obtain a contradiction as above. Thus we suppose that
\[
\lambda_1 \left( \delta^+_{G_2}(u'') \right) = \bar{z}_2 \left( \delta^+_{G_2}(u'') \right) + \rho \\
\lambda_2 \left( \delta^+_{G_2}(u'') \right) = \bar{z}_2 \left( \delta^+_{G_2}(u'') \right) - \rho,
\]
with \(\rho > 0\).

We can assume that \(\epsilon = \rho\), otherwise we can change \(\lambda_1\) and \(\lambda_2\). Thus we have \((\bar{z}_1, \bar{z}_2) = 1/2(\omega_1, \lambda_1) + 1/2(\omega_2, \lambda_2)\), with \((\omega_1, \lambda_1)\) and \((\omega_2, \lambda_2)\) satisfying (45). A contradiction.

Case 2: \(0 < \bar{z}_1(u')\).

We have that \((\bar{z}_1, \bar{\alpha}) \in UFLP(G'_1)\) and \((\bar{z}_2, \bar{\beta}) \in UFLP(G'_2)\). Thus \((\bar{z}_1, \bar{\alpha})\) is a convex combination of extreme points \(\mu_i\) of \(UFLP(G'_1)\) that satisfy with equality every constraint that is satisfied with equality by \((\bar{z}_1, \bar{\alpha})\). Also \((\bar{z}_2, \bar{\beta})\) is a convex combination of extreme points \(\phi_j\) of \(UFLP(G'_2)\) that satisfy with equality every constraint satisfied with equality by \((\bar{z}_2, \bar{\beta})\).

We can assume that \(\mu_1(u') = 1 = \phi_1(u'')\). After putting together these two vectors we obtain a 0-1 vector that satisfies with equality every constraint that is satisfied with equality by the original vector \((\bar{z}_1, \bar{z}_2)\), a contradiction. \(\square\)

5.2. Algorithmic decomposition. The polyhedral decomposition shown in the last sub-section, has the following algorithmic counterpart. To decompose the optimization problem, we treat in \(G_1\) the following three cases.

- Let \(\lambda_0\) be the value of an optimal solution in \(G_1\) with the restriction \(y(u') + x(\delta^+_{G_1}(u')) = 0\).
- Let \(\lambda_1\) be the value of an optimal solution in \(G_1\) with the restriction \(y(u') = 1\).
- Let \(\lambda_2\) be the value of an optimal solution in \(G_1\) with the restriction \(x(\delta^+_{G_1}(u')) = 1\).

Then in \(G_2\) we give the weight \(\lambda_1 - \lambda_0\) to \(u''\), and the weight \(\lambda_2 - \lambda_0\) to \((u'', t'')\). Let \(W\) be the weight of an optimal solution in \(G_2\) with these weights, then the weight of an optimal solution of \(G\) is \(W + \lambda_0\).

5.3. The proof of Theorem 23. Let \(G = (V, E)\) be an undirected graph that is a cactus. The directed graph \(\overrightarrow{G} = (V, A)\) may be decomposed by means of 1-sum into bidirected cycles and graphs with two arcs \((u, v)\) and \((v, u)\). Given a bidirected cycle \(BIC_n\) and a subset \(T \subseteq V(BIC_n)\) add the arc \((t, t')\) and the node \(t'\) for each \(t \in T\). These new arcs will be called extra arcs. Call the resulting graph and extended bidirected cycle of \(BIC_n\). A trivial graph is the graph with two nodes two arcs \((u, v)\) and \((v, u)\) with possibly the arcs \((u, u')\) or/and \((v, v')\).

Notice that from Theorem 24 the polytope \(UFLP(\overrightarrow{G})\) when \(G\) is a cactus is fully described from the polytopes of extended bidirected cycles and those of trivial graphs.

The polytope \(UFLP(G)\) when \(G\) is a trivial graph is described by (39)-(42). It follows from Theorem 6 that the polytope \(UFLP(G)\) when \(G\) is an extended bidirected cycle is defined by (10), (15) and (39)-(42). It suffices to look at the variables associated with the extra arcs as the slack variables associated with inequalities (2).
6. Algorithmic consequences

In [9] the authors they give the first polynomial algorithm to solve the minimum weighted dominating set problem (WDSP) in a cycle. They showed that the separation of the inequalities defining the dominating set polytope in a cycle may be done in polynomial time. We have seen that this problem may be reduced to the uncapacitated facility problem in a bidirected cycle. From the proof of Lemma 9 it is not difficult to see that the uncapacitated facility location problem (UFLP) when the underlying graph is a bidirected cycle, may be reduced to the maximum maximum weight stable set problem (MWSSP) in a quasi-line graph and hence may be solved in polynomial time using any combinatorial algorithm in the more general class of claw-free graphs [29, 35, 22]. In the next subsection, we give a simple linear time combinatorial algorithm to solve the uncapacitated facility location problem when the underlying graph is a bidirected cycle. As a consequence, we obtain a linear time algorithm to solve both problems MWDSP and MWSSP, in cycles and in circular graphs, respectively.

In subsection 6.2 we give the first polynomial time algorithms to solve the UFLP in $\overrightarrow{G}$ when $G$ is a cactus graph. As a consequence we obtain the first polynomial time algorithms to solve the MWDSP in cacti.

6.1. Linear time algorithm for bidirected cycles. In this section we give a linear time combinatorial algorithm to solve the prize-collecting incapacitated facility location (pc-UFLP), when $G = BIC_n$. That is we want to solve (1)-(5) with the additional constraint that $(x, y)$ must be a 0-1 vector.

For any index $i$ we can decompose in the following three cases:

- Neither of $(i, i + 1)$ nor $(i + 1, i)$ is in the solution.
- $(i, i + 1)$ is in the solution.
- $(i + 1, i)$ is in the solution.

Each of the three preceding cases reduces to a pc-UFLP problem in a bidirected path. Now let us solve pc-UFLP in a bidirected path.

Suppose that we deal with a double path with nodes $1, \ldots, n$, and $n \geq 4$. The algorithm consists of the following two parts.

- First consider the bidirected path induced by $n - 2, n - 1, n$. We denote it by $P_0$. We keep the original weights, but we set $w(n - 2) = 0$. Let $\lambda_0$ be the weight of an optimal solution in $P_0$ without the arcs $(n - 2, n - 1)$ and $(n - 1, n - 2)$. Let $\lambda_1$ be the weight of an optimal solution in $P_0$ with $(n - 2, n - 1)$ in the solution. Let $\lambda_2$ be the weight of an optimal solution in $P_0$ with $(n - 1, n - 2)$ in the solution.
- Then denote by $P_1$ the bidirected path induced by $1, \ldots, n - 1$. We give the weight $\lambda_1 - \lambda_0$ to $(n - 2, n - 1)$ and the weight $\lambda_2 - \lambda_0$ to $(n - 1, n - 2)$. All other nodes and arcs keep their original weights. Let $W$ be the weight of an optimal solution in $P_1$, then the weight of an optimal solution in the original path is $W + \lambda_0$.

The same procedure is applied recursively to $P_1$. Since dealing with $P_0$ takes constant time, we have a linear time algorithm. Also, since the treating a bidirected cycle reduces to treating three bidirected paths, we have a linear time algorithm to pc-UFLP when the underlying graph is a bidirected cycle.
Notice that the same algorithm is applied to solve the uncapacitated facility location problem (UFLP). In this problem, all inequalities (2) are replaced by equalities. We need to solve the UFLP when dealing with the bidirected path $P_0$. As a consequence we have the following result.

**Theorem 25.** We can solve in linear time the MWDSP in a cycle, the UFLP in a bidirected cycle $BIC_n$ and the MWSSP in the circular graph $G_{2n} = I(BIC_n)$.

### 6.2. Polynomial time algorithm for cacti.

First we will give a cutting-plane polynomial time algorithm to solve pc-UFLP in the graph $\overrightarrow{G}$ when $G$ is a cactus. From Theorem (23) it suffices to develop a polynomial time algorithm to solve the separation problem associated with inequalities (10) and (15). Recall that $\overrightarrow{G}$ may be decomposed by means of 1-sum into bidirected cycles and bidirected paths of size one. From Theorems 24 and 23, the number of bidirected cycles is at most the number of nodes of $G$ and hence one can easily introduce the bidirected cycle inequalities (10) in any linear program. Hence we only need to solve the separation problem for the lifted g-odd inequalities (15) for each component of $\overrightarrow{G}$ that is a bidirected cycle.

#### 6.2.1. Separating lifted g-odd inequalities in a bidirected cycle.

Given a vector $(x, y)$ we want to verify if there is a lifted g-odd cycle inequality (15) violated by $(x, y)$ if there is any.

**Theorem 26.** The g-odd lifted cycle inequalities (15) may be separated in linear time for bidirected cycles.

**Proof.** A lifted g-odd cycle inequality (15) has the form

$$
\sum_{a \in A(C)} x(a) + \sum_{\hat{a} \in \hat{A}(C)} x(a) - \sum_{v \in \hat{C}} y(v) \leq \frac{|\hat{C}| + |\hat{C}| - 1}{2},
$$

with $|A(C)| + |\hat{C}|$ odd. It can also be written as

$$
\sum_{a \in A(C)} 2x(a) + \sum_{\hat{a} \in \hat{A}(C)} 2x(a) + \sum_{v \in \hat{C}} (1 - 2y(v)) \leq |A(C)| - 1,
$$

or

$$
(46) \sum_{a \in A(C)} (1 - 2x(a)) - \sum_{\hat{a} \in \hat{A}(C)} 2x(a) + \sum_{v \in \hat{C}} (2y(v) - 1) \geq 1.
$$

Thus we look for a cycle that violates (46). For that we create a directed graph $D' = (V', A')$ as follows. For every arc $(i, i+1)$ and $(i+1, i)$ we create a node in $D'$. The arcs in $A'$ are as below. See Figure 9.

- From $(i, i+1)$ to $(i+1, i+2)$ we create an arc with weight $1 - 2x(i+1, i + 2)$ and label “odd.”
- From $(i, i+1)$ to $(i+2, i+1)$ we create an arc with weight $2y(i+1) - 2x(i+2, i+1)$ and label “even.”
- From $(i+1, i)$ to $(i+1, i+2)$ we create an arc with weight $1 - 2x(i+1, i + 2)$ and label “odd.”
- From $(i+1, i)$ to $(i+2, i+1)$ we create an arc with weight $1 - 2x(i+2, i + 1)$ and label “odd.”
• From \((i, i - 1)\) to \((i + 1, i + 2)\) we create an arc with weight
\[
2 - 2x(i, i + 1) - 2x(i + 1, i) - 2x(i + 1, i + 2)
\]
and label “even.” This arc corresponds to the case when either \((i, i + 1)\) or \((i + 1, i)\) is in the lifting set \(\tilde{A}(C)\).

![Figure 9](image)

Then we look for a minimum weight directed cycle with an odd number of odd arcs in \(D'\). If the weight of such a cycle is less than one, we have found a violated inequality.

Now we give the details of how to find a minimum weight directed cycle with an odd number of odd arcs. We pick and index \(i\), and remove the arcs entering \((i, i + 1)\) and \((i + 1, i)\). We add an extra node \(s\) and connect it to \((i, i + 1)\) and \((i + 1, i)\) with even arcs of weight zero. For each node \(v\) in \(D'\) let \(f_o(v)\) (resp. \(f_e(v)\)) be the weight of a shortest path from \(s\) to \(v\) having an odd (resp. even) number of odd arcs. We set \(f_e(s) = 0\), \(f_o(s) = f_o(v) = f_e(v) = \infty\) for every other node \(v\) in \(D'\). We call the labels of \(s\) permanent and all others temporary. For each arc \((u, v)\) we denote by \(w(u, v)\) its weight. Then for a node \(v\) such that all its predecessors have permanent labels we update its labels as below.

\[
\begin{align*}
\tag{47} f_o(v) &= \min \left\{ \min_u \{f_o(u) + w(u, v) : (u, v) \text{ is even}\}, \right. \\
&\left. \min_u \{f_e(u) + w(u, v) : (u, v) \text{ is odd}\} \right\} \\
\tag{48} f_e(v) &= \min \left\{ \min_u \{f_o(u) + w(u, v) : (u, v) \text{ is odd}\}, \right. \\
&\left. \min_u \{f_e(u) + w(u, v) : (u, v) \text{ is even}\} \right\}
\end{align*}
\]

Then the labels of \(v\) are called permanent, and we continue.

Once all labels are permanent, we use the arcs entering \((i, i + 1)\) and \((i + 1, i)\) to find a shortest directed cycle with an odd number of odd arcs and including either \((i, i + 1)\) or \((i + 1, i)\). Next we have to consider the case when neither \((i, i + 1)\) nor \((i + 1, i)\) is in the shortest cycle. This is when the arc from \((i, i - 1)\) to \((i + 1, i + 2)\) is part of the shortest cycle. For that we repeat the same procedure with \(i' = i + 1\).

Since the indegree of each node in \(D'\) is at most three, the labels in (47) and (48) are computed in constant time for each node. Therefore this is a linear time algorithm. \(\square\)

From the above discussion we obtain the following.

**Theorem 27.** Let \(G\) a non oriented graph. We have a polynomial time cutting plane algorithm to solve the MWDSP when \(G\) is a cactus graph, the pc-UFLP and UFLP with respect to the directed graph \(\vec{G}\) and the MWSSP with respect to \(I(\vec{G})\).

Notice that the intersection graph \(I(\vec{G})\) is not a claw-free graph.
6.2.2. Linear time combinatorial algorithm. We conclude this section by noticing that the algorithmic decomposition given in subsection 5.2 together with the algorithm of subsection 6.1 give a linear time combinatorial algorithm for the problems mentioned in Theorem 27.

7. Concluding remarks

Let \( G = BIC_n \). In Theorem 6, we have proved that \( UFLP'(G) \) is described by inequalities (2)-(5), (10) and (15). We also obtain a characterization of \( UFLP(G) \) since it is a face of \( UFLP'(G) \). The polytope \( UFLP(G) \) may also be obtained from a transformation to the the stable set polytope as seen in subsection 4.1. In fact, eliminating the node variables from \( UFLP(BIC_n) \) gives the the convex hull of the stable sets in \( I(BIC_n) = G_{2n} \). The stable set polytope \( SSP(G_{2n}) \) has been described in [20] by a minimal system of linear inequalities. Using this description one may obtain the description of \( UFLP(BIC_n) \) and with a careful development we may also show that this description contain the inequalities (2)-(5), (10) and some of the inequalities (15). But we cannot obtain \( UFLP'(BIC_n) \), we need some additional valid inequalities.

The \( p \)-median problem is obtained from UFLP by adding the following quality

\[
(49) \quad \sum_{v \in V} y(v) = p,
\]

for some integer \( 1 \leq p \leq n \). That is the solutions of the \( p \)-median problem are those of the UFLP satisfying (49). Hence a linear relaxation of the \( p \)-median problem is given by the linear system \( U_p(G) \) defined by (39)-(42) and (49). In [4], a complete description of the graphs such that \( U_p(G) \) is integral has been given by means of forbidden subgraphs.

Now consider the polytope \( P_p(G) \) described by inequalities (2)-(5) and (49). We have the following result.

**Theorem 28.** For \( G = BIC_n \), the polytope \( P_p(G) \) is integral.

**Proof.** Suppose that \((\bar{x}, \bar{y})\) is an extreme fractional point of \( P_p(BIC_n) \). Let \( T \subseteq V \) be the set of nodes \( u \in V(BIC_n) \) such that inequality (2) with respect to \( u \) is not satisfied as equality by \((\bar{x}, \bar{y})\). For each node \( u \in T \) add the arc \((u, u')\), \( u' \) is a new node. This is an extended bidirected graph call it \( G' \). Notice that \( G' \) does not contain any of the forbidden subgraphs given in [4]. Therefore, \( U_k(G') \) is integral for any \( k \).

Extend \((\bar{x}, \bar{y})\) to a solution \((\bar{x}, \bar{y}) \in U_p'(G')\) with \( p' = p + |T| \); \( \bar{y}(u') = 1 \) for each node \( u' \) such that \( u \in T \); and for each \( u \in T \) set \( \bar{x}(u, u') = 1 - \bar{x}(\delta_{\leftarrow}(u)) \). It is easy to check that \((\bar{x}, \bar{y})\) is an extreme fractional point of \( U_p'(G') \), a contradiction. \( \square \)

This means that with the addition of the hyperplane (49), all the inequalities (15) with inequality (10) are not defining facets anymore. This is not the case for neither \( UFLP'(G) \) nor \( UFLP(G) \) when \( G \) is a cactus graph. Let us conclude this paper with the following important remark that states the relationship between the dominating set problem and the stable set problem.

**Remark 29.** Let \( G = (V, E) \) a undirected graph where each node \( v \in V \) is associated with a weight \( w(v) \). Define \( \overrightarrow{G} = (V, A) \) from \( G \) and let \( I(\overrightarrow{G}) = (A, E') \) be its intersection graph. For each node \((u, v) \) in \( I(\overrightarrow{G}) \) (this is an arc of \( \overrightarrow{G} \)) associate the weight
Then the minimum weight dominating set problem, MWDSF, reduces to the maximum weight stable set problem MWSP that finds a stable set $S$ in $I(G)$ that maximize $\sum_{(u,v) \in S} w(u)$. Also notice that if the value of the optimal solution of MWSP is $W$, then the value of the optimal solution of MWDSF is $\sum_{v \in V} w(v) - W$.

References


(M. Baïou) CNRS, and Université Clermont II, Campus des cézeaux BP 125, 63173 Aubière cedex, France.
E-mail address, M. Baïou: baiou@isima.fr

(F. Barahona) IBM T. J. Watson research Center, Yorktown Heights, NY 10589, USA.
E-mail address, F. Barahona: barahon@us.ibm.com