

## Graph Orientation Problems for Multiple $st$ -Reachability

TAKEHIRO ITO,<sup>†1</sup> YUICHIRO MIYAMOTO,<sup>†2</sup>  
HIROTA KA ONO,<sup>†3</sup> HISAO TAMAKI<sup>†4</sup>  
and RYUHEI UEHARA<sup>†5</sup>

Consider the situation in which we wish to give one-way restrictions to aisles in a limited area, such as an industrial factory. We model this situation as graph orientation problems, where multiple  $st$ -pairs are given together with an edge-weighted graph and we seek an orientation that minimizes some objective function reflecting directed  $st$ -distances under the orientation. In this paper, we introduce two objectives, and study the corresponding two minimization problems: the first is MIN-SUM type, and the second is MIN-MAX type. We first show that both MIN-SUM ORIENTATION and MIN-MAX ORIENTATION problems are strongly NP-hard for planar graphs, and that MIN-MAX ORIENTATION remains NP-hard even for cacti. We then show that both problems can be solved in polynomial-time for cycles. Finally, we consider the problems restricted to cacti. Then, MIN-SUM ORIENTATION can be solved in polynomial-time, and MIN-MAX ORIENTATION admits a polynomial-time 2-approximation algorithm and a pseudo-polynomial-time algorithm.

### 1. Introduction

Consider the situation in which we wish to give one-way restrictions to aisles in a limited area, such as an industrial factory. Since traffic jams rarely occur in industrial factories, the distances of detours are more important for the efficiency. We model this situation as graph orientation problems, in which we wish to find an orientation so that the distances of detours are not so long for given multiple  $st$ -pairs.

Let  $G = (V, E)$  be an undirected graph together with an assignment of a non-

**Table 1** Summary of our results

	MIN-SUM ORIENTATION	MIN-MAX ORIENTATION
planar graphs	strongly NP-hard even for unweighted graphs	strongly NP-hard even for unweighted graphs
cacti	$O( V q^2)$	<ul style="list-style-type: none"> <li>• NP-hard even for <math>q = 2</math></li> <li>• polynomial-time 2-approximation</li> <li>• pseudo-polynomial-time algorithm</li> </ul>
cycles	$O( V q^2)$	$O( V q^2)$

negative integer weight  $\omega(e)$  to each edge  $e$  in  $G$ . Assume that we are given  $q$  pairs of vertices  $s_i$  and  $t_i$ ,  $1 \leq i \leq q$ , in  $G$ . Then, an *orientation* of  $G$  is an assignment of exactly one direction to each edge in  $G$  so that there exists a directed path from  $s_i$  to  $t_i$  for every pair  $(s_i, t_i)$ ,  $1 \leq i \leq q$ . We denote by  $\omega(P)$  the total weight of a directed path  $P$ , that is,  $\omega(P) = \sum_{e \in P} \omega(e)$ . For an oriented graph  $\vec{G}$  and a pair  $(s_i, t_i)$ , let  $f(\vec{G}, s_i, t_i)$  be the total weight of a shortest directed path from  $s_i$  to  $t_i$  in  $\vec{G}$ , that is,

$$f(\vec{G}, s_i, t_i) = \min\{\omega(P) \mid P \text{ is a directed path from } s_i \text{ to } t_i \text{ in } \vec{G}\}.$$

We introduce two objective functions for orientations of a graph  $G$ , and study the corresponding two minimization problems. The first objective function is  $g(\vec{G}) = \sum_{1 \leq i \leq q} f(\vec{G}, s_i, t_i)$ , that is, the sum of the total weights of shortest directed paths for all pairs  $(s_i, t_i)$ ,  $1 \leq i \leq q$ . Then, the corresponding problem, called the MIN-SUM ORIENTATION problem, is to find an orientation of  $G$  such that  $g(\vec{G})$  is minimum. The second objective function is  $h(\vec{G}) = \max\{f(\vec{G}, s_i, t_i) \mid 1 \leq i \leq q\}$ , that is, the maximum total weight of a shortest directed path among all pairs  $(s_i, t_i)$ ,  $1 \leq i \leq q$ . Then, the corresponding problem, called the MIN-MAX ORIENTATION problem, is to find an orientation of  $G$  such that  $h(\vec{G})$  is minimum. Clearly, both MIN-SUM ORIENTATION and MIN-MAX ORIENTATION can be solved in polynomial-time if we are given a single pair  $(s_1, t_1)$ , that is,  $q = 1$ ; in this case, we simply seek a shortest path between  $s_1$  and  $t_1$ .

Robbins<sup>7)</sup> showed that every 2-edge-connected graph can be oriented so that the resulting digraph is strongly connected. On the other hand, Hakimi, Schmeichel and Young<sup>3)</sup> proposed a quadratic algorithm for the problem of orienting a 1-edge-connected graph (namely, with cut edges) so as to maximize the number of ordered vertex-pairs  $(x, y)$  having a directed path from  $x$  to  $y$ . It is easy to see that the problem of<sup>3)</sup> can be reduced to our MIN-SUM ORIENTATION.

<sup>†1</sup> Tohoku University

<sup>†2</sup> Sophia University

<sup>†3</sup> Kyushu University

<sup>†4</sup> Meiji University

<sup>†5</sup> Japan Advanced Institute of Science and Technology

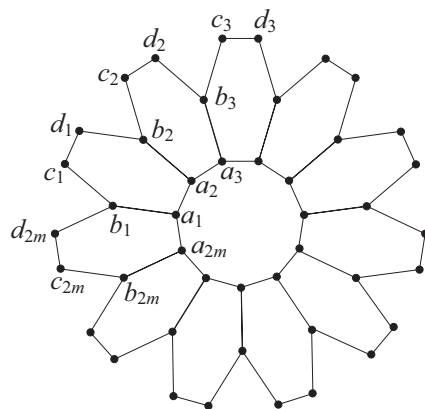


Fig. 1 Petal  $H(M, m)$ .

In this paper, we show the computational hardness, and give algorithms for cycles and cacti. (Our results are summarized in Table 1.) In Section 2, we show the computational hardness of our problems. More specifically, we show that both problems are strongly NP-hard for planar graphs even if  $\omega(e) = 1$  for all edges  $e$ , and that MIN-MAX ORIENTATION remains NP-hard even for cacti with  $q = 2$ . We then show in Section 3 that both problems can be solved in polynomial-time for cycles. Finally, in Section 4, we consider the problems restricted to cacti. Then, MIN-SUM ORIENTATION can be solved in polynomial-time, and MIN-MAX ORIENTATION admits a polynomial-time 2-approximation algorithm and a pseudo-polynomial-time algorithm.

## 2. Computational Hardness

In this section, we show the computational hardness of our problems. We first show in Section 2.1 that our two problems are both strongly NP-hard for planar graphs. In Section 2.2, we then show that MIN-MAX ORIENTATION remains NP-hard even for cacti.

### 2.1 Planar graphs

Before giving the main theorem, we consider the following special instance of MIN-MAX ORIENTATION. Let  $m$  and  $M$  be fixed large integers. Consider the

planar graph  $G = (V, E)$  such that  $V = \{a_i, b_i, c_i, d_i \mid 1 \leq i \leq 2m\}$  and  $E = \{\{a_{i+1}, a_i\}, \{a_i, b_i\}, \{b_i, c_i\}, \{c_i, d_i\}, \{d_i, b_{i+1}\} \mid 1 \leq i \leq 2m\}$ , where  $a_{2m+1} = a_1$  and  $b_{2m+1} = b_1$ . (See Figure 1.) Then,  $G$  is composed of  $2m$  hexagonal elementary cycles. For the sake of convenience, we fix the embedding of  $G$  such that the outer face consists of  $b_i, c_i, d_i$ ,  $1 \leq i \leq 2m$ , and such that  $a_2 a_1 b_1 c_1 d_1 b_2$  are placed in a clockwise direction, as illustrated in Figure 1. The weights of edges are defined as follows:  $\omega(\{a_{i+1}, a_i\}) = \omega(\{b_i, c_i\}) = \omega(\{d_i, b_{i+1}\}) = M$  and  $\omega(\{a_i, b_i\}) = \omega(\{c_i, d_i\}) = 1$  for each  $i$ ,  $1 \leq i \leq 2m$ . Finally, we define  $6 \times 2m$  pairs, as follows:

$$\{(a_i, d_i), (d_i, a_i), (b_i, b_{i+1}), (b_{i+1}, b_i), (c_i, a_{i+1}), (a_{i+1}, c_i) \mid 1 \leq i \leq 2m\}.$$

Then, it is easy to see that this instance of MIN-MAX ORIENTATION has only two optimal orientations: the one is to orient each elementary cycle  $a_{i+1} b_i c_i d_i b_{i+1}$  in a clockwise direction (or, in an anticlockwise direction) if  $i$  is odd (respectively, even); another is the reverse of the other. We call this instance a *petal*, and denote it by  $H(M, m)$ . We say that the petal  $H(M, m)$  is oriented in a clockwise direction if  $H(M, m)$  is optimally oriented such that the cycle  $a_2 a_1 b_1 c_1 d_1 b_2$  is oriented in a clockwise direction.

**Theorem 1.** MIN-SUM ORIENTATION and MIN-MAX ORIENTATION is strongly NP-hard for planar graphs even if all edge-weights are identical.

*Proof.* We show that PLANAR 3-SAT (known to be strongly NP-complete<sup>2),6)</sup> can be reduced in polynomial time to the decision versions of our problems.

Let  $U$  be the set of Boolean variables, and let  $n = |U|$ . Let  $C$  be the set of clauses over  $U$  such that each  $c \in C$  satisfies  $|c| = 3$  and such that the bipartite graph  $G = (V, E)$ , where  $V = U \cup C$  and  $E$  contains exactly those pairs  $\{u, c\}$  such that either  $u$  or  $\bar{u}$  belongs to the clause  $c$ , is planar; and let  $m = |C|$ . The PLANAR 3-SAT problem is to decide whether there is a satisfying truth assignment for  $C$ .

We now construct a graph  $G'$  as an instance of MIN-MAX ORIENTATION, as follows. We fix an embedding of  $G$  on the plain arbitrarily. For each clause  $c_i \in C$ ,  $1 \leq i \leq m$ , we introduce three vertices  $r_i, s_i, t_i$ . For each variable  $u_j \in U$ ,  $1 \leq j \leq n$ , we introduce a petal  $H_j(M, 2m)$  for a fixed constant  $M \geq 5$ . Then, we introduce edges  $\{r_i, s_i\}$  and  $\{r_i, t_i\}$  for each  $i$ ,  $1 \leq i \leq m$ , whose weights are  $2M$ .

For each clause  $c_i$ , let  $l_{i1}, l_{i2}, l_{i3}$  be literals of  $c_i$ . Without loss of generality, we assume that  $l_{i1}, l_{i2}, l_{i3}$  are placed in a clockwise direction around  $c_i$ . If  $l_{i1} = u_j$  (or  $l_{i1} = \bar{u}_j$ ), we connect  $r_i, s_i$  and  $H_j(M, 2m)$  such that if  $H_j(M, 2m)$  is oriented in a clockwise direction (respectively, in an anticlockwise direction), the length of  $s_i-r_i$  path via  $H_j(M, 2m)$  is 3. If  $l_{i2} = u_j$  (or  $l_{i2} = \bar{u}_j$ ), we connect  $s_i, t_i$  and  $H_j(M, 2m)$  such that if  $H_j(M, 2m)$  is oriented in an anticlockwise direction (respectively, in a clockwise direction), the length of  $s_i-t_i$  path via  $H_j(M, 2m)$  is  $M + 3$ . If  $l_{i3} = u_j$  (or  $l_{i3} = \bar{u}_j$ ), we connect  $r_i, t_i$  and  $H_j(M, 2m)$  such that if  $H_j(M, 2m)$  is oriented in a clockwise direction (in an anticlockwise direction), the length of  $r_i-t_i$  path via  $H_j(M, 2m)$  is 3. Finally, we replace each edge  $e$  with a path of length  $\omega(e)$ . Remember that  $M$  is a fixed constant, and hence all the weights are fixed constant. By combining all of them, we obtain the graph  $G'$ . Clearly,  $G'$  is planar.

It is easy to see that there is an orientation of  $G'$  such that the length of each  $s_i-t_i$  path is at most  $2M + 3$  if and only if there is a satisfying truth assignment of  $C$ . Therefore, deciding whether  $G'$  has an orientation  $\vec{G}'$  with  $h(\vec{G}') \leq 2M + 3$  is strongly NP-complete.

Similarly, we can show that MIN-SUM ORIENTATION is strongly NP-hard for planar graphs even if all edge-weights are identical.  $\square$

From the above proof, for any positive constant  $\varepsilon$ , both problems admit no polynomial-time  $(2 - \varepsilon)$ -approximation algorithm unless  $P = NP$ .

## 2.2 Cacti

A graph  $G$  is a *cactus* if every edge is part of at most one cycle in  $G^1$ . The class of cacti is a subclass of planar graphs. However, MIN-MAX ORIENTATION is still hard for cacti.

**Theorem 2.** MIN-MAX ORIENTATION is NP-hard for cacti even if  $q = 2$ .

*Proof.* We show that the PARTITION problem (known to be NP-complete<sup>2),4)</sup> can be reduced in polynomial time to the decision version of MIN-MAX ORIENTATION for cacti.

Let  $A$  be a finite set in which each element  $a \in A$  has a positive integer size  $s(a)$ . The PARTITION problem is to decide whether there is a subset  $A' \subset A$  such that  $\sum_{a \in A'} s(a) = \sum_{a \in A \setminus A'} s(a)$ .

From a given instance  $A$  of PARTITION, we now construct a graph  $G$  as the corresponding instance of the decision problem. Let  $n = |A|$ . We introduce vertices  $V = \{v_0, v_1, \dots, v_n\}$  and  $U = \{u_1, \dots, u_n\}$ . We also introduce edges  $E_1 = \{\{v_i, v_{i+1}\} \mid 0 \leq i \leq n - 1\}$ ,  $E_2 = \{\{v_i, u_{i+1}\} \mid 0 \leq i \leq n - 1\}$  and  $E_3 = \{\{u_i, v_i\} \mid 1 \leq i \leq n\}$ . (See Figure 2.) We define the weights of edges, as follows:  $\omega(\{v_{i-1}, v_i\}) = \omega(\{v_{i-1}, u_i\}) = 1$  and  $\omega(\{u_i, v_i\}) = s(a_i)$  for each  $i$ ,  $1 \leq i \leq n$ . Then, let  $G = (V \cup U, E_1 \cup E_2 \cup E_3)$ . Clearly,  $G$  is a simple cactus. Let  $(s_1, t_1) = (v_0, v_n)$  and  $(s_2, t_2) = (v_n, v_0)$ , and hence  $q = 2$ .

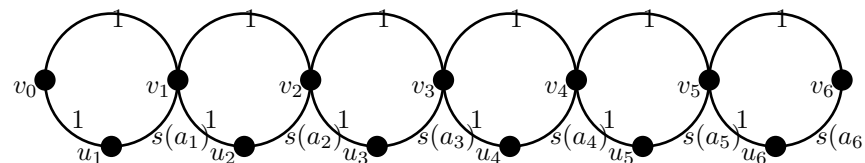


Fig. 2 Cactus in case  $n = 6$

Suppose that  $G$  has an orientation such that  $h(\vec{G}) \leq \frac{1}{2} \sum_{a \in A} s(a) + n$ . Then, the following subset  $A'$  of  $A$  is a feasible solution for PARTITION:

$A' = \{a_i \mid (u_i, v_i) \text{ is in the directed path from } s_0 (= v_0) \text{ to } t_0 (= v_n) \text{ in } \vec{G}\}$ .

Similarly, the opposite holds.  $\square$

## 3. Algorithms for Cycles

The main result of this section is the following theorem.

**Theorem 3.** Both MIN-SUM ORIENTATION and MIN-MAX ORIENTATION can be solved in polynomial time for cycles.

Suppose that we are given an edge-weighted cycle  $C = (V, E)$  and  $q$  pairs  $(s_i, t_i)$ ,  $1 \leq i \leq q$ . Notice that there always exists a feasible orientation for  $C$ : simply orienting  $C$  in a clockwise direction.

Let  $\text{cw}(i)$  be the set of edges in the path on  $C$  from  $s_i$  to  $t_i$  in a clockwise direction, and let  $\text{acw}(i)$  be the set of edges in the path on  $C$  from  $s_i$  to  $t_i$  in an anticlockwise direction. Clearly, for each  $i$ ,  $1 \leq i \leq q$ ,  $\{\text{cw}(i), \text{acw}(i)\}$  is a partition of  $E$ , that is,

$$\text{cw}(i) \cap \text{acw}(i) = \emptyset \quad (1)$$

and

$$\text{cw}(i) \cup \text{acw}(i) = E. \quad (2)$$

We introduce a variable  $x_i$  for each pair  $(s_i, t_i)$ ,  $1 \leq i \leq q$ : if  $x_i = 0$ , then we orient the edges in  $\text{cw}(i)$  in a clockwise direction; if  $x_i = 1$ , then we orient the edges in  $\text{acw}(i)$  in an anticlockwise direction.

For two pairs  $(s_i, t_i)$  and  $(s_j, t_j)$ , it is easy to see that the following constraint holds on the two corresponding variables  $x_i$  and  $x_j$ : if  $\text{cw}(i) \cap \text{acw}(j) \neq \emptyset$ , then  $x_i = x_j$ ; if  $\text{cw}(i) \cap \text{acw}(j) = \emptyset$ , then  $x_i \leq x_j$ ; and if  $\text{acw}(i) \cap \text{cw}(j) = \emptyset$ , then  $x_i \geq x_j$ . We now construct a *constraint graph*  $\mathcal{C}$  in which each vertex  $v_i$  corresponds to a pair  $(s_i, t_i)$  (and hence to a variable  $x_i$ ) and there is an edge between two vertices  $v_i$  and  $v_j$  if and only if

$$\text{cw}(i) \cap \text{acw}(j) \neq \emptyset, \quad (3)$$

that is,  $x_i$  and  $x_j$  have the constraint  $x_i = x_j$ . Clearly, from a feasible orientation of  $\mathcal{C}$ , we can obtain an assignment of  $\{0, 1\}$  to each variable  $x_k$ ,  $1 \leq k \leq q$ , such that any two variables satisfy their constraint; and hence two variables  $x_i$  and  $x_j$  receive the same value if their corresponding vertices  $v_i$  and  $v_j$  are contained in the same (connected) component of  $\mathcal{C}$ . Note that  $v_i$  and  $v_j$  are not necessarily adjacent; in this case, either  $x_i \leq x_j$  or  $x_i \geq x_j$  holds, and  $\mathcal{C}$  contains a path between  $v_i$  and  $v_j$ .

Let  $\mathcal{V} = \{V_1, V_2, \dots, V_m\}$  be the partition of the vertex set of  $\mathcal{C}$  in which each  $V_i$ ,  $1 \leq i \leq m$ , forms a component of  $\mathcal{C}$ . Then, we define a relation “ $\leq$ ” on the set  $\mathcal{V}$ , as follows:  $V_i \leq V_j$  if and only if there exists two vertices  $v_i \in V_i$  and  $v_j \in V_j$  such that their corresponding variables  $x_i$  and  $x_j$  have the constraint  $x_i \leq x_j$ . We will show later in Lemma 1 that  $\mathcal{V}$  is totally ordered under the relation  $\leq$ . Then, it is easy to see that both MIN-SUM ORIENTATION and MIN-MAX ORIENTATION can be solved in polynomial time.

**Lemma 1.**  $\mathcal{V}$  is totally ordered under the relation  $\leq$ .

*Proof.* Consider any two subsets  $V_i$  and  $V_j$  in  $\mathcal{V}$  such that  $V_i \neq V_j$ . We will show that exactly one of  $V_i \leq V_j$  and  $V_i \geq V_j$  holds. It suffices to show that, for any two vertices  $v_{i_1}$  and  $v_{i_2}$  in  $V_i$  and a vertex  $v_j$  in  $V_j$ , their corresponding variables have exactly one of the following two constraints: (a)  $x_{i_1} \leq x_j$  and  $x_{i_2} \leq x_j$ ; or (b)  $x_{i_1} \geq x_j$  and  $x_{i_2} \geq x_j$ .

Suppose for a contradiction that the variables have the constraints  $x_{i_1} \leq x_j$  and  $x_{i_2} \geq x_j$ ; it is similar for the case  $x_{i_1} \geq x_j$  and  $x_{i_2} \leq x_j$ . Then, we have

$$\text{cw}(i_1) \cap \text{acw}(j) = \emptyset, \quad (4)$$

and

$$\text{acw}(i_2) \cap \text{cw}(j) = \emptyset. \quad (5)$$

There are the following two cases to consider.

**Case (i):  $v_{i_1}$  and  $v_{i_2}$  are adjacent in  $\mathcal{C}$**

In this case, there is a constraint  $x_{i_1} = x_{i_2}$ , and hence we have  $\text{cw}(i_1) \cap \text{acw}(i_2) \neq \emptyset$ . Let

$$e \in \text{cw}(i_1) \cap \text{acw}(i_2), \quad (6)$$

then by Eq. (4) we have  $e \notin \text{acw}(j)$ . By Eqs. (1) and (2) we have  $e \in \text{cw}(j)$ , and hence by Eq. (6) we have  $e \in \text{acw}(i_2) \cap \text{cw}(j) \neq \emptyset$ . This contradicts Eq. (5).

**Case (ii):  $v_{i_1}$  and  $v_{i_2}$  are not adjacent in  $\mathcal{C}$**

In this case,  $\mathcal{C}$  contains a path between  $v_{i_1}$  and  $v_{i_2}$ , whose length is more than one. Then, there is a vertex  $v_h$  in  $V_i$  which is adjacent with  $v_{i_1}$ , and hence by Eq. (3) we have  $\text{cw}(i_1) \cap \text{acw}(h) \neq \emptyset$ . Let

$$e \in \text{cw}(i_1) \cap \text{acw}(h), \quad (7)$$

then by Eq. (4) we have  $e \notin \text{acw}(j)$ . By Eqs. (1) and (2) we have  $e \in \text{cw}(j)$ , and hence by Eq. (7) we have  $e \in \text{cw}(j) \cap \text{acw}(h) \neq \emptyset$ . Therefore, by Eq. (3) there is an edge between  $v_j$  and  $v_h$ . This contradicts that  $V_i \neq V_j$ .  $\square$

#### 4. Algorithms for Cactus

In this section, we deal with the cactus. Though the MIN-MAX ORIENTATION is NP-hard for cacti, the MIN-SUM ORIENTATION is solvable in polynomial-time. We also show a polynomial-time 2-approximation algorithm and a pseudo-polynomial-time algorithm for the MIN-MAX ORIENTATION.

##### 4.1 MIN-SUM ORIENTATION

By extending Theorem 3, we obtain the following theorem.

**Theorem 4.** *The MIN-SUM ORIENTATION can be solved in  $O(|V|q^2)$  time for cacti.*

*Proof.* The objective value of the MIN-SUM ORIENTATION for the given cactus is the sum of lengths of directed  $s_i$ - $t_i$  shortest paths for  $i = 1, \dots, q$ . Because each elementary path on a cactus is composed of bridges and subpaths on elementary cycles, the length of the shortest path from  $s_i$  to  $t_i$  is the sum of the length of bridges and subpaths on elementary cycles. Then the objective value is decomposed to

$$\sum_{e \in \text{bridges of } G} \omega(e) \times |\{(s_i, t_i) \mid s_i-t_i \text{ path must include } e, i = 1, \dots, q\}|$$

$$+ \sum_{c \in \text{cycles of } G} \min \sum_{i=1}^q (\text{the length of } s_i-t_i \text{ subpath on } c)$$

The first term of the above equation is automatically determined in  $O(nq)$  time. The second term of the above equation is calculated by solving the MIN-SUM ORIENTATION of each cycle independently. The MIN-SUM ORIENTATION of each cycle terminates in  $O(|V|q^2)$  time from Theorem 3. Then the MIN-SUM ORIENTATION can be solved in

$$O\left(nq + \sum_{c \in \text{cycles of } G} |c|q^2\right) = O(|V|q^2)$$

time, where  $|c|$  is the number of vertices of the cycle  $c$ .  $\square$

In the following of this section, we discuss the MIN-MAX ORIENTATION that is NP-hard from Theorem 2.

#### 4.2 Approximation Algorithm Using Linear Programming

Let  $C$  be a set of elementary cycles of the given cactus. Let  $C_i \subseteq C$  be a set of elementary cycles that each elementary (undirected)  $s_i$ - $t_i$  path must have an intersection. Let  $d_i$  be the sum of lengths of bridges that each elementary (undirected)  $s_i$ - $t_i$  path must include. Note that both  $C_i$  and  $d_i$  of each  $i$  are uniquely determined.

Let  $a_i^c$ ,  $c \in C_i$ ,  $i = 1, \dots, q$  be the sum of lengths of edges that both the cycle  $c$  and an  $s_i$ - $t_i$  path passing through  $c$  clockwise include. Let  $b_i^c$ ,  $c \in C_i$ ,  $i = 1, \dots, q$  be the sum of lengths of edges that both the cycle  $c$  and an  $s_i$ - $t_i$  path passing through  $c$  anticlockwise include. We call  $\{i, j\}$  is a *conflicting pair* on  $c$  if an  $s_i$ - $t_i$  path passing through  $c$  clockwise and an  $s_j$ - $t_j$  path passing through  $c$  anticlockwise share a common (undirected) edge.

We introduce two kinds of  $\{0, 1\}$ -variables  $x_i^c$  and  $y_i^c$ . If there is a directed  $s_i$ - $t_i$  path passing through  $c$  clockwise,  $x_i^c = 1$  otherwise  $x_i^c = 0$ . If there is a directed  $s_i$ - $t_i$  path passing through  $c$  anticlockwise,  $y_i^c = 1$  otherwise  $y_i^c = 0$ .

The MIN-MAX ORIENTATION of a given cactus is formulated as follows:

$$\text{minimize } z \tag{8}$$

$$\text{subject to } x_i^c + y_j^c \leq 1, \forall \{i, j\} \in \text{conflicting pairs on } c, \forall c \in C, \tag{9}$$

$$x_i^c + y_i^c = 1, \forall c \in C_i, i = 1, \dots, q, \tag{10}$$

$$\sum_{c \in C_i} (a_i^c x_i^c + b_i^c y_i^c) + d_i \leq z, i = 1, \dots, q, \tag{11}$$

$$x_i^c, y_i^c \in \{0, 1\}, \forall c \in C_i, i = 1, \dots, q. \tag{12}$$

The size of the above integer programming formulation is polynomial. The linear relaxation problem of the integer programming is solvable in polynomial-time.

We propose an algorithm using linear programming for cacti. Our algorithm is very simple. At first, we solve the linear relaxation problem. And then we round each variable as follows:

$$x_i^c \text{ is set to } \begin{cases} 1, & x_i^c \geq 0.5, \\ 0, & x_i^c < 0.5, \end{cases} \quad y_i^c \text{ is set to } \begin{cases} 1, & y_i^c > 0.5, \\ 0, & y_i^c \leq 0.5. \end{cases}$$

The algorithm clearly terminates in polynomial-time.

The algorithm yields a feasible solution. The rounding procedure clearly satisfies (9), (10), (12).

Because the value of each variable increases at most twice by the rounding procedure, the left side terms of (11) increase at most twice. Then, the objective value increases at most twice of the optimum value of the linear relaxation problem. Because the optimum value of the linear relaxation is the lower bound of the MIN-MAX ORIENTATION, the above algorithm is a polynomial-time 2-approximation algorithm.

#### 4.3 Pseudo-Polynomial-Time algorithm

Here, we show a pseudo-polynomial-time algorithm for the MIN-MAX ORIENTATION of cacti. Let  $G = (V, E)$  be a cactus. We add a root vertex to  $G$ , and connect the root and a vertex of  $G$ . A vertex  $v$  with the property that  $G \setminus \{v\}$  is not connected is called an *articulation* vertex. Let  $\text{desc}(v)$  be a set of vertices such that any path from these vertices to the root include  $v$ . Intuitively,  $\text{desc}(v)$  are descendants of  $v$  of the rooted graph  $G$ . Note that  $\text{desc}(v)$  includes  $v$  itself. For

an articulation vertex  $v$ , we define  $G_v$  as a subgraph induced by  $\text{desc}(v)$ . Because  $G$  is a cactus,  $\{G_v \mid v \text{ is articulation}\}$  is laminar<sup>5</sup>). The key idea of our pseudo-polynomial-time algorithm is a dynamic programming on  $\{G_v \mid v \text{ is articulation}\}$  in the bottom up manner.

For a subgraph  $G_v$  and  $q$ -tuple of integers  $(x_1, x_2, \dots, x_q)$ , an orientation of  $G_v$  is called a  $(x_1, x_2, \dots, x_q)$ -orientation if the following three conditions are satisfied:

- (1)  $f(G_v, s_i, t_i) \leq x_i$  for all  $(s_i, t_i)$  such that  $s_i \in \text{desc}(v), t_i \in \text{desc}(v)$ ;
- (2)  $f(G_v, s_i, v) \leq x_i$  for all  $(s_i, t_i)$  such that  $s_i \in \text{desc}(v), t_i \notin \text{desc}(v)$ ; and
- (3)  $f(G_v, v, t_i) \leq x_i$  for all  $(s_i, t_i)$  such that  $s_i \notin \text{desc}(v), t_i \in \text{desc}(v)$ .

For a subgraph  $G_v$  and  $q$ -tuple  $(x_1, x_2, \dots, x_q)$ , we define

$$g(G_v, x_1, x_2, \dots, x_q) = \begin{cases} 0 & \text{if } G_v \text{ has } (x_1, x_2, \dots, x_q)\text{-orientation,} \\ \infty & \text{otherwise.} \end{cases}$$

Filling  $g(G_v, x_1, x_2, \dots, x_q)$  with dynamic programming, we obtain the optimal solution of the MIN-MAX ORIENTATION. We omit the detail of the dynamic programming since it is clear.

## References

- 1) Brandstädt, A., Le, V. and Spinrad, J.: *Graph Classes: A Survey*, SIAM (1999).
- 2) Garey, M.R. and Johnson, D.S.: *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W. H. Freeman and Company (1979).
- 3) Hakimi, S.L., Schmeichel, E.F. and Young, N.E.: Orienting Graphs to Optimize Reachability, *Information Processing Letters*, Vol.63, pp.229–235 (1997).
- 4) Karp, R.M.: Reducibility Among Combinatorial Problems, *Complexity of Computer Computations* (Miller, R.E. and Thatcher, J.W., eds.), Plenum Press, New York, pp.85–103 (1972).
- 5) Korte, B. and Vygen, J.: *Combinatorial Optimization: Theory and Algorithms*, Algorithms and Combinatorics, Vol.21, Springer-Verlag, 4th edition (2007).
- 6) Lichtenstein, D.: Planar Formulae and Their Uses, *SIAM Journal on Computing*, Vol.11, pp.329–343 (1982).
- 7) Robbins, H.E.: A theorem on graphs with an application to a problem of traffic control, *American Mathematical Monthly*, Vol.46, pp.281–283 (1939).