Route-Enabling Graph Orientation Problems*

Takehiro Ito[1], Yuichiro Miyamoto[2],
Hirotaka Ono[3], Hisao Tamaki[4], and Ryuhei Uehara[5]

[1] Graduate School of Information Sciences, Tohoku University,
Aoba-yama 6-6-05, Sendai, 980-8579, Japan
takehiro@ecei.tohoku.ac.jp

[2] Faculty of Science and Technology, Sophia University,
Kioi-cho 7-1, Chiyoda-ku, Tokyo, 102-8554, Japan
miyamoto@sophia.ac.jp

[3] Graduate School of Information Science and Electrical Engineering,
Kyushu University, 744 Motooka, Nishi-ku, Fukuoka, 819-0395, Japan
ono@csce.kyushu-u.ac.jp

[4] School of Science and Technology, Meiji University,
Higashi-mita 1-1-1, Tama-ku, Kawasaki-shi, Kanagawa, 214-8571, Japan
tamaki@cs.meiji.ac.jp

[5] School of Information Science, JAIST,
Asahidai 1-1, Nomi, Ishikawa 923-1292, Japan
uehara@jaist.ac.jp

Abstract. Given an undirected and edge-weighted graph G together with a set of ordered vertex-pairs, called st-pairs, we consider the problems of finding an orientation of all edges in G: MIN-SUM ORIENTATION is to minimize the sum of the shortest directed distances between all st-pairs; and MIN-MAX ORIENTATION is to minimize the maximum shortest directed distance among all st-pairs. In this paper, we first show that both problems are strongly NP-hard for planar graphs even if all edge-weights are identical, and that both problems can be solved in polynomial time for cycles. We then consider the problems restricted to cacti, which form a graph class that contains trees and cycles but is a subclass of planar graphs. Then, MIN-SUM ORIENTATION is solvable in polynomial time, whereas MIN-MAX ORIENTATION remains NP-hard even for two st-pairs. However, based on LP-relaxation, we present a polynomial-time 2-approximation algorithm for MIN-MAX ORIENTATION. Finally, we give a fully polynomial-time approximation scheme (FPTAS) for MIN-MAX ORIENTATION on cacti if the number of st-pairs is a fixed constant.

1 Introduction

Consider the situation in which we wish to assign one-way restrictions to (narrow) aisles in a limited area, such as in an industrial factory, with keeping reachability between several sites. Since traffic jams rarely occur in industrial factories, the distances of routes between important sites are of great interest for the

* This work is partially supported by Grant-in-Aid for Scientific Research: 20650002, 20700003 and 21680001.

Y. Dong, D.-Z. Du, and O. Ibarra (Eds.): ISAAC 2009, LNCS 5878, pp. 403–412, 2009.

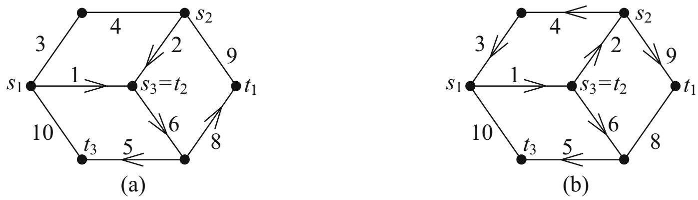

Fig. 1. (a) Solution for MIN-SUM ORIENTATION and (b) solution for MIN-MAX ORIENTATION

efficiency. This situation frequently appears in the context of the scheduling of automated guided vehicles without collision [5]. In this paper, we model the situation as graph orientation problems, in which we wish to find an orientation so that the distances of (directed) routes are not so long for given multiple st-pairs.

Let $G = (V, E)$ be an undirected graph together with an assignment of a non-negative integer, called the *weight* $\omega(e)$, to each edge e in G. Assume that we are given q ordered vertex-pairs (s_i, t_i), $1 \le i \le q$, called *st-pairs*. Then, an *orientation* of G is an assignment of exactly one direction to each edge in G so that there exists a directed (s_i, t_i)-path (i.e., a directed path from s_i to t_i) for every st-pair (s_i, t_i), $1 \le i \le q$. For an orientation $\boldsymbol{G}$ of G and an st-pair (s_i, t_i), we denote by $\omega(\boldsymbol{G}, s_i, t_i)$ the total weight of a shortest directed (s_i, t_i)-path on $\boldsymbol{G}$, that is, $\omega(\boldsymbol{G}, s_i, t_i) = \min\{\omega(P) \mid P \text{ is a directed } (s_i, t_i)\text{-path on } \boldsymbol{G}\}$ where $\omega(P)$ is the sum of weights of all edges in a path P.

We introduce two objective functions for orientations $\boldsymbol{G}$ of a graph G, and study the corresponding two minimization problems. The first objective is SUM-type, defined as follows: $g(\boldsymbol{G}) = \sum_{1 \le i \le q} \omega(\boldsymbol{G}, s_i, t_i)$. Its corresponding problem, called the MIN-SUM ORIENTATION problem, is to find an orientation $\boldsymbol{G}$ of G such that $g(\boldsymbol{G})$ is minimum; we denote by $g^*(G)$ the optimal value for G. The second objective is MAX-type, defined as follows: $h(\boldsymbol{G}) = \max\{\omega(\boldsymbol{G}, s_i, t_i) \mid 1 \le i \le q\}$. Its corresponding problem, called the MIN-MAX ORIENTATION problem, is to find an orientation $\boldsymbol{G}$ of G such that $h(\boldsymbol{G})$ is minimum; we denote by $h^*(G)$ the optimal value for G. For the sake of convenience, let $g^*(G) = +\infty$ and $h^*(G) = +\infty$ if G has no orientation for a given set of st-pairs. Clearly, both problems can be solved in polynomial time if we are given a single st-pair (s_1, t_1); in this case, we simply seek a shortest path between s_1 and t_1.

Figure 1 illustrates two orientations of the same graph G for the same set of st-pairs, where the weight $\omega(e)$ is attached to each edge e and the direction assigned to an edge is indicated by an arrow (but the direction is not indicated if the edge is not used in any shortest directed (s_i, t_i)-path, $1 \le i \le 3$). The orientation $\boldsymbol{G}$ in Fig.1(a) is an optimal solution for MIN-SUM ORIENTATION, where $g^*(G) = g(\boldsymbol{G}) = (1+6+8)+2+(6+5) = 28$. On the other hand, Fig.1(b) illustrates an optimal solution for MIN-MAX ORIENTATION, in which the st-pair (s_1, t_1) has the maximum distance; $h^*(G) = \max\{1+2+9,\ 4+3+1,\ 6+5\} = 12$.

Robbins [7] showed that every 2-edge-connected graph can be directed so that the resulting digraph is strongly connected. Therefore, a graph G has at least one orientation for any set of st-pairs if G is 2-edge-connected. Chvátal

Table 1. Summary of our results

	MIN-SUM ORIENTATION	MIN-MAX ORIENTATION
planar graphs	• strongly NP-hard • no $(2-\varepsilon)$-approximation	• strongly NP-hard • no $(2-\varepsilon)$-approximation
cacti	$O(nq^2)$	• NP-hard even for $q = 2$ • polynomial-time 2-approximation • FPTAS for a fixed constant q
cycles	$O(n + q^2)$	$O(n + q^2)$

and Thomassen [2] showed that it is NP-complete to determine whether a given unweighted graph can be directed so that the resulting digraph is strongly connected and whose (directed) diameter is 2. This implies that our MIN-MAX ORIENTATION is NP-hard in general. On the other hand, Hakimi *et al.* [4] proposed a quadratic algorithm for the problem of directing a 1-edge-connected graph so as to maximize the number of ordered vertex-pairs (x, y) having a directed (x, y)-path. The problem of [4] can be easily reduced to our MIN-SUM ORIENTATION.

In this paper, we mainly give the following three results. (Table1 summarizes our results, where n is the number of vertices in a graph.) The first is to show that both problems are strongly NP-hard for planar graphs even if all edge-weights are identical. We remark that the known result of [2] does not imply NP-completeness for planar graphs. The second is to show that both problems can be solved in polynomial time for cycles. By extending the algorithm for cycles, we show that MIN-SUM ORIENTATION is solvable in polynomial time for cacti, whereas MIN-MAX ORIENTATION remains NP-hard even for cacti with $q = 2$. (Cacti form a graph class that contains trees and cycles, but is a subclass of planar graphs.) The third is to give a fully polynomial-time approximation scheme (FPTAS) for MIN-MAX ORIENTATION on cacti if q is a fixed constant.

In addition, we give several results on the way to the three main results above. Firstly, our proof of strong NP-hardness implies that, for any constant $\varepsilon > 0$, both problems admit no polynomial-time $(2-\varepsilon)$-approximation algorithm unless P = NP. Secondly, in order to obtain a lower bound and an upper bound on $h^*(G)$ for a cactus G, we present a polynomial-time 2-approximation algorithm based on LP-relaxation; we remark that q is not required to be a fixed constant for this 2-approximation algorithm. We finally remark that our complexity analysis for MIN-MAX ORIENTATION on cacti is tight in some sense: the problem is in P if $q = 1$, and the problem for cacti cannot be strongly NP-hard if q is a fixed constant because our third result gives an FPTAS for the problem [6, p. 307].

2 Computational Hardness

In this section, we first show that our two problems are both strongly NP-hard for planar graphs, and then show that MIN-MAX ORIENTATION remains NP-hard even for cacti with $q = 2$.

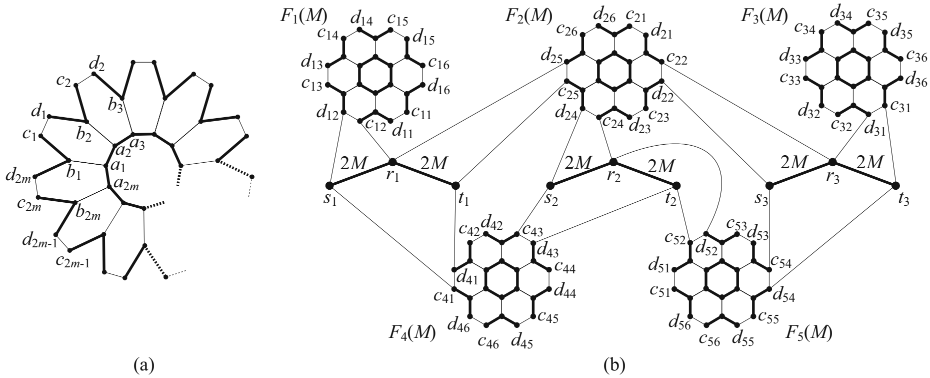

Fig. 2. (a) Flower gadget $F_i(M)$, and (b) planar graph G_ϕ corresponding to a Boolean formula ϕ with three clauses $c_1 = (u_1 \vee \bar{u}_2 \vee u_4)$, $c_2 = (u_2 \vee u_5 \vee u_4)$ and $c_3 = (u_2 \vee \bar{u}_3 \vee \bar{u}_5)$

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

Proof. We show that the PLANAR 3-SAT problem, which is known to be strongly NP-complete [3,6], can be reduced in polynomial time to MIN-MAX ORIENTATION. (The reduction to MIN-SUM ORIENTATION is similar.)

In PLANAR 3-SAT, we are given a Boolean formula ϕ in conjunctive normal form, say with set U of n variables $u_1, u_2, \ldots, u_n$ and set C of m clauses $c_1, c_2, \ldots, c_m$, such that each clause $c_j \in C$ contains exactly three literals and the following bipartite graph $B = (V', E')$ is planar: $V' = U \cup C$ and E' contains exactly those pairs $\{u_i, c_j\}$ such that either u_i or $\bar{u}_i$ appears in c_j. The PLANAR 3-SAT problem is to determine whether there is a satisfying truth assignment for ϕ. Given an instance of PLANAR 3-SAT, we construct the corresponding instance of MIN-MAX ORIENTATION. We first make a *flower gadget* $F_i(M)$ for each variable $u_i \in U$, and then construct the whole graph G_ϕ corresponding to ϕ.

We first define a flower gadget $F_i(M)$. Let M be a fixed constant (integer) such that $M \geq 3$. The flower gadget $F_i(M) = (V_i, E_i)$ for a variable $u_i \in U$ consists of $2m$ hexagonal elementary cycles, as illustrated in Fig.2(a). (Remember that m is the number of clauses in ϕ.) More precisely, $V_i = \{a_k, b_k, c_k, d_k \mid 1 \leq k \leq 2m\}$ and $E_i = \{\{a_{k+1}, a_k\}, \{a_k, b_k\}, \{b_k, c_k\}, \{c_k, d_k\}, \{d_k, b_{k+1}\} \mid 1 \leq k \leq 2m\}$, where $a_{2m+1} = a_1$ and $b_{2m+1} = b_1$. The edge-weights are defined as follows: for each k, $1 \leq k \leq 2m$, $\omega(\{a_{k+1}, a_k\}) = \omega(\{b_k, c_k\}) = \omega(\{d_k, b_{k+1}\}) = M$ and $\omega(\{a_k, b_k\}) = \omega(\{c_k, d_k\}) = 1$. (In Fig.2(a), the weight-M edges are depicted by thick lines.) Finally, we define the set ST_i of $12m$ *st*-pairs, as follows:

$$ST_i = \{(a_k, d_k), (d_k, a_k), (b_k, b_{k+1}), (b_{k+1}, b_k), (c_k, a_{k+1}), (a_{k+1}, c_k) \mid 1 \leq k \leq 2m\}.$$

For each k, $1 \leq k \leq 2m$, the kth hexagonal elementary cycle $a_k b_k c_k d_k b_{k+1} a_{k+1}$ is called the *kth petal* P_k; P_k is called an *odd petal* if k is odd, while is called an *even petal* if k is even. We call the edge $\{c_k, d_k\}$ in each petal P_k, $1 \leq k \leq 2m$, an *external edge* of P_k. For the sake of convenience, we fix the embedding of $F_i(M)$ such that the outer face consists of b_k, c_k, d_k, $1 \le k \le 2m$, which are placed in a clockwise direction, as illustrated in Fig.2(a).

It is easy to see that $F_i(M)$ has only two optimal orientations for ST_i: the one is to direct each odd petal in a clockwise direction and to direct each even petal in an anticlockwise direction; and the other is the reversed one. In the first optimal orientation, the external edges $\{c_k, d_k\}$ are directed from c_k to d_k in all odd petals P_k, while directed from d_k to c_k in all even petals; we call this optimal orientation of $F_i(M)$ a *true-orientation*, which corresponds to assigning TRUE to the variable u_i. On the other hand, the other optimal orientation of $F_i(M)$ is called a *false-orientation*, which corresponds to assigning FALSE to u_i. Clearly, $h^*(F_i(M)) = 2M + 1$.

We now construct the planar graph G_ϕ corresponding to the formula ϕ, as follows. We fix an embedding of the bipartite graph $B = (V', E')$ arbitrarily. For each variable u_i, $1 \le i \le n$, we replace it with a flower gadget $F_i(M)$. For each clause c_j, $1 \le j \le m$, we replace it with a path consisting of three vertices s_j, r_j, t_j; let $\omega(\{s_j, r_j\}) = \omega(\{r_j, t_j\}) = 2M$. We then connect flower gadgets $F_i(M)$, $1 \le i \le n$, with paths $s_j r_j t_j$, $1 \le j \le m$, as follows. For each clause c_j, $1 \le j \le m$, let l_{j1}, l_{j2}, l_{j3} be three literals in c_j, and assume without loss of generality that three flower gadgets corresponding to l_{j1}, l_{j2}, l_{j3} are placed in a clockwise direction around the path $s_j r_j t_j$ corresponding to c_j. Assume that l_{jk} is either u_i or $\bar{u}_i$. Then, we replace the edge of B joining variable u_i and clause c_j with a pair of weight-1 edges which, together with an external edge in $F_i(M)$, forms a path between two vertices chosen from $\{s_j, r_j, t_j\}$, according to the following rules (see Fig.2(b) as an example):

(i) The endpoints of this path are s_j and r_j if $k = 1$; r_j and t_j if $k = 2$; and s_j and t_j if $k = 3$.
(ii) The external edge is from an even petal if $l_{j1} = u_i$, $l_{j2} = u_i$, or $l_{j3} = \bar{u}_i$; while it is from an odd petal if $l_{j1} = \bar{u}_i$, $l_{j2} = \bar{u}_i$, or $l_{j3} = u_i$.
(iii) From the viewpoint of variable u_i, we choose a distinct external edge for each clause containing u_i, honoring the order of those clauses around u_i and thereby preserving the planarity of the embedding.

Finally, we replace each edge e in G_ϕ with a path of length $\omega(e)$ in which all edges are of weight 1. (Remember that M is a fixed constant.) Clearly, the resulting graph G_ϕ is planar, and can be constructed in polynomial time. The set of all st-pairs in this instance is defined as follows: $(\bigcup_{i=1}^{n} ST_i) \cup \{(s_j, t_j) \mid 1 \le j \le m\}$. Therefore, there are $(12mn+m)$ st-pairs in total. This completes the construction of the corresponding instance of MIN-MAX ORIENTATION.

Then, deciding whether $h^*(G_\phi) \le 2M + 3$ is equivalent to solving PLANAR 3-SAT for ϕ. (We omit the details due to the page limitation.) □

From our proof of Theorem 1, we immediately obtain the following corollary.

Corollary 1. *For any constant* $\varepsilon > 0$*, both* MIN-SUM ORIENTATION *and* MIN-MAX ORIENTATION *admit no polynomial-time* $(2 - \varepsilon)$*-approximation algorithm for planar graphs unless* $\mathrm{P} = \mathrm{NP}$.

A graph G is a *cactus* if every edge is part of at most one cycle in G [1,8]. Cacti form a subclass of planar graphs. However, we have the following theorem.

Theorem 2. Min-max orientation *is NP-hard for cacti even if* $q = 2$.

3 Polynomial-Time Algorithms

The main result of this section is the following theorem.

Theorem 3. *Both* min-sum orientation *and* min-max orientation *can be solved in time* $O(n + q^2)$ *for a cycle* C*, where* n *is the number of vertices in* C*.*

Proof. Suppose that we are given an edge-weighted cycle $C = (V, E)$ and q *st*-pairs (s_i, t_i), $1 \le i \le q$. Note that C has at least one orientation for any set of *st*-pairs: simply directing C in a clockwise direction.

For each *st*-pair (s_i, t_i), $1 \le i \le q$, let $\mathrm{cw}(i)$ be the set of all edges in the directed (s_i, t_i)-path when all edges in C are directed in a clockwise direction, and let $\mathrm{acw}(i)$ be the set of all edges in the directed (s_i, t_i)-path when all edges in C are directed in an anticlockwise direction. Clearly, for each i, $1 \le i \le q$, $\{\mathrm{cw}(i), \mathrm{acw}(i)\}$ is a partition of E, that is, $\mathrm{cw}(i) \cap \mathrm{acw}(i) = \emptyset$ and $\mathrm{cw}(i) \cup \mathrm{acw}(i) = E$. We introduce a $\{0, 1\}$-variable x_i for each *st*-pair (s_i, t_i), $1 \le i \le q$: if $x_i = 0$, then the edges in $\mathrm{cw}(i)$ are directed in a clockwise direction; if $x_i = 1$, then the edges in $\mathrm{acw}(i)$ are directed in an anticlockwise direction. For two *st*-pairs (s_i, t_i) and (s_j, t_j), it is easy to see that the two corresponding variables x_i and x_j have the following constraints (a)–(c):

(a) if $\mathrm{cw}(i) \cap \mathrm{acw}(j) \ne \emptyset$ and $\mathrm{acw}(i) \cap \mathrm{cw}(j) \ne \emptyset$, then $x_i = x_j$;
(b) if $\mathrm{cw}(i) \cap \mathrm{acw}(j) = \emptyset$ and $\mathrm{acw}(i) \cap \mathrm{cw}(j) \ne \emptyset$, then $x_i \le x_j$; and
(c) if $\mathrm{cw}(i) \cap \mathrm{acw}(j) \ne \emptyset$ and $\mathrm{acw}(i) \cap \mathrm{cw}(j) = \emptyset$, then $x_i \ge x_j$.

We now construct a *constraint graph* $\mathcal{C}$ in which each vertex v_i corresponds to an *st*-pair (s_i, t_i) and there is an edge between two vertices v_i and v_j if and only if $\mathrm{cw}(i) \cap \mathrm{acw}(j) \ne \emptyset$ and $\mathrm{acw}(i) \cap \mathrm{cw}(j) \ne \emptyset$, that is, the corresponding variables x_i and x_j have the constraint $x_i = x_j$. From an orientation of C, we can obtain an assignment of $\{0, 1\}$ to each variable x_k, $1 \le k \le q$; clearly, 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 $\mathcal{C}$.

Let $\mathcal{V} = \{V_1, V_2, \ldots, V_m\}$ be the partition of the vertex set of $\mathcal{C}$ such that each V_i, $1 \le i \le m$, forms a connected component of $\mathcal{C}$. Then, we define a relation "$\le$" on $\mathcal{V}$, as follows: $V_i \le V_j$ if and only if there exist 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 $x_i \le x_j$. We show that $\mathcal{V}$ is totally ordered under the relation $\le$. (However, its proof is omitted from this extended abstract.) Then, for some index k, $1 \le k \le m$, we have $x_i = 0$ for all variables x_i whose corresponding vertices are contained in V_j with $V_j \le V_k$; otherwise $x_i = 1$. Therefore, both min-sum orientation and min-max orientation can be reduced simply to finding such an appropriate index k on $\mathcal{V} = \{V_1, V_2, \ldots, V_m\}$. It is now easy to see that both problems can be solved in time $O(n + q^2)$. □

By extending Theorem 3, we can easily obtain the following theorem.

Theorem 4. MIN-SUM ORIENTATION *can be solved in time $O(nq^2)$ for a cactus G, where n is the number of vertices in G.*

4 FPTAS for MIN-MAX ORIENTATION on Cacti

In contrast to MIN-SUM ORIENTATION, MIN-MAX ORIENTATION remains NP-hard even for cacti with $q = 2$. However, in this section, we give an FPTAS for MIN-MAX ORIENTATION on cacti if q is a fixed constant.

In Section 4.1 we first present a polynomial-time 2-approximation algorithm based on LP-relaxation, which gives us both lower and upper bounds on $h^*(G)$ for a given cactus G. We then show in Section 4.2 that the problem can be solved in pseudo-polynomial time for cacti. In Section 4.3, we finally give our FPTAS based on the algorithm in Section 4.2 and using the lower and upper bounds on $h^*(G)$ obtained in Section 4.1.

It can be easily determined in time $O(nq)$ whether a given cactus $G = (V, E)$ has an orientation for the given set of st-pairs; we simply check the placements of st-pairs which pass through each bridge in G. Therefore, we may assume without loss of generality that G has at least one orientation, and hence $h^*(G) \neq +\infty$.

[Cactus and its underlay tree]
A cactus G can be represented by an *underlay tree* T, which is a rooted tree and can be easily obtained from G in a straightforward way. In the underlay tree T of G, each node represents either a bridge of G or an elementary cycle of G; and if there is an edge between nodes u and v of T, then bridges or cycles of G represented by u and v share exactly one vertex in G in common. (A similar idea can be found in [8, Theorem 11].) Each node v of T corresponds to a subgraph G_v of G induced by all bridges and cycles represented by the nodes that are descendants of v in T. Clearly, G_v is a cactus for each node v of T, and $G = G_r$ for the root r of T. It is easy to see that an underlay tree T of a given cactus G can be found in linear time, and hence we may assume that a cactus G and its underlay tree T are given. In Section 4.2, we solve MIN-MAX ORIENTATION by a dynamic programming approach based on the underlay tree T of G.

4.1 2-Approximation Algorithm Based on LP-Relaxation

In this subsection, we give the following theorem. It should be noted that the number q of st-pairs is not required to be a fixed constant in the theorem.

Theorem 5. *There is a polynomial-time 2-approximation algorithm for* MIN-MAX ORIENTATION *on cacti.*

For each st-pair (s_i, t_i), $1 \leq i \leq q$, let C_i be the set of elementary cycles represented by the nodes which are on the path from v_{s_i} to v_{t_i} in the underlay tree T of a given cactus G, where v_{s_i} and v_{t_i} are the nodes in T containing s_i and t_i, respectively. Let d_i be the sum of weights of bridges represented by the nodes which are on the path from v_{s_i} to v_{t_i} in T. Clearly, both C_i and d_i can be computed in time $O(nq)$ for all st-pairs (s_i, t_i), $1 \le i \le q$.

Consider the following two orientations of G: the one, denoted by $\boldsymbol{G}^a$, directs all elementary cycles in G in a clockwise direction; the other, denoted by $\boldsymbol{G}^b$, directs all elementary cycles in G in an anticlockwise direction. Clearly, both $\boldsymbol{G}^a$ and $\boldsymbol{G}^b$ are (feasible) orientations of G. For an st-pair (s_i, t_i), $1 \le i \le q$, and each elementary cycle $c \in C_i$, we denote by a_i^c and b_i^c the sums of weights of the edges which are contained in c and are in the directed (s_i, t_i)-paths on $\boldsymbol{G}^a$ and $\boldsymbol{G}^b$, respectively. For each elementary cycle c in G, we call an ordered index-pair (i, j), $1 \le i, j \le q$, a *conflicting pair on c* if the directed (s_i, t_i)-path on $\boldsymbol{G}^a$ and the directed (s_j, t_j)-path on $\boldsymbol{G}^b$ share at least one edge of c in common.

For an st-pair (s_i, t_i), $1 \le i \le q$, and each elementary cycle $c \in C_i$, we introduce two kinds of $\{0, 1\}$-variables x_i^c and y_i^c: if $x_i^c = 1$, then we direct edges of c so that there is a directed (s_i, t_i)-path which passes through c in a clockwise direction; if $y_i^c = 1$, then we direct edges of c so that there is a directed (s_i, t_i)-path which passes through c in an anticlockwise direction.

We are now ready to formulate MIN-MAX ORIENTATION for a cactus G.

$$\begin{aligned}
&\text{minimize } z && (1)\\
&\text{subject to } x_i^c + y_i^c = 1,\ \forall c \in C_i,\ i = 1, \ldots, q, && (2)\\
&\qquad x_i^c + y_j^c \le 1,\ \forall (i, j) \in \text{conflicting pairs on } c, \forall c \text{ in } G, && (3)\\
&\qquad d_i + \sum_{c \in C_i} (a_i^c x_i^c + b_i^c y_i^c) \le z,\ i = 1, \ldots, q, && (4)\\
&\qquad x_i^c,\ y_i^c \in \{0, 1\},\ \forall c \in C_i,\ i = 1, \ldots, q. && (5)
\end{aligned}$$

Equations (2) and (3) ensure that there are directed (s_i, t_i)-paths for all st-pairs (s_i, t_i), $1 \le i \le q$. Therefore, according to the values of x_i^c and y_i^c, we can find an orientation $\boldsymbol{G}$ of G such that $h(\boldsymbol{G}) = z$. Thus, minimizing z in Eq. (1) is equivalent to computing $h^*(G)$ for G. Since the size of the above integer programming formulation is polynomial in n, its linear relaxation problem can be solved in polynomial time.

We now propose a polynomial-time 2-approximation algorithm for cacti. Our algorithm is very simple. We first solve the linear relaxation problem, and obtain a fractional solution $\bar{x}_i^c$ and $\bar{y}_i^c$, whose objective value is $\bar{z}$. Clearly, $h^*(G) \ge \bar{z}$ since $h^*(G)$ is the optimal value for the IP above. We then obtain an integer solution x_i^c and y_i^c by rounding the values of $\bar{x}_i^c$ and $\bar{y}_i^c$, as follows: $x_i^c = 1$ if $\bar{x}_i^c \ge 0.5$, otherwise $x_i^c = 0$; $y_i^c = 1$ if $\bar{y}_i^c > 0.5$, otherwise $y_i^c = 0$. Clearly, x_i^c and y_i^c satisfy Eqs. (2), (3) and (5), and hence x_i^c and y_i^c form a feasible solution for the IP above; we can thus obtain an orientation of G. Moreover, this algorithm clearly terminates in polynomial time. Therefore, it suffices to show that the approximation ratio of this algorithm is 2. Let z_A be the objective value for the solution x_i^c and y_i^c. Since $\bar{x}_i^c \ge \frac{1}{2} x_i^c$ and $\bar{y}_i^c \ge \frac{1}{2} y_i^c$, by Eq. (4) we have

$$h^*(G) \geq \bar{z} = \max\Big\{d_i + \sum_{c \in C_i} (a_i^c \bar{x}_i^c + b_i^c \bar{y}_i^c) \mid 1 \leq i \leq q\Big\}$$
$$\geq \frac{1}{2} \max\Big\{d_i + \sum_{c \in C_i} (a_i^c x_i^c + b_i^c y_i^c) \mid 1 \leq i \leq q\Big\} = \frac{1}{2} z_A. \quad (6)$$

4.2 Pseudo-polynomial-time Algorithm

From now on, assume that the number q of st-pairs is a fixed constant. The main result of this subsection is the following theorem.

Theorem 6. MIN-MAX ORIENTATION *can be solved in time $O(nU^{2q})$ for a cactus G if q is a fixed constant, where U is an arbitrary upper bound on $h^*(G)$ and n is the number of vertices in G.*

As the upper bound U on $h^*(G)$, we will employ the approximation value z_A obtained by the 2-approximation algorithm in Section 4.1; z_A can be computed in polynomial time.

Let $G = (V, E)$ be a given cactus, let v be a node of an underlay tree T of G, and let G_v be the subgraph of G for the node v. Then, G_v and $G \setminus G_v$ share exactly one vertex u in common. Consider an optimal orientation $\boldsymbol{G}$ of G. (Remember that G has at least one orientation for the given set of st-pairs.) Then, $\boldsymbol{G}$ naturally induces the "edge-direction" $\boldsymbol{G}_v$ of G_v, which is not always an orientation for the given set of st-pairs but satisfies the following four conditions: for each st-pair (s_i, t_i), $1 \leq i \leq q$,

(a) if both s_i and t_i are in G_v, then a shortest directed (s_i, t_i)-path on $\boldsymbol{G}$ is on $\boldsymbol{G}_v$ because $\boldsymbol{G}$ is optimal and all edge-weights are non-negative;
(b) if s_i is in G_v but t_i is in $G \setminus G_v$, then there is a directed (s_i, u)-path on $\boldsymbol{G}_v$;
(c) if s_i is in $G \setminus G_v$ but t_i is in G_v, then there is a directed (u, t_i)-path on $\boldsymbol{G}_v$; and
(d) if neither s_i nor t_i are in G_v, then $\boldsymbol{G}$ has a shortest directed (s_i, t_i)-path which contains no edge of G_v.

For a q-tuple $(x_1, x_2, \ldots, x_q)$ of integers $0 \leq x_i \leq U$, $1 \leq i \leq q$, an edge-direction $\boldsymbol{G}_v$ of G_v is called an $(x_1, x_2, \ldots, x_q)$-*orientation* of G_v if the following three conditions (a)–(c) are satisfied: for each st-pair (s_i, t_i), $1 \leq i \leq q$,

(a) if both s_i and t_i are in G_v, then $\omega(\boldsymbol{G}_v, s_i, t_i) \leq x_i$;
(b) if s_i is in G_v but t_i is in $G \setminus G_v$, then $\omega(\boldsymbol{G}_v, s_i, u) \leq x_i$; and
(c) if s_i is in $G \setminus G_v$ but t_i is in G_v, then $\omega(\boldsymbol{G}_v, u, t_i) \leq x_i$.

We then define a set $F(G_v)$ of q-tuples, as follows:

$$F(G_v) = \{(x_1, x_2, \ldots, x_q) \mid G_v \text{ has an } (x_1, x_2, \ldots, x_q)\text{-orientation}\}.$$

Our algorithm computes $F(G_v)$ for each node v of T from the leaves to the root r of T by means of dynamic programming. Since $G = G_r$, we clearly have

$$h^*(G) = \min\left\{\max_{1 \leq i \leq q} x_i \mid (x_1, x_2, \ldots, x_q) \in F(G_r)\right\}. \quad (7)$$

Note that $F(G_r) \neq \emptyset$ since G has at least one orientation for the given set of *st*-pairs. Therefore, we can always compute $h^*(G)$ by Eq. (7). We omit the details of our pseudo-polynomial-time algorithm due to the page limitation.

4.3 FPTAS

We finally give the main result of this section, as in the following theorem.

Theorem 7. Min-max orientation *admits a fully polynomial-time approximation scheme for cacti if q is a fixed constant.*

As a proof of Theorem 7, we give an algorithm to find an orientation $\boldsymbol{G}$ of a cactus G with $h(\boldsymbol{G}) < (1+\varepsilon)h^*(G)$ in time polynomial in both n and $1/\varepsilon$ for any real number $\varepsilon > 0$, where n is the number of vertices in G. Thus, our approximation value $h_A(G)$ for G is $h(\boldsymbol{G})$, and hence the error is bounded by $\varepsilon h^*(G)$, that is,

$$h_A(G) - h^*(G) = h(\boldsymbol{G}) - h^*(G) < \varepsilon h^*(G). \tag{8}$$

We now outline our algorithm and its analysis. We extend the ordinary "scaling and rounding" technique [9], and apply it to min-max orientation for a cactus $G = (V, E)$. For some scaling factor $\tau > 0$, let G_τ be the graph with the same vertex set V and edge set E as G, but the weight of each edge $e \in E$ is defined as follows: $\bar{\omega}(e) = \lceil \omega(e)/\tau \rceil$. We optimally solve min-max orientation for G_τ by using the pseudo-polynomial-time algorithm in Section 4.2. We take the optimal orientation $\boldsymbol{G}_\tau$ for G_τ as our approximation solution for G. Then, we can show that $h_A(G) - h^*(G) < \tau|E|$. Intuitively, this inequality holds because the error occurs at most τ at each edge in G_τ. By Eq. (6) and taking $\tau = \varepsilon z_A/2|E|$, we have Eq. (8). Since $h^*(G_\tau) \leq |E| + \frac{z_A}{\tau} = (1 + \frac{2}{\varepsilon})|E|$, by Theorem 6 we can find the optimal orientation $\boldsymbol{G}_\tau$ for G_τ in time $O\Big(n\big(|E| + \frac{2|E|}{\varepsilon}\big)^{2q}\Big) = O\Big(\frac{n^{2q+1}}{\varepsilon^{2q}}\Big)$; since G is a cactus, $|E| = O(n)$.

References

1. Brandstädt, A., Le, V.B., Spinrad, J.P.: Graph Classes: A Survey. Society for Industrial and Applied Mathematics, Philadelphia (1999)
2. Chvátal, V., Thomassen, C.: Distances in orientations of graphs. J. Combinatorial Theory, Series B 24, 61–75 (1978)
3. Garey, M.R., Johnson, D.S.: Computers and Intractability: A Guide to the Theory of NP-Completeness. Freeman, San Francisco (1979)
4. Hakimi, S.L., Schmeichel, E.F., Young, N.E.: Orienting graphs to optimize reachability. Information Processing Letters 63, 229–235 (1997)
5. Lee, C.-Y., Lei, L., Pinedo, M.: Current trends in deterministic scheduling. Annals of Operations Research 70, 1–41 (1997)
6. Papadimitriou, C.H.: Computational Complexity. Addison-Wesley, Reading (1994)
7. Robbins, H.E.: A theorem on graphs with an application to a problem of traffic control. American Mathematical Monthly 46, 281–283 (1939)
8. Uehara, R., Uno, Y.: On computing longest paths in small graph classes. International Journal of Foundations of Computer Science 18, 911–930 (2007)
9. Vazirani, V.V.: Approximation Algorithms. Springer, Heidelberg (2001)