RC25488 (WAT1408-020) August 13, 2014
Mathematics

IBM Research Report

Algorithms for Minimum Weighted Dominating Sets in Cycles and Cacti

Mourad Baïou
CNRS
and
Université Clermont II
Campus des Cézeaux BP 125
63173 Aubière
Cedex, France

Francisco Barahona
IBM Research Division
Thomas J. Watson Research Center
P.O. Box 218
Yorktown Heights, NY 10598
USA

ALGORITHMS FOR MINIMUM WEIGHTED DOMINATING SETS IN CYCLES AND CACTI

MOURAD BAÏOU AND FRANCISCO BARAHONA

ABSTRACT. We study the minimum weighted dominating set problem in graphs. We give polynomial algorithms for cacti, and the first polynomial combinatorial algorithm for cycles. The previously known polynomial algorithm for cycles is based on the ellipsoid method. We also show that by adding extra variables, this can be formulated as a linear program of polynomial size.

Additionally we study the p-dominating set problem, where the cardinality of the set is required to be exactly p. We show that the natural linear programming formulation gives an integral polytope when the graph is a cycle. We also give a polynomial combinatorial algorithm for cacti.

1. INTRODUCTION

Let $G = (V, E)$ be an undirected graph. A set $D \subseteq V$ is called a *dominating set* if every node of $V \setminus D$ is adjacent to a node of D. The *minimum weighted dominating set problem* (MWDSP) consists of finding a dominating set D that minimizes $\sum_{v \in D} w(v)$, where $w(v)$ is a weight associated with each node $v \in V$. The natural relaxation of the MWDSP is defined by the linear program below

$$\min \sum_{v \in V} w(v)x(v) \tag{1}$$

$$\sum_{u \in N[v]} x(u) \geq 1 \quad \forall v \in V, \tag{2}$$

$$x(v) \geq 0 \quad \forall v \in V, \tag{3}$$

$$x(v) \leq 1 \quad \forall v \in V, \tag{4}$$

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 (2)-(4).

The MWDSP 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 [18, 17]. It has been shown that when the weights are all equal to 1, the MWDSP is solvable in many classes of graphs, a non-exhaustive list is cacti [22], trees [22], series-parallel graphs [19], permutation graphs [9, 10, 11, 15], cocomparability graphs [20], (see chapter 2 in [17] for more classes). For the weighted case of the MDWDSP, we only know three classes of graphs where this problem may be solved in polynomial time, namely for threshold graphs [21], for cycles [7] and for strongly chordal graphs [14]. Little is known about the polyhedral approach, and complete characterizations of the polytope are known only for the three classes of graphs

Date: August 13, 2014.

Key words and phrases. uncapacitated facility location problem, dominating set problem, lifted g-odd cycle inequalities.

mentioned above. For the case of strongly chordal graphs Farber [14] gives a primal-dual algorithm to solve the MWDSP this shows that $DSP(G)$ is defined by (2)-(4).

The polytope $DSP(G)$ has been first characterized for cycle graphs in [7], and later published in [8]. This result has also been established in [24] using a different approach. Namely if $C = (V, E)$ is a a cycle, $V = \{1, \ldots, n\}$, they proved that two types of inequalities have to be added to (2)-(4) to define $DSP(C)$. These are

$$\sum_{v \in V} x(v) \geq \left\lceil \frac{|V|}{3} \right\rceil, \tag{5}$$

when $|V|$ is not a multiple of 3. And

$$2 \sum_{v \in W} x(v) + \sum_{v \in V \setminus W} x(v) \geq \sum k_i + \left\lceil \frac{p}{2} \right\rceil, \tag{6}$$

where $W = \{v_1, \ldots, v_p\} \subset V$, $v_1 < v_2 < \ldots < v_p$, $p \geq 3$, p is odd, and

$$|C(v_i + 1, v_{i+1} - 1)| = 3k_i,$$

$k_i \geq 1$, for $i = 1, \ldots, p$ (mod p). Here $C(u, v)$ denotes the path $u, u+1, \ldots, u+t$ between u and v, where t is such that $u + t = v$, (the integers are taken modulo n). Notice that for each set W satisfying the definition above, one inequality (6) is required.

For a family $\mathcal{F}$ of inequalities, the *separation problem* consists of given a vector $\bar{x}$, finding an inequality in $\mathcal{F}$ violated by $\bar{x}$, or show that none exists. In [8] they gave a polynomial algorithm for the separation problem for inequalities (6). This combined with the Ellipsoid Method [16] shows that the MWDSP is polynomially solvable for cycles. A combinatorial algorithm for the MWDSP in cycles was not given in [8].

One may also use results related to the set covering polytope [13, 5, 6, 12, 23], 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.

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. Here we give a polynomial combinatorial algorithm for MWDSP in cycles. We extend this to cacti, giving polynomial algorithms for MWDSP in cacti. We use results on facility location to derive these algorithms. We also show that by adding extra variables, the MWDSP in cacti can be formulated as a linear program of polynomial size.

We also use results about the p-median problem to study the p-dominating set problem. Here the dominating set is required to have a fixed cardinality p. For cycles we show that the natural linear programming formulation gives a polytope with all integer extreme points. We also give a polynomial combinatorial algorithm for cacti.

We complete this introduction with some definitions. An *integral polytope* is a polytope with all integral extreme points. An undirected graph $G = (V, E)$ decomposes by means of a 1-sum, if G may be decomposed into two graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$, with $V = V_1 \cup V_2$, $V_1 \cap V_2 = \{u\}$ and $E_1 \cup E_2 = E$, $E_1 \cap E_1 = \emptyset$. For directed graphs a 1-sum is defined similarly. If a graph is a cactus, it can be obtained by means of 1-sums of cycles and paths.

To an undirected graph $G = (V, E)$ we associate a directed graph $\overset{\leftrightarrow}{G} = (V, A)$, where for each edge $uv \in E$ we include the arcs (u, v) and (v, u) in A.

For a directed graph $G = (V, A)$ and a set $W \subset V$, we denote by $\delta^+(W)$ the set of arcs $(u, v) \in A$, with $u \in W$ and $v \in V \setminus W$. Also we denote by $\delta^-(W)$ the set of arcs (u, v), with $v \in W$ and $u \in V \setminus W$. We write $\delta^+(v)$ and $\delta^-(v)$ instead of $\delta^+(\{v\})$ and $\delta^-(\{v\})$, respectively. If there is a risk of confusion we use δ^+_G and δ^-_G.

This paper is organized as follows. In Section 2 we review polyhedral results on facility location and we used them to derive a polynomial size extended formulation for $DSP(G)$ when G is a cactus. In Section 3 we discuss algorithmic aspects. In Section 4 we study the p-dominating set problem.

2. A polynomial size linear program for cacti

Here we review results on the facility location polytope that will be used in the sequel. If $G = (V, A)$ is 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.

$$\max \sum_{(u,v)\in A} w(u, v)y(u, v) + \sum_{v\in V} w(v)x(v) \tag{7}$$

$$\sum_{(u,v)\in A} y(u, v) + x(u) \leq 1 \quad \forall u \in V, \tag{8}$$

$$y(u, v) \leq x(v) \quad \forall (u, v) \in A, \tag{9}$$

$$y(u, v) \geq 0 \quad \forall (u, v) \in A, \tag{10}$$

$$x(v) \geq 0 \quad \forall v \in V, \tag{11}$$

For each node u, the variable $x(u)$ takes the value 1 if the node u is selected and 0 otherwise. For each arc (u, v) the variable $y(u, v)$ takes the value 1 if u is assigned to v and 0 otherwise. Inequalities (8) express the fact that either node u can be selected or it can be assigned to another node. Inequalities (9) 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 (8)-(10), and let $LP(G)$ be the convex hull of $P(G) \cap \{0, 1\}^{|V|+|A|}$. Define also $P^=(G)$ to be the polytope defined by (8)-(11) when inequalities (8) are replaced by equations. Let $LP^=(G)$ be the convex hull of $P^=(G) \cap \{0, 1\}^{|V|+|A|}$. Notice that $LP^=(G)$ is a face of $LP(G)$ and $LP(G) \subseteq P(G)$. In most cases new inequalities should be added to (8)-(11) to obtain $LP(G)$. Below we study this for cacti. Two new types of inequalities are needed for cacti, they are shown below.

Bidirected cycle inequalities. A bidirected cycle BIC_r is defined by the set of nodes $V = \{1, \dots, r\}$ and the set of arcs $A(BIC_r)$ consisting of $(i, i + 1)$ and $(i + 1, i)$ for each $i = 1, \dots, r$; the indices are taken modulo r. A bidirected path is defined in a similar way. It may be easily seen that the inequality

$$\sum_{a \in A(BIC_r)} y(a) \leq \left\lfloor \frac{2|r|}{3} \right\rfloor, \tag{12}$$

is valid for $LP(G)$. This inequality is called the *bidirected cycle inequality.* It has been introduced in [1].

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 $\dot{C}$ 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 $\tilde{C}$ 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 $|\hat{C}| = |\dot{C}|$. A cycle is called *g-odd* (generalized odd) if $p + |\dot{C}|$ (or $|\dot{C}| + |\tilde{C}|$) is odd, otherwise it is called *g-even*. A cycle C with $\dot{C} = \hat{C} = \emptyset$ 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* $\tilde{A}(C)$ as follows. For each node $i \in \dot{C}$ we have two cases:

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

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

$$\sum_{a \in A(C)} y(a) + \sum_{a \in \tilde{A}(C)} y(a) - \sum_{v \in \hat{C}} x(v) \leq \frac{|\tilde{C}| + |\hat{C}| - 1}{2}. \tag{13}$$

These inequalities have been introduced in [4]. Notice that given a g-odd cycle C, we might have several lifting sets $\tilde{A}(C)$, thus we might have several lifted g-odd cycle inequalities.

2.1. **Polytope description for a cactus.** An extended formulation with exponentially many inequalities has been given in [4] to describe $DSP(G)$ when G is a cactus. We built the directed graph $\overset{\leftrightarrow}{G}$ from the cactus undirected graph $G = (V, E)$ and we studied $LP^{=}(\overset{\leftrightarrow}{G})$. We first concentrated on $LP(G)$. Since G is obtained by means of 1-sums of cycles and paths, the first step was to characterize $LP(\overset{\leftrightarrow}{G'})$ when G' is a cycle. Then we used a polyhedral composition theorem in [2], which states that the families of inequalities needed for the description of $LP(D)$, when D is any graph obtained as a 1-sum of two graphs D_1 and D_1, is exactly the same families of inequalities needed to describe $LP(D_1)$ and $LP(D_2)$. Once $LP(\overset{\leftrightarrow}{G})$ is obtained it is straightforward to obtain $LP^=(G)$, it suffices to replace inequalities (8) by equations. This result is given in the theorem below.

Theorem 1. *If G is a cactus, then $LP(\overset{\leftrightarrow}{G})$ is described by the constraints (8)-(11), the bidirected cycle inequalities (12), and the lifted g-odd cycle inequalities (13).*

The following lemma is easy to prove.

Lemma 2. *For any undirected graph G, the projection of $LP^=(\overset{\leftrightarrow}{G})$ onto the x's variables is exactly $DSP(G)$.*

From this lemma and Theorem 1 we obtain the following.

Corollary 3. *If G is a cactus, then $LP^=(\overset{\leftrightarrow}{G})$ is described by the constraints (8)-(11) where* (8) *are replaced by equations, the bidirected cycle inequalities (12), and the lifted g-odd cycle inequalities (13). Since $DSP(G)$ is a projection of $LP^=(\overset{\leftrightarrow}{G})$, we have an extended formulation for $DSP(G)$.*

2.2. **A polynomial size linear program.** Now we use the formulation of $LP^=(\overset{\leftrightarrow}{G})$ when G is a cactus, but we add new variables so that inequalities (13) are replaced by a system of polynomial size. This is based on the separation algorithm developed in [4]. Given a vector $(\bar{x}, \bar{y})$ we want to find a lifted g-odd cycle inequality (13) violated by $(\bar{x}, \bar{y})$, if there is any.

A lifted g-odd cycle inequality (13) has the form

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

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

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

or

$$\sum_{a \in A(C)} (1 - 2y(a)) - \sum_{a \in \tilde{A}(C)} 2y(a) + \sum_{v \in \hat{C}} (2x(v) - 1) \geq 1. \tag{14}$$

Thus we look for a cycle that violates (14). For that we create a directed graph $D = (U, A)$ 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 1.

- From $(i, i+1)$ to $(i+1, i+2)$ we create an arc with label "odd" and weight
$$1 - 2\bar{y}(i+1, i+2). \tag{15}$$
- From $(i, i+1)$ to $(i+2, i+1)$ we create an arc with label "even" and weight
$$2\bar{x}(i+1) - 2\bar{y}(i+2, i+1). \tag{16}$$
- From $(i+1, i)$ to $(i+1, i+2)$ we create an arc with label "odd" and weight
$$1 - 2\bar{y}(i+1, i+2). \tag{17}$$

- From $(i+1,i)$ to $(i+2,i+1)$ we create an arc with label "odd" and weight

$$1 - 2\bar{y}(i+2,i+1). \tag{18}$$

- From $(i,i-1)$ to $(i+1,i+2)$ we create an arc with label "even" and weight

$$2 - 2\bar{y}(i,i+1) - 2\bar{y}(i+1,i) - 2\bar{y}(i+1,i+2). \tag{19}$$

  This arc corresponds to the case when either $(i,i+1)$ or $(i+1,i)$ is in the lifting set $\tilde{A}(C)$.

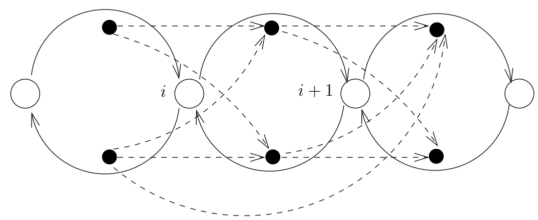

FIGURE 1

Then we look for a minimum weight directed cycle with an odd number of odd arcs in D. The cycle should be of one of the three types described below. If the weight of such a cycle is less than one, we have found a violated inequality. We pick any index i, and we treat the following three cases.

(a) The node associated with $(i,i+1)$ is in the cycle, the node associated with $(i+1,i)$ is not in the cycle, and the arc from $(i,i-1)$ to $(i+1,i+2)$ is not.
(b) The node associated with $(i+1,i)$ is in the cycle, the node associated with $(i,i+1)$ is not in the cycle, and the arc from $(i,i-1)$ to $(i+1,i+2)$ is not.
(c) The arc from $(i,i-1)$ to $(i+1,i+2)$ is in the cycle, and the nodes associated with $(i,i+1)$ and $(i+1,i)$ are not.

Now we show how to find such a cycle in case (a), the other two cases are similar. We remove the node associated with $(i+1,i)$ and the arc from $(i,i-1)$ to $(i+1,i+2)$. Denote by s the node associated with $(i,i+1)$. We split s into s' and s''. Arcs leaving s now leave s', and arcs entering s now enter s''. Denote by $D' = (U', A')$ the resulting graph.

To deal with the parity condition, we build a new graph $\bar{D} = (\bar{U}, \bar{A})$ as follows.

- $\bar{U}$ consists of two disjoint sets V_1 and V_2, with $s' \in V_1$, $s'' \in V_2$.
- For every other node $v \in U'$, $v \neq s', s''$, we make two copies, $v_1 \in V_1$, and $v_2 \in V_2$.
- For an even arc $(u,v) \in A'$, $u \neq s'$, $v \neq s''$, we add the arcs (u_1,v_1) and (u_2,v_2) to $\bar{A}$.
- For an odd arc $(u,v) \in A'$, $u \neq s'$, $v \neq s''$, we add the arcs (u_1,v_2) and (u_2,v_1) to $\bar{A}$.
- Arcs (s',v) and (v,s'') are treated in a similar way, depending upon they are odd or even.
- The weights of the new arcs are the same as the weights of their corresponding arcs in D'.

Then in $\bar{D}$ we look for a shortest path P from s' to s''. Once we identify s' and s'', and v_1 with v_2 for every other node v in D', we obtain a cycle. Since the graph D' contains no directed cycle, the cycle obtained from P is simple.

Now we formulate this as a linear program. Using network flows, we write this as

$$\min \sum_{(u,v)\in \bar{A}} f_{(u,v)}(x,y)\phi(u,v)$$
$$\sum_u \phi(u,v) - \sum_u \phi(v,u) = \begin{cases} -1 & \text{if } v = s' \\ 1 & \text{if } v = s'' \\ 0 & \text{otherwise} \end{cases}$$
$$\phi(u,v) \geq 0 \quad \text{for all } (u,v) \in \bar{A}.$$

Here $f_{(u,v)}(x,y)$ is the weight of the arc (u,v) as defined in (15)-(19). The variable $\phi(u,v)$ corresponds to the flow of the arc (u,v). Using the dual problem, we conclude that to satisfy the g-odd cycle inequalities, we should have a vector γ such that

$$\gamma(s'') - \gamma(s') \geq 1 \tag{20}$$
$$\gamma(v) - \gamma(u) \leq f_{(u,v)}(x,y), \quad \text{for every arc } (u,v) \in \bar{A}. \tag{21}$$

Notice that since the separation problem has been reduced to three shortest path problems, we need three linear systems like (20)-(21). Thus if a cactus graph is decomposed into k cycles, we need $3k$ linear systems like (20)-(21) to replace all lifted g-odd cycle inequalities. We conclude the following.

Theorem 4. *If G is a cactus, then the MWDSP can be formulated as a linear program of polynomial size. The same is true for the pc-UFLP in $\overleftrightarrow{G}$.*

3. Algorithmic aspects

In [8] the authors gave the first polynomial algorithm to solve the MWDSP in a cycle. They showed that the separation of the inequalities defining the dominating set polytope in a cycle, can be done in polynomial time, then their algorithm is based on the ellipsoid method [16]. In this section we show that using facility location techniques one can derive a simple combinatorial algorithm. 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 the MWDSP in cycles. In subsection 3.2 we give polynomial time algorithms to solve the UFLP in $\overleftrightarrow{G}$ when G is a cactus graph. As a consequence we obtain polynomial time algorithms to solve the MWDSP in cacti.

3.1. **Linear time algorithm for bidirected cycles.**

In this subsection we give an algorithm to solve the prize-collecting uncapacitated facility location (pc-UFLP), when $G = BIC_n$. That is we want to solve (7)-(11) 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 bidirected 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 λ_0 be the weight of an optimal solution in P_0 without the arcs $(n-2, n-1)$ and $(n-1, n-2)$. Let λ_1 be the weight of an optimal solution in P_0 with $(n-2, n-1)$ in the solution. Let λ_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 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 (8) are replaced by equations. The MWDSP in a cycle reduces to the UFLP in a bidirected cycle. As a consequence we have the following result

Theorem 5. *The MWDSP in a cycle (and the UFLP in a bidirected cycle) can be solved in linear time.*

3.2. **Polynomial time algorithms for cacti.** Let $D = (V, A)$ be a directed graph that is a 1-sum of $D_1 = (V_1, A_1)$ and $D_2 = (V_2, A_2)$, with $V_1 \cap V_2 = \{u\}$.

Consider the pc-UFLP in D. To decompose it we first treat in D_1 the following three cases.

- Let λ_0 be the value of an optimal solution in D_1 with the restriction
$$x(u) + \sum\{y(u,v) \mid (u,v) \in \delta^+_{D_1}(u)\} = 0.$$
- Let λ_1 be the value of an optimal solution in D_1 with the restriction $x(u) = 1$.
- Let λ_2 be the value of an optimal solution in D_1 with the restriction
$$\sum\{y(u,v) \mid (u,v) \in \delta^+_{D_1}(u)\} = 1.$$

Then we add a new node t to D_2 and the arc (u, t). Let D_2' be this new graph. Then in D_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 D_2' with these weights, then the weight of an optimal solution of D is $W + \lambda_0$.

We conclude this section by noticing that his algorithmic decomposition together with the algorithm of subsection 3.1 imply the following.

Theorem 6. *If G is a cactus, then the MWDSP can be solved in linear time. Also the pc-UFLP in $\overleftrightarrow{G}$ can be solved in linear time.*

4. The p-dominating set problem

In this section we extend results from the p-median problem to the dominating set problem. For a graph $G = (V, E)$ and a positive integer p, we consider now dominating sets $D \subseteq V$ with $|D| = p$. In this section we show that for a cycle it is enough to add an equation to system (2)-(4) to have an integral polytope. Surprisingly inequalities (5) and (6) are not needed. This is stated in Theorem 7 below.

Theorem 7. *If $G = (V, E)$ is a cycle, then system below defines an integral polytope.*

$$x(N[v]) \geq 1 \quad \forall v \in V, \tag{22}$$

$$x(v) \geq 0 \quad \forall v \in V, \tag{23}$$

$$x(v) \leq 1 \quad \forall v \in V, \tag{24}$$

$$\sum_{v \in V} x(v) = p. \tag{25}$$

Proof. Consider $\overleftrightarrow{G} = (V, A)$, and the system below,

$$\sum_{(u,v) \in A} y(u, v) + x(u) = 1 \quad \forall u \in V, \tag{26}$$

$$y(u, v) \leq x(v) \quad \forall (u, v) \in A, \tag{27}$$

$$y(u, v) \geq 0 \quad \forall (u, v) \in A, \tag{28}$$

$$x(v) \geq 0 \quad \forall v \in V, \tag{29}$$

$$\sum_{v \in V} x(v) = p. \tag{30}$$

This is a linear relaxation of the p-median problem in $\overleftrightarrow{G}$, where the number of open facilities is required to be exactly p. It was shown in [3] that this system defines an integral polytope if and only if G is a path or a cycle.

To complete the proof we have to show that when we project the variables y, we obtain the system (22)-(25). In order to apply Fourier-Motzkin elimination, we re-write the system (26)-(30) as below.

$$\sum_{(u,v) \in A} y(u, v) + x(u) \leq 1 \quad \forall u \in V, \tag{31}$$

$$-\sum_{(u,v) \in A} y(u, v) - x(u) \leq -1 \quad \forall u \in V, \tag{32}$$

$$y(u, v) - x(v) \leq 0 \quad \forall (u, v) \in A, \tag{33}$$

$$-y(u, v) \leq 0 \quad \forall (u, v) \in A, \tag{34}$$

$$-x(v) \leq 0 \quad \forall v \in V, \tag{35}$$

$$\sum_{v \in V} x(v) \leq p, \tag{36}$$

$$-\sum_{v \in V} x(v) \leq -p. \tag{37}$$

Then to eliminate a variable $y(u, v)$, we sum an inequality where this variable has the coefficient 1 with an inequality where the variable has the coefficient -1. This is done for all pairs of inequalities with these characteristics. The details are as follows.

- The combination of (32) and (33) gives inequalities (22).
- The combination of (31) and (34) gives inequalities (24).
- The combination of (33) and (34) gives inequalities (23).
- Inequalities (35), (36) and (37) remain unchanged.

The proof is complete. □

We obtain the result below.

Corollary 8. *The p-dominating set problem in cycles is polynomially solvable.*

Remark 9. *In general, for cacti the polytope defined by* (22)-(25) *is not integral. To see this, consider the graph $G = (V, E)$, where*

$$V = \{1, 2, 3, 4, 5, 6, 7\} \text{ and}$$
$$E = \{\{1, 2\}, \{1, 3\}, \{2, 4\}, \{3, 5\}, \{4, 6\}, \{5, 6\}, \{6, 7\}\}.$$

Consider the vector $x(1) = 1$, $x(2) = x(3) = 0$, $x(4) = x(5) = x(6) = x(7) = 1/2$. This is an extreme point, to see this, notice that it is the unique solution of the following system of equations.

$$\begin{aligned} x(1) &= 1 \\ x(2) &= 0 \\ x(3) &= 0 \\ x(2) + x(4) + x(6) &= 1 \\ x(3) + x(5) + x(6) &= 1 \\ x(6) + x(7) &= 1 \\ x(1) + x(2) + x(3) + x(4) + x(5) + x(6) + x(7) &= 3 \end{aligned}$$

4.1. **Extension to Cacti.** Here we give an algorithm for the p-median problem in cacti, that is used to solve the p-dominating set problem.

4.1.1. *Decomposition algorithm.* Suppose that D is a directed graph that is a 1-sum of $D_1 = (V_1, A_1)$ and $D_2 = (V_2, A_2)$, and $V_1 \cap V_2 = \{u\}$. We need the definitions below.

- Let $\lambda_0(k)$ be the optimal value of the k-median problem in $D_2 \setminus u$.
- Let $\lambda_1(k)$ be the optimal value of the k-median problem in D_2 with the constraint that u is included in the solution.
- Let $\lambda_2(k)$ be the optimal value of the k-median problem in D_2 with the constraint that u is not included in the solution. In this case an arc (u, t) should be in the solution, with $t \in V_2 \setminus u$.

Let D_1' be the graph obtained from D_1 by adding the node u', and the arc (u, u'). Then in D_1' we give the weight $\lambda_1(k+1) - \lambda_0(k)$ to u, and the weight $\lambda_2(k) - \lambda_0(k)$ to the arc (u, u'). If we solve the $(p - k + 1)$-median problem in D_1', we obtain a solution of value $\alpha(k)$. It contains $p - k$ nodes in V_1, and can be combined with a solution in D_2 with k nodes in $V_2 \setminus u$. This gives a solution in D whose value is $\beta(k) = \alpha(k) + \lambda_0(k)$. From this we obtain the optimal value of the p-median problem as

$$\min_{0 \le k \le p} \beta(k).$$

4.1.2. *Combinatorial algorithm for a cycle.* Here we assume that $D = (V, A)$ is a bidirected cycle, and D' has been obtained by adding a node u' for each $u \in V$, and also the arc (u, u'). These new arcs are called *artificial*. This is the general case that has to be used in the decomposition algorithm above.

For any index i we 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.

In each case we have a bidirected path with the artificial arcs defined above. This can be decomposed by means of 1-sums. So the decomposition algorithm above is applied, where one piece consists of a bidirected path with two original nodes and two artificial nodes. This piece is treated in constant time. Since the algorithm uses p values of the parameter k, we have an algorithm that requires quadratic time. We state this below.

Theorem 10. *The p-dominating set problem in a cycle $C = (V, E)$ can be solved in $O(|V|^2)$ time.*

When we apply this decomposition algorithm to a cactus, each piece is a cycle. Treating one piece once takes quadratic time, and since at most p values of the parameter k are needed, the algorithm requires cubic time. We have the following.

Theorem 11. *The p-dominating set problem in a cactus $G = (V, A)$ can be solved in $O(|V|^3)$ time.*

5. Concluding remarks

We have used results for facility location to study the dominating set problem. Instead of having only the node variables, having the arc variables allowed us to not only derive polyhedral characterizations, but to also obtain linear time algorithms. By adding extra variables we have formulated the MWDSP in cacti as a polynomial size linear program.

We also used results from the p-median problem to study the p-dominating set problem, and prove Theorem 7. We have not been able to find a direct proof of this apparently simple result.

References

[1] P. Avella, A. Sassano, and I. Vasilév, *Computational study of large-scale p-median problems*, Mathematical Programming, 109 (2007), pp. 89–114.
[2] M. Baïou and F. Barahona, *On the integrality of some facility location polytopes*, SIAM Journal on Discrete Mathematics, 23 (2009), pp. 665–679.
[3] M. Baïou and F. Barahona, *On the linear relaxation of the p-median problem*, Discrete Optimization, 8 (2011), pp. 344 – 375.
[4] M. Baïou and F. Barahona, *Simple extended formulation for the dominating set polytope via facility location*, tech. rep., 2014.
[5] E. Balas and S. M. Ng, *On the set covering polytope: I. all the facets with coefficients in 0, 1, 2*, Mathematical Programming, 43 (1989), pp. 57–69.
[6] ———, *On the set covering polytope: Ii. lifting the facets with coefficients in 0, 1, 2*, Mathematical Programming, 45 (1989), pp. 1–20.
[7] M. Bouchakour, *I: Composition de graphes et le polytope des absorbants, II: Un algorithme de coupes pour le problème du flot à coût fixes*, PhD thesis, Université de Rennes 1, Rennes, France, December 1996.

[8] M. Bouchakour, T. M. Contenza, C. W. Lee, and A. R. Mahjoub, *On the dominating set polytope*, Eur. J. Comb., 29 (2008), pp. 652–661.
[9] A. Brandstadt and D. Kratsch, *On the restriction of some np-complete graph problems to permutation graphs*, in Fundamentals of Computation Theory, L. Budach, ed., vol. 199 of Lecture Notes in Computer Science, Springer Berlin / Heidelberg, 1985, pp. 53–62.
[10] A. Brandstadt and D. Kratsch, *On domination problems for permutation and other graphs*, Theoretical Computer Science, 54 (1987), pp. 181 – 198.
[11] D. G. Corneil and L. K. Stewart, *Dominating sets in perfect graphs*, in Topics on Domination, S. Hedetniemi, ed., vol. 48 of Annals of Discrete Mathematics, Elsevier, 1991, pp. 145 – 164.
[12] G. Cornuéjols and B. Novick, *Ideal 0, 1 matrices*, Journal of Combinatorial Theory, Series B, 60 (1994), pp. 145 – 157.
[13] G. Cornuéjols and A. Sassano, *On the 0, 1 facets of the set covering polytope*, Mathematical Programming, 43 (1989), pp. 45–55.
[14] M. Farber, *Domination, independent domination, and duality in strongly chordal graphs*, Discrete Applied Mathematics, 7 (1984), pp. 115 – 130.
[15] M. Farber and J. M. Keil, *Domination in permutation graphs*, Journal of Algorithms, 6 (1985), pp. 309 – 321.
[16] M. Grotschel, L. Lovász, and A. Schrijver, *Geometric algorithms and combinatorial optimizations*, (1993).
[17] T. Haynes, S. Hedetniemi, and P. Slater, *Domination in Graphs: Volume 2: Advanced Topics*, Monographs and Textbooks in Pure and Applied Mathematics, Taylor & Francis, 1998.
[18] ———, *Fundamentals of Domination in Graphs*, Monographs and Textbooks in Pure and Applied Mathematics, Taylor & Francis, 1998.
[19] T. Kikuno, N. Yoshida, and Y. Kakuda, *A linear algorithm for the domination number of a series-parallel graph*, Discrete Applied Mathematics, 5 (1983), pp. 299 – 311.
[20] D. Kratsch and L. Stewart, *Domination on cocomparability graphs*, SIAM J. Discret. Math., 6 (1993), pp. 400–417.
[21] A. R. Mahjoub, *Polytope des absorbants dans une classe de graphe a seuil*, North-Holland Mathematics Studies, 75 (1983), pp. 443–452.
[22] Ø. Ore, *Theory of Graphs*, no. pt. 1 in Colloquium Publications - American Mathematical Society, American Mathematical Society, 1962.
[23] M. Sánchez-García, M. Sobrón, and B. Vitoriano, *On the set covering polytope:facets with coefficients in* $\{0, 1, 2, 3\}$, Annals of Operations Research, 81 (1998), pp. 343–356.
[24] A. Saxena, *Some results on the dominating set polytope of a cycle*. GSIA Working paper 2004-E28, 2004.

(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`