RC25232 (W1111-042) November 7, 2011
Mathematics

IBM Research Report

A Combinatorial Algorithm for Facility Location with No G-odd Cycles

Mourad Baïou
CNRS
Laboratoire LIMOS
Campus des Cézeaux BP 125,
63173 Aubiere Cedex, France

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

A COMBINATORIAL ALGORITHM FOR FACILITY LOCATION WITH NO G-ODD CYCLES

MOURAD BAÏOU AND FRANCISCO BARAHONA

Abstract. In [1] we had proved that the linear relaxation of a facility location model defines an integral polytope if and only if the graph has no g-odd cycles. Here we give a combinatorial algorithm for facility location in this class of graphs.

1. Introduction

Let $G = (V, A)$ be a directed graph, not necessarily connected, where each arc and each node has a weight associated with it. The linear system below defines a linear programming relaxation of a "prize collecting" version of a *location problem* studied in [1].

$$\max \sum w(u,v)x(u,v) + \sum w(v)y(v) \tag{1}$$

$$\sum_{(u,v)\in A} x(u,v) + y(u) \le 1 \quad \forall u \in V, \tag{2}$$

$$x(u,v) - y(v) \le 0 \quad \forall (u,v) \in A, \tag{3}$$

$$y(v) \ge 0 \quad \forall v \in V, \tag{4}$$

$$x(u,v) \ge 0 \quad \forall (u,v) \in A. \tag{5}$$

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 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.

It was proved in [1] that this system of inequalities defines an integral polytope if and only if the graph does not contain a g-odd cycle. The definition of g-odd cycle will be given in the next section. In [1] it was also shown how recognize this class of graphs. Later Chen et al. [2] showed that the constraint matrix is totally unimodular if an only there is no g-odd cycle. This implies that the system (2)-(5) is totally dual integral. This means that for any objective function with integer coefficients, the linear program (1)-(5) has an optimal dual solution that is integral. Based on Yannakakis's work [3] on restricted totally unimodular matrices, Chen et al. [2] show that the linear program (1)-(5) can be reduced to a b-matching problem in a bipartite graph and to a maximum weighted independent set in a bipartite graph. In this paper we give a direct combinatorial algorithm to solve the linear program (1)-(5) when graph does not contain a g-odd cycle. This algorithm produces an optimal primal solution that is integral, and if the objective function coefficients are integral then the algorithm also produces an optimal dual solution that is integral.

Date: November 4, 2011.

Key words and phrases. facility location, g-odd cycles.

This paper is organized as follows. In Section 2 we give some basic definitions. In Section 3 we study the complemenary slackness conditions. Section 4 contains a labeling procedure to change the dual variables. Then in Section 5 we analyse this procedure. Section 6 shows how to change the primal variables. In Section 7 we study the Uncapacitated facility location problem.

2. Preliminary definitions

In this section we give some basic definitions. 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. A simple cycle C is an ordered sequence

$$v_0, a_0, v_1, a_1, \dots, a_{p-1}, v_p,$$

where

- v_i, $0 \le i \le p-1$, are distinct nodes,
- a_i, $0 \le i \le p-1$, 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 $0 \le i \le p-1$, 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, $1 \le i \le p$.
- 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, $1 \le i \le p$.
- 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, $1 \le i \le p$.

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

2.1. First labeling procedure.

2.1. First labeling procedure. Given a path $P = v_0, a_0, \dots, a_{p-1}, v_p$. Assume that the label of a_0, $l(a_0)$ has the value 1 or -1. We define the *first labeling procedure* used in [1] as follows.

For $i = 1$ to $p - 1$ do

L_1: If v_i is the head of a_{i-1} and it is the tail of a_i then $l(v_i) = l(a_{i-1})$, $l(a_i) = -l(v_i)$.
L_2: If v_i is the head of a_{i-1} and it is the head of a_i then $l(v_i) = l(a_{i-1})$, $l(a_i) = l(v_i)$.
L_3: If v_i is the tail of a_{i-1} and it is the head of a_i then $l(v_i) = -l(a_{i-1})$, $l(a_i) = l(v_i)$.
L_4: If v_i is the tail of a_{i-1} and it is the tail of a_i then $l(v_i) = 0$, $l(a_i) = -l(a_{i-1})$.

Notice that the labels of v_0 and v_p were not defined.

This procedure will be used in two different cases as below.

Case 1. G contains a directed cycle $C = v_0, a_0, \dots, a_{p-1}, v_p$. Assume that the head of a_0 is v_1, set $l(v_0) = -1$, $l(a_0) = 1$ and extend the labels as above.

Case 2. G contains a cycle $C = v_0, a_0, \ldots, a_{p-1}, v_p$ and $\dot{C} \neq \emptyset$. Assume $v_0 \in \dot{C}$. Set $l(v_0) = 0$, $l(a_0) = 1$ and extend the labels.

The lemma below was proved in [1].

Lemma 1. *If C is g-even, then after labeling as in Cases 1 and 2 we have $l(a_{p-1}) = -l(a_0)$.*

On the other hand if C is g-odd we have the following.

Lemma 2. *If C is g-odd, then after labeling as in Cases 1 and 2 we have $l(a_{p-1}) = l(a_0)$.*

Proof. Let $a_{p-1} = (u, v)$. Replace a_{p-1} by $(u, v'), v', (v', v)$, and let C' be the new cycle. Since C' is g-even, it follows from Lemma 1 that in Case 1 we have $l(v', v) = -l(a_0)$ and $l(u, v') = l(a_0)$.

Similarly in Case 2 we have $l(u, v') = -l(a_0)$ and $l(v', v) = l(a_0)$. □

From these two lemmas we derive the following basic property of the labeling procedure.

Lemma 3. *If we apply the labeling procedure on a cycle C, the labels are consistent with the rules $L_1, \ldots, L_4$ if and only if C is g-even.*

2.2. **Second labeling procedure.** In this paper we need a second labeling procedure derived from the one above as follows.

- Construct a graph G' by reversing the orientation of all arcs.
- Apply the above labeling procedure to G'.
- Copy the labels to G and reverse the signs of the node labels.

Thus the rules above become as follows

L'_1: If v_i is the head of a_{i-1} and it is the tail of a_i then $l(v_i) = l(a_{i-1})$, $l(a_i) = -l(v_i)$.
L'_2: If v_i is the head of a_{i-1} and it is the head of a_i then $l(v_i) = 0$, $l(a_i) = -l(a_{i-1})$.
L'_3: If v_i is the tail of a_{i-1} and it is the head of a_i then $l(v_i) = -l(a_{i-1})$, $l(a_i) = l(v_i)$.
L'_4: If v_i is the tail of a_{i-1} and it is the tail of a_i then $l(v_i) = -l(a_{i-1})$, $l(a_i) = l(a_{i-1})$.

Lemma 3 is translated as below.

Lemma 4. *If we apply the second labeling procedure on a cycle C, the labels are consistent with the rules $L'_1, \ldots, L'_4$ if and only if C is g-even.*

3. Complementary Slackness

The dual of (1)-(5) is

$$\min \sum \alpha(u)$$

$$\alpha(u) + \beta(u, v) \geq w(u, v) \quad \forall (u, v) \in A, \tag{6}$$

$$\alpha(u) - \sum_{w} \beta(w, u) \geq w(u) \quad \forall u \in V, \tag{7}$$

$$\alpha, \beta \geq 0. \tag{8}$$

The complementary slackness conditions are

$$x(u,v) > 0 \Longrightarrow \alpha(u) + \beta(u,v) = w(u,v) \tag{9}$$

$$y(u) > 0 \Longrightarrow \alpha(u) - \sum_{w} \beta(w,u) = w(u) \tag{10}$$

$$\alpha(u) > 0 \Longrightarrow \sum_{(u,v)\in A} x(u,v) + y(u) = 1 \tag{11}$$

$$\beta(u,v) > 0 \Longrightarrow x(u,v) = y(v). \tag{12}$$

The algorithm will start with a primal solution that satisfies (2)-(5), and a dual solution that satisfies (6)-(8), and the only complementary slackness conditions that are violated are (11). At each iteration the algorithm produces solutions with the same properties, and such that the number of violated conditions (11) decreases by at least one at every iteration.

4. Dual Changes

We start with $\alpha(u) = M$, a big number, for each node u. We set to zero all other variables. Then we look for a node r such that (11) is violated and we give it the label $''-''$. This means that we will try to decrease the value of $\alpha(r)$. The node r will be called the *root*. Nodes and arcs will receive the labels $''+''$ or $''-''$. This means that we will try to add or substract some value ϵ from the dual variables associated with them. The labels are propagated through different paths as described below.

After labeling a node (resp. an arc), if no further labels are needed from this node (resp. arc), we say that we have a success for this node (resp. arc). On the other hand if after labeling this node (resp. arc), we conclude that a dual change is not possible, we have a failure for this node (resp. arc).

In Figure 1 we illustrate the different cases that will be treated in the labeling procedure below. If an arc (u,v) is depicted with a thin line, it means $x(u,v) = 0$. If it is depicted with a thick line, it means $x(u,v) = 1$. A node u with $y(u) = 0$ is represented by a circle, and it is represented by a square if $y(u) = 1$. The numbers correspond to the different subsections. It should be easy to see that the labeling procedure below follows the rules $L'_1, \ldots, L'_4$ of Subsection 2.2.

4.1. **Treating a node u with the label $''-''$.**

If a node u has the label $''-''$, we should treat tight constraints (6) associated with u as in 4.2, and a tight constraint (7) associated with u as in 4.3. If none of these constraints is tight, we can decrease the value of $\alpha(u)$ and we have a success.

4.2. **Treating a node u with the label $''-''$ to satisfy (6).**

We give the label $''+''$ to each arc (u,v) with $x(u,v) = 0$ for which (6) is tight. Then each of these arcs should be treated to satisfy (7) as in 4.4.

Remark 5. *If an arc (u,v) had a label, it should be the label $''+''$ and we should have $x(u,v) = 1$.*

Proof. If (u,v) had the label $''-''$, it was given in 4.3 or in 4.4, and we would have a g-odd cycle.

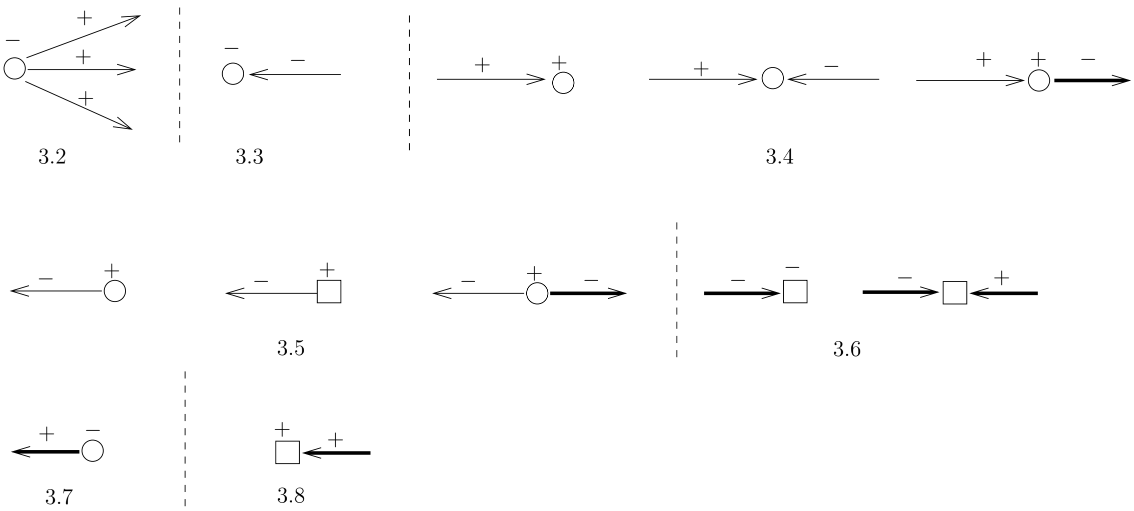

FIGURE 1. Overview of the labeling procedure

If (u, v) had the label $''+''$, it had been given in 4.6 or in 4.8. In both cases we had $x(u, v) = 1$. □

4.3. **Treating a node u with the label $''-''$ to satisfy (7).** If (7) for u is tight, we give the label $''-''$ to an arc (w, u) with $\beta(w, u) > 0$. It follows from Remark 6 that (w, u) did not have a label. Then (w, u) should be treated to satisfy (6) as in 4.5. If there are several canditate arcs, we should search for a successful path using each of them. If there is no candidate arc, we have found a failure.

Remark 6. *There is no arc (v, u) that has a label.*

Proof. If u is the root it is obvious. Otherwise u had received its label in 4.7, and there is an arc leaving u with the label $''+''$. An arc (v, u) could have received a label in 4.2 or in 4.4. In both cases there is an arc directed into u with the label $''+''$, this implies the existance of a g-odd cycle. □

4.4. **Treating an arc (u, v) with the label $''+''$ and $x(u, v) = 0$ to satisfy (7).** If constraint (7) for v is not tight, condition (10) implies $y(v) = 0 = x(u, v)$, then we can increase the value of $\beta(u, v)$ and we have a successful path. Otherwise (7) is tight and we have the five cases below.

- If $y(v) = 1$, then $\beta(u, v) = 0$ and the value of $\beta(u, v)$ cannot be increased, because a complementary slackness condition (12) would be violated. Here we have found a failure.
- If v already has a label, it should be the label $''+''$, and no other arc (w, v), $w \neq u$, can have a label. This follows from Remarks 7 and 8 below. In this case we have a success.
- If $y(v) + \sum_w x(v, w) = 0$, and $\alpha(v) > 0$, we give the label $''+''$ to v, and we have a successful path. Here (11) for v was violated and remains violated.
- If $y(v) + \sum_w x(v, w) = 0$, and $\alpha(v) = 0$. If there is an arc (w, v) with $\beta(w, v) > 0$, $w \neq u$, we give the label $''-''$ to (w, v), then (w, v) is treated to satisfy (6) as in 4.5. It follows from Remark 8 that such an arc did not have a label. If there are several arcs with these characteristics, a search for a dual change should be done

from each of them. If there is no arc (w,v) with $\beta(w,v) > 0$, $w \neq u$, we have found a failure.

- If there is an arc (v,w) with $x(v,w) = 1$, we have two possibilities.
  - We can give the label $''+''$ to v, then in order to satisfy (9) we give the label $''-''$ to (v,w). If $\beta(v,w) = 0$ we have a failure, otherwise if $\beta(v,w) > 0$, the arc (v,w) should be treated to satisfy (10) as in 4.6.
  - If there is an arc (t,v) with $\beta(t,v) > 0$, we can give it the label $''-''$, then (t,v) is treated to satisfy (6) as in 4.5.

It follows from Remarks 8 and 9, that the arcs being labeled in the two cases above, did not have a label before being treated here. A search for a dual change should be done for each of these two possibilities. If none of these posibilities exists we have a failure.

Remark 7. *If $y(v) = 0$ and v had a label before being treated here it should be $''+''$, and it should have been received in 4.5.*

Proof. Assume that v had a label. If it had the label $''-''$, it would have been received in 4.7 and we would have a g-odd cycle. If it had the label $''+''$, and it had been received in 4.4 we would have a g-odd cycle. So it had been received in 4.5. ☐

Remark 8. *If $y(v) = 0$, an arc (w,v), $w \neq u$, could not have a label before being treated here.*

Proof. If (w,v) had received a label, it could not have been in 4.3 because v would have the label $''-''$ and this contradicts the previous remark. So it would have been either in 4.2 or in 4.4. In either case we would have an arc different from (u,v), directed into v and with the label $''+''$. This implies the existance of a g-odd cycle. ☐

Remark 9. *If $y(v) = 0$, and v had no label, then an arc (v,t) with $x(v,t) = 1$, could not have a label before being treated here.*

Proof. If an arc (v,t), with $x(v,t) = 1$, had received a label, it would have been either in 4.6 or in 4.8. In both cases we would have a g-odd cycle. ☐

4.5. **Treating an arc (u,v) with the label $''-''$ and $x(u,v) = 0$ to satisfy (6).** Let (u,v) be an arc with the label $''-''$, if (6) is not tight, we have a successful path. Otherwise (6) is tight and we have five cases.

- If u already has a label, it should be the label $''+''$, otherwise there is a g-odd cycle. In this case we have found a successful path.
- If u has no label, $\alpha(u) = 0$ and $y(u) + \sum_w x(u,w) = 0$, we have found a failure.
- If u has no label, $\alpha(u) > 0$ and $y(u) + \sum_w x(u,w) = 0$, we just give the label $''+''$ to u and we have found a successful path. Here condition (11) for u is violated, and it will remain violated after the dual change.
- If u has no label and $y(u) = 1$, we give the label $''+''$ to u. Then u is treated to satisfy (10) as in 4.8.
- If u has no label and there is an arc (u,w) with $x(u,w) = 1$, we have two cases.
  - If $\beta(u,w) > 0$, we give the label $''+''$ to u and in order to satisfy (9) we give the label $''-''$ to (u,w). It follows from Remark 10 that (u,w) did not have a label before being treated here. Then (u,w) is treated to satisfy (10) as in 4.6.
  - If $\beta(u,w) = 0$, we have found a failure.

Remark 10. *If $x(u,w) = 1$ for an arc (u,w), then (u,w) did not have a label before being treated here.*

Proof. If such an arc would have received a label, it would be either in 4.6 or in 4.8. In both cases we would have a g-odd cycle. □

4.6. **Treating an arc (u,v) with label $''-''$ and $x(u,v) = 1$ to satisfy (10).** Here we have three cases.

- If $\alpha(v) > 0$, we can give it the label $''-''$ to v, then v should be treated as in 4.2. It follows from Remark 11 that v did not have a label before being treated here.
- Or if there is an arc (w,v) with $x(w,v) = 1$, we can give it the label $''+''$, then (w,v) should be treated to satisfy (9) as in 4.7. Remark 12 shows that (w,v) did not have a label before being treated here.
- If $\alpha(v) = 0$ and there is no other arc (w,v) with $x(w,v) = 1$, then a failure has been found.

A search for a dual change should be done for each of the first two cases, if they exist.

Remark 11. *The node v did not have a label before being treated here.*

Proof. If v had received a label in 4.5 it would be $''+''$, this implies the existance of a g-odd cycle. If v had received a label in 4.6 we would have a g-odd cycle. □

Remark 12. *An arc (w,v) with $x(w,v) = 1$ and $w \neq u$, could not have a label before being treated here.*

Proof. The first time that such an arc receives a label, it would be in 4.4 or in 4.5 and it would be $''-''$. This implies the existance of a g-odd cycle. □

4.7. **Treating an arc (u,v) with $x(u,v) = 1$ and with the label $''+''$ to satisfy (9).** Here we have two cases.

- If $\alpha(u) > 0$ we give the label $''-''$ to u, and u should be treated as in 4.1.
- If $\alpha(u) = 0$, we have found a failure.

Remark 13. *The node u could not have a label before being treated here.*

Proof. If it had a label it should be $''-''$, otherwise there is a g-odd cycle. Here is the only case where a node u with $y(u) = 0$, different from r receives the label $''-''$, so we should have another arc (u,w) with $x(u,w) = 1$, this violates (2). □

Remark 14. *There is no other arc incident to u that has a label*

Proof. If an arc (w,u) has a label, it should be the label $''-''$, otherwise we would have a g-odd cycle. From the previous remark we have that u did not have a label before, so (w,u) had received a label in 4.4. In this case we would have another arc directed into u with the label $''+''$. This implies the existance of a g-odd cycle.

If an arc (u,w), with $w \neq v$, has a label, it should be the label $''+''$, otherwise there is a g-odd cycle. But it is not possible for such an arc to receive the label $''+''$. □

4.8. **Treating a node u with the label $''+''$ and $y(u) = 1$ to satisfy (10).** If the node u has the label $''+''$ and $y(u) = 1$, in order to satisfy (10), we should give the label $''+''$ to an arc (w, u) with $x(w, u) = 1$, then (w, u) should be treated to satisfy (9) as in 4.7. If there are several arcs with the above properties, we should search for a successful path from each of them. If such an arc does not exist, we have found a failure.

Remark 15. *The node u received its label in 4.5 from an arc (u, v) having the label $''-''$. There is no arc (w, u) with $x(w, u) = 1$, that had a label before treating u here.*

Proof. Suppose that an arc (w, u) with $x(w, u) = 1$ had a label. If it is the label $''-''$ we would have a g-odd cycle. If it is the label $''+''$, it was given in 4.6. Then we would have another arc arc (t, u) with $x(t, u) = 1$ and with the label $''-''$. This implies the existence of a g-odd cycle. □

5. Analysis of the labels

From all remarks in the preceding section it follows that each time that a node or an arc receives a label, it did not have a label before. For a particular root, the labels should be extendeded in a depth first search fashion. When there are several choices, one should pick one and if it does not lead to a success, one should backtrack and try a different choice. Since each node and arc is labeled at most once, the labeling procedure takes $O(n + m)$ operations. Each path ends either with a success or a failure.

If the labeling procedure determines that a dual change is possible, then a value ϵ is added or subtracted from the dual variables based on the labels. This is the largest value so that all dual constraints remain satisfied. We have the following integrality property.

Lemma 16. *If the objective function coefficients are all integral, and and the initial dual solution is integral, then ϵ is an integer, and every dual vector produced by this procedure is integral.*

Proof. Consider an inequality (6) that is not tight. If it leads to a fractional value of ϵ, then u and (u, v) should have the label $''-''$. In this case we would have a g-odd cycle.

Consider now an inequality (7) that is not tight and that leads to a fractional value of ϵ. This can happen in the following two cases.

- If u has the label $''-''$ and and arc (w, u) has the label $''+''$. Here we would have a g-odd cycle.
- If two arcs (w, u) and (t, u) have the label $''+''$. Again this implies the existance of a g-odd cycle.

□

After each dual change, a new dual constraint becomes tight. Once a constraint becomes tight, it remains tight after all dual changes made using the same root. So $2(m + n)$ is a bound for the number of consecutive dual changes before a new condition (11) is satisfied, or a primal change is needed.

6. Primal Changes

If a dual change is not possible, we have found at least one path that leads to a failure. We use this path to change the primal solution. We denote by (x', y') the new primal vector. We start with $x' = x$, $y' = y$. We repeat the labeling procedure starting from r, but with the following changes. These changes are so that primal feasibility is maintained, and all complementary slackness conditions that are satisfied remain satisfied.

- In 4.1 we have to decide if the failure comes from 4.2 or 4.3. If both lead to a failure we choose arbitrarily one. Then the primal change is propagated as below.
- If in 4.2 we had one arc (u, v) that leads to a failure, we set $x'(u, v) = 1$ and the primal change is propagated as in 4.4. If several arcs lead to a failure, we choose one arbitrarily.
- If in 4.3 we had a failure, then we set $y'(u) = 1$ and for every arc (w, u) with $\beta(w, u) > 0$ we set $x'(w, u) = 1$. Then the primal change is propagated for each of these arcs as in 4.5.
- Now we discuss the five cases arising in 4.4:
  - If $y(v) = 1$, nothing else has to be done here.
  - The case when v already has a label cannot arise here, because it would imply that it is a successful path.
  - For the same reasons the case when $y(v)+\sum x(v, w) = 0$ and $\alpha(v) > 0$, does not arise.
  - If $y(v) + \sum x(v, w) = 0$ and $\alpha(v) = 0$, we set $y'(v) = 1$. We also set $x'(w, v) = 1$ for each arc (w, v) with $\beta(w, v) > 0$, $w \neq u$. Then for each of these arcs, we keep propagating the primal change as in 4.5. If arcs with these characteristics do no exist, nothing else is needed here.
  - If there is an arc (v, w) with $x(v, w) = 1$, then we set $x'(v, w) = 0$. If $\beta(v, w) > 0$ then the arc (v, w) should be treated as in 4.6. Also we set $y'(v) = 1$ and if there is some arc (w, v) with $\beta(w, v) > 0$, $w \neq u$, we set $x'(w, v) = 1$ for each of these arcs, and we keep propagating the primal change as in 4.5.
- In 4.5 there are five cases:
  - The case when u already has a label cannot arise, because it implies a succesful path.
  - If u has no label, $\alpha(u) = 0$ and $y(u) + \sum_w x(u, w) = 0$, nothing else is done here.
  - The third case (u has no label, $\alpha(u) > 0$ and $y(u)+\sum_w x(u, w) = 0$) cannot arise. We would have a successful path.
  - If u has no label and $y(u) = 1$, then we set $y'(u) = 0$ and u is treated as in 4.8.
  - If u has no label and there is an arc (u, w) with $x(u, w) = 1$, we set $x'(u, w) = 0$; and if $\beta(u, w) > 0$ then (u, w) is treated as in 4.6, otherwise $\beta(u, w) = 0$ and nothing else is done here.
- In 4.6 we set $y'(v) = 0$. Then there are three cases to treat:
  - If $\alpha(v) > 0$ then v should be treated as in 4.2.
  - For any arc (w, v), $w \neq u$, with $x(w, v) = 1$, we set $x'(w, v) = 0$ and (w, v) should be treated as in 4.7.
  - In the third case ($\alpha(v) = 0$ and there is no other arc (w, v), $w \neq u$, with $x(w, v) = 1$), nothing else is done here.
- In 4.7 we have two cases:
  - In the first case ($\alpha(u) > 0$), we continue to propagate the primal change as in 4.1.
  - In the second case ($\alpha(u) = 0$), nothing else has to be done here.

- In 4.8 we have set $y'(u) = 0$ for the node u. Then for each arc (w, u) with $x(w, u) = 1$, we set $x'(w, u) = 0$ and (w, u) should be treated as in 4.7. If there is no arc as described, nothing else is done here.

When we start the primal change from the root r, either a variable $x'(r, v)$ is set to one, or the variable $y'(r)$ is set to one, so a new condition (11) is satisfied.

Remark 17. *During the primal change each variable is changed at most once.*

Proof. It follows from all remarks in Section 4 that each node and each arc receives a label at most once. □

After a primal change, condition (11) for r is satisfied. When condition (11) for r is satisfied we say that we have completed a *major iteration.*

To summarize, the algorithm consists of the following steps:

Step 1. Pick a root r for which (11) is violated. I none exists stop.
Step 2. Apply the labeling procedure starting from r. If a dual change is possible go to Step 3, otherwise go to Step 4.
Step 3. Change the dual solution. If (11) for r is still violated go to Step 2, otherwise go to Step 1.
Step 4. Change the primal solution, go to Step 1.

For a fixed root r we can have at most $2(m + n)$ dual changes, so the complexity of one major iteration is $O(m^2)$ (Can we improve this bound?). Since we have at most n major iterations, the entire algorithm takes $O(nm^2)$ operations.

We have now an algorithmic proof of the following theorem proved in [1].

Theorem 18. *If G is a graph with no g-odd cycle, then* (2)-(5) *defines an integral polytope.*

Also we have a proof of the theorem below proved in [2].

Theorem 19. *If G is a graph with no g-odd cycle, then the system* (2)-(5) *is totally dual integral.*

7. Uncapacitated Facility Location

A commonly studied case is the *Uncapacitated Facility Location Problem.* Here we assume that V is partitioned into V_1 and V_2, $A \subseteq V_1 \times V_2$, and we deal with the linear programming relaxation

$$\max \sum w(u,v)x(u,v) + \sum w(v)y(v) \tag{13}$$

$$\sum_{(u,v)\in A} x(u,v) = 1 \quad \forall u \in V_1, \tag{14}$$

$$x(u,v) - y(v) \le 0 \quad \forall (u,v) \in A, \tag{15}$$

$$0 \le y(v) \le 1 \quad \forall v \in V_2, \tag{16}$$

$$x(u,v) \ge 0 \quad \forall (u,v) \in A. \tag{17}$$

We denote by $\Pi(G)$ the polytope defined by (14)-(17). Notice that $\Pi(G)$ is a face of $P(G)$. Let $\bar{V}_1$ be the set of nodes $u \in V_1$ with $|\delta^+(u)| = 1$. Let $\bar{V}_2$ be the set of nodes in V_2 that are adjacent to a node in $\bar{V}_1$. It is clear that the variables associated with nodes in $\bar{V}_2$ should be fixed, i.e., $y(v) = 1$ for all $v \in \bar{V}_2$. Let us denote by $\bar{G}$ the subgraph induced by $V \setminus \bar{V}_2$. In [1] we proved that $\Pi(G)$ is an integral polytope if and only if $\bar{G}$ has no g-odd cycle. Later in [2] it was proved that the system (14)-(17) is totally dual integral if and only if $\bar{G}$ has no g-odd cycle.

In order to use our combinatorial algorithm to solve (13)-(17) when $\bar{G}$ has no g-odd cycle, we apply the following transformations.

- Split the nodes in $\bar{V}_2$ as follows. For $v \in \bar{V}_2$ and for each arc $(u, v) \in \delta^-(v)$ we add a node v_u and replace (u, v) by (u, v_u). We set $w(v_u) = 0$, and $w(u, v_u) = w(u, v)$. Once all arcs in $\delta^-(v)$ are treated, the node v is removed. Let $\tilde{G}$ be this new graph. Clearly $\tilde{G}$ has no g-odd cycle and we can solve (1)-(5).
- Set $M = \sum_{(u,v)} w(u, v) + \sum_v w(v)$. Give the weight $-M$ to each node $u \in V_1$, this implies $y(u) = 0$.
- Add M to each weight $w(u, v)$ for each arc (u, v). This is to impose equations (14).

Let λ be the optimal value obtained after solving (1)-(5). Then the optimal value of (13)-(17) is $\lambda - M|V_1| + \sum_{v\in\bar{V}_2} w(v)$.

References

[1] M. Baïou and F. Barahona, *On the integrality of some facility location polytopes*, SIAM J. Discrete Math., 23 (2009), pp. 665–679.

[2] X. Chen, Z. Chen, and W. Zang, *Total dual integrality in some facility location problems*, Technical Report, The University of Hong Kong, 2009.

[3] M. Yannakakis, *On a class of totally unimodular matrices*, Math. Oper. Res., 10 (1985), pp. 280–304.

(M. Baïou) CNRS, Laboratoire LIMOS, 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