Series-parallel orientations preserving the cycle-radius

Nili Guttmann-Beck* Refael Hassin†

June 22, 2011

Abstract

Let G be an undirected 2-edge connected graph with nonnegative edge weights and a distinguished vertex z. For every node consider a shortest cycle containing this node and z in G. The cycle-radius of G is the maximum length of a cycle in this set. Let H be a directed graph obtained by directing the edges of G. The cycle-radius of H is similarly defined except that cycles are replaced by directed closed walks. We prove that there exists for every nonnegative edge weight function an orientation H of G whose cycle-radius equals that of G if and only if G is series-parallel.

1 Introduction

Orienting an edge weighted undirected graph (or a multigraph) $G = (V, E)$ means replacing each of its edges $e = \{u, v\}$ by a directed arc (u, v) or (v, u) (in this paper edges are undirected and arcs are directed). We denote both the orientation and the resulting directed graph by H.

Let $G = (V, E)$ be a 2-edge connected undirected (multi) graph, with non-negative edge weights and a distinguished node $z \in V$. For every node $v \in V \setminus \{z\}$ find a shortest (with respect to its total edge weight) *undirected cycle* containing this node and z, and define the cycle-radius of G with respect to z, $\mathrm{CR}_z(G)$, as the maximum length of these cycles. Similarly, for every node $v \in V \setminus \{z\}$ find a shortest *closed walk* containing this node and z in H, and define the cycle-radius of the orientation H with respect to z, $\mathrm{CR}_z(H)$, as the maximal length of these walks. To simplify the exposition we often refer in the rest of the paper to closed walks simply as cycles. We refer to cycles with no repeated nodes as simple cycles.

This paper focuses on series-parallel graphs, i.e., graphs that do not contain a K_4 subdivision [6]. These are also the graphs whose tree-width is at most 2. Other characterizations of series-parallel graphs are also available, for example, as in our paper, the characterization by Bein, Brucker and Tamir [3] requires that a certain property holds for arbitrary selection of edge weights: An acyclic multigraph with a single source and a single sink is series-parallel if and only if for arbitrary linear cost functions and arbitrary capacities the corresponding minimum cost flow problem can be solved by a greedy algorithm.

Our main result is the following Theorem:

*Department of Computer Science, The Academic College of Tel-Aviv Yaffo, Yaffo, Israel. Email: becknili@mta.ac.il

†Department of Statistics and Operations Research, Tel-Aviv University, Tel-Aviv 69978, Israel. Email: hassin@post.tau.ac.il

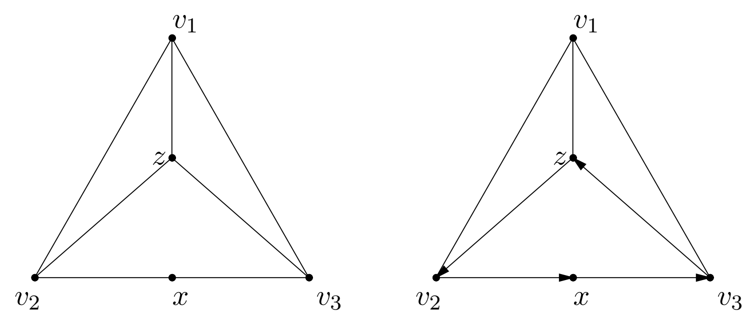

Figure 1: A proper subdivision of K_4.

Theorem 1.1 *Suppose that $G = (V, E)$ is a 2-edge connected undirected graph with $|V| > 4$. G is series-parallel if and only if for arbitrary nonnegative edge weights and every $z \in V$ there exists an orientation H satisfying* $\mathrm{CR}_z(H) = \mathrm{CR}_z(G)$.

A *proper subdivision* of K_4 is any graph derived from a complete graph K_4 by a sequence of at least one edge subdivision - see Figure 1-left. Note that an undirected graph with more than four nodes is series-parallel if and only if it does not contain a proper subdivision of K_4. We first show that in this case there is a weight function w for which no such orientation exists. We give all subdividing paths total length one except for one path which contains an inner node, denoted as x, to which we assign total length of two. Other edges are of length three. If the cycle containing z and x is oriented (see Figure 1-right), then it is not possible to complete the orientation so that both the cycle containing z,v_1 and v_2, and the cycle containing z,v_1 and v_3, are oriented into directed cycles. The other direction of the proof follows from Theorem 2.1 which we state and prove in Section 2.

For every pair of nodes in G, mark the shortest (not necessarily simple) cycle that contains these points and define the *cycle-diameter*, $\mathrm{CD}(G)$, of a graph as the length of the longest, over all node pairs, such undirected cycle. Similarly, for every pair of nodes in H, mark the shortest *closed walk* containing these nodes, and define the cycle-diameter $\mathrm{CD}(H)$ of the orientation H as the longest of these walks. We note that a walk may contain repetitions of arcs. As an illustration to these definitions, consider the graph G in Figure 2 (left), where all edges have unit weights. In this example, for every pair of nodes there is a common cycle with four edges, and $\mathrm{CD}(G) = 4$. For the orientation H shown in Figure 2 (right) $\mathrm{CD}(H) = 8$ is determined by $(u, x, w, y, v, x, w, y, u)$, which is the shortest directed closed walk containing v and u.

Clearly, every orientation H satisfies that $\mathrm{CD}(H) \leq 2\,\mathrm{CR}_z(H)$ for every z, because we can build a closed walk between u and v by combining the closed walk connecting v and z with the closed walk connecting u and z. Also, by definition, $\mathrm{CR}_z(G) \leq \mathrm{CD}(G)$ for every $z \in V$.

The cycle-diameter of a graph is our new measure for closeness of the nodes of the graph to each other. The directed cycle-diameter can serve as a measure for the quality of an orientation.

An interesting question is to bound $\max_{G\in\mathcal{G}}\{\rho(G)\}$ for various classes $\mathcal{G}$ of graphs, where $\rho(G) = \min\left\{\frac{\mathrm{CD}(H)}{\mathrm{CD}(G)} \;:\; H \text{ is an orientation of } G\right\}$. A by-product of such a bound is that an algorithm that computes an orientation with minimum, or approximate, cycle-diameter can also be used to compute an orientation with a small diameter. We now explain this point.

The diameter of a graph (digraph) is the maximum over all (ordered) node pairs u, v, of the

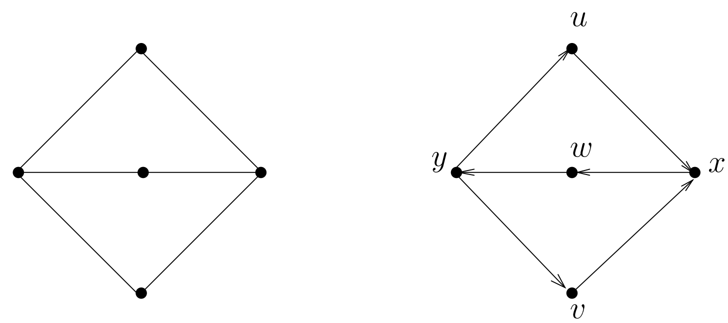

Figure 2: G and H.

length of the shortest u-v ($u \rightarrow v$) path. Denote by $\mathrm{D}(G)$ and $\mathrm{D}(H)$ the diameters of G and H, respectively.

Denote by $\mathrm{D}^*(G)$ the smallest possible diameter of an orientation of G. The orientation which gives $\mathrm{D}^*(G)$ contains a $u \rightarrow v$ path and a $v \rightarrow u$ path for every $u, v \in V$. Ignoring directions, the union of the shortest such paths contains a cycle in G that contains both u and v. Thus, the length of this cycle is at most twice the diameter of the oriented graph, giving that $\mathrm{CD}(G) \leq 2\mathrm{D}^*(G)$. Suppose that, for every $G \in \mathcal{G}$, we know how to compute an orientation $H = H(G)$ such that $\frac{\mathrm{CD}(H)}{\mathrm{CD}(G)} \leq \rho$. Then $\mathrm{D}(H)$ satisfies $\mathrm{D}(H) \leq \mathrm{CD}(H) \leq \rho\,\mathrm{CD}(G) \leq 2\rho\,\mathrm{D}^*(G)$.

In this paper we prove that $\rho = 2$ for series-parallel graphs (see Theorem 3.2 below). This bound is tight as can be seen from the series-parallel graph shown in Figure 2.

In [11] we present a proof (associated with a polynomial time algorithm) that $\rho \leq 810$ for planar graphs. The bound can clearly be reduced, and we conjecture that the real value is quite small. However, our proof is very tedious and too long for a journal paper.

There are two common approaches in the literature to define a good orientation. One approach requires that the orientation maintains high connectivity (see [10, 13, 16, 17]). The other approach requires short directed diameter. Chvátal and Thomassen [5] show that there exists an orientation such that $\mathrm{D}(H) \leq 2\mathrm{D}(G)^2 + 2\mathrm{D}(G)$. On the other hand, there are graphs where for every orientation $\mathrm{D}(H) \geq \frac{1}{4}\mathrm{D}(G)^2 + \mathrm{D}(G)$. In addition, they show that finding a minimum diameter orientation of an undirected graph G is NP-hard, and even deciding whether a graph admits an orientation H with $\mathrm{D}(H) \leq 2$ is NP-complete. Koh and Tay [14] present a survey on the minimum directed diameter in restricted classes of graphs (see also [2, 9, 8]). Burkard et al [4] consider minimizing the sum of shortest path lengths between node pairs in the orientation. Medvedovsky, Bafna, Zwick and Sharan [15] (also in [1, 12]) maximize the number of pairs from a given set that admit a directed path in the oriented graph. Eggemann and Noble [7] describe an algorithm that decides if a planar graph G has an orientation with diameter at most l using tree decomposition of planar graphs. Yen [18] investigates a version of the problem where each node x has a cost $C(x)$, for every node we add the sum of length of the edges leaving this node, and we which to minimize this size for all the nodes of the graph.

A naive approach for orienting a series-parallel graph and obtain a small cycle-diameter follows the construction of the graph from a cycle. Direct the initial cycle in an arbitrary orientation, then when an edge is replaced by two edges in series direct them in the same orientation as the edge they replace, and when an edge is added in parallel direct it also in the same orientation.

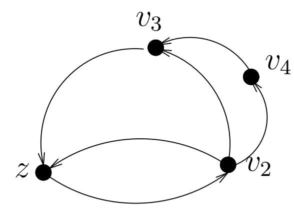

Figure 3: Naive orientation.

This approach may fail even for the simple example of a graph consisting of three parallel edges (with one end-node denoted as z), two of them of zero weight and one having unit weight. If the initial cycle contains the unit length edge, then the two zero length edges may end up with the same orientation and $\mathrm{CR}_z(H) = 1$, whereas the optimal orientation has $\mathrm{CR}_z(H) = 0$. A possible conclusion of this example might be that it is worth starting with a short cycle. The example in Figure 3 shows that even this approach can fail. Here the weight of the edge $\{z, v_3\}$ is one and the other edges have zero weight. We start by orienting the parallel edges $\{z, v_2\}$ in opposite direction. Then the other edges are oriented according to (z, v_1) implying again $\mathrm{CR}_z(H) = 1$, whereas the optimal orientation has $\mathrm{CR}_z(H) = 0$.

The next section completes the proof of Theorem 1.1 by constructing (in polynomial time) an orientation with the same cycle-radius.

2 Cycle-radius preserving orientations

Let $G = (V, E)$ be a 2-edge connected undirected series-parallel graph, with non-negative edge weights and a distinguished node $z \in V$. For every node $v \in V \setminus \{z\}$ let $C_z(v)$ be a shortest undirected cycle containing v and z. Denote $\mathcal{C}_z = \{C_z(v) \ : \ v \in V\}$, and let $G_{\mathcal{C}_z}$ be the graph induced by the edges in $\mathcal{C}_z$. *Without loss of generality we assume that all the cycles in $\mathcal{C}_z$ are of different lengths.*

In this section we constructively prove the following theorem:

Theorem 2.1 *If $G_{\mathcal{C}_z}$ is series-parallel then there is an orientation H such that every cycle $C \in \mathcal{C}_z$ is a directed cycle in H.*

We note that if G is series-parallel then $G_{\mathcal{C}_z}$ is also series-parallel, hence Theorem 1.1 is a corollary of Theorem 2.1.

Definition 2.2 Every cycle $C \in \mathcal{C}_z$ consists of a sequence of simple cycles, say $C_{v_1}, \ldots, C_{v_k}$, where $z \in C_{v_1}$, $C_{v_i} \cap C_{v_{i+1}} = \{v_i\}$ is a cut-node of C, for $i = 1, \ldots, k-1$, and the cycles are otherwise vertex-disjoint. Our algorithm orients these cycles by increasing index. It works recursively on subgraphs with new distinguished nodes called *anchors* which are cut nodes of cycles in $\mathcal{C}_z$. At any stage of the algorithm the anchor which applies to the cycle is the unique cut node v_i such that $C_{v_1}, \ldots, C_{v_{i-1}}$ have already been oriented, C_i is either undirected or partially directed, and $C_{i+1}, \ldots, C_k$ are still undirected. Initially the anchor is z and the subgraph is $G_{\mathcal{C}_z}$.

We denote by $V(G')$ and $E(G')$ the node and edge sets of a subgraph G', respectively, and $P \setminus Q$ as the subgraph induced by $E(P) \setminus E(Q)$, for subgraphs P and Q.

Definition 2.3 Let $v_1, v_2, \ldots, v_l$ be the neighbors of z in $G_{\mathcal{C}_z}$. For every $i, j \in \{1, \ldots, l\}$, $i \neq j$ define $A_{\{i,j\}} = \{C \in \mathcal{C}_z : C \text{ contains the path } (v_i, z, v_j)\}$, and let $C_{\{i,j\}}$ be a shortest cycle in $A_{\{i,j\}}$. Define $F(z)$, the *flower* of z, as $F(z) = \{C_{\{i,j\}} | A_{\{i,j\}} \neq \emptyset\}$.

The algorithm first orients the cycles in $F(z)$, next all the simple cycles in $\mathcal{C}_z$, and then it finds new anchors with their relevant subgraphs, and recursively orients these subgraphs.

Definition 2.4 Given $F(z)$, define $T_f(z)$ as the graph whose node set consists of the neighbors $\{v_1, v_2, \ldots, v_l\}$ of z in $G_{\mathcal{C}_z}$, and contains an edge between v_i and v_j if $A_{\{i,j\}} \neq \emptyset$. Consider for example the flower $F(z)$ in Figure 4-left and $T_f(z)$ in Figure 4-right.

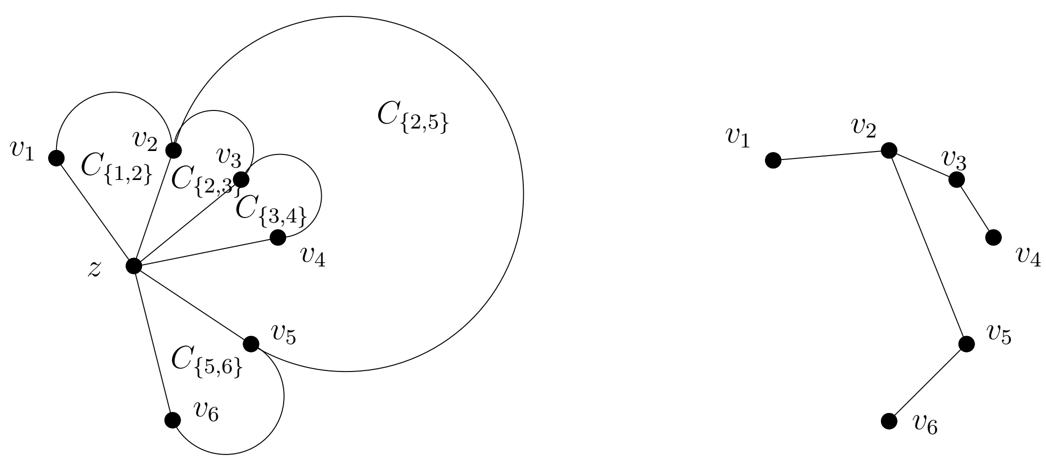

Figure 4: Example for Definition 2.4

Lemma 2.5 *$T_f(z)$ is a forest.*

Proof: If $T_f(z)$ contains a cycle (as in Figure 5-right), then this cycle corresponds to cycles in $G_{\mathcal{C}_z}$ which intersect each other, (see Figure 5 -left). Thus, a K_4 subdivision has been created, contradicting the assumption that $G_{\mathcal{C}_z}$ is series-parallel. ■

Remark 2.6 The forest $T_f(z)$ may contain more than one component. Consider for example the graph in Figure 6-left. The corresponding $T_f(z)$ is shown in Figure 6-right, a forest containing two components.

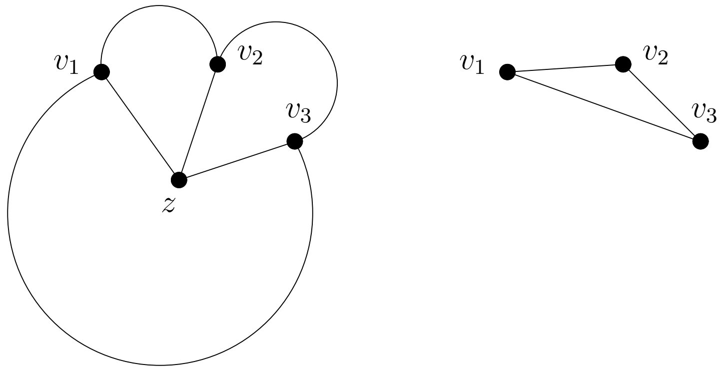

Figure 5: Example for Lemma 2.5

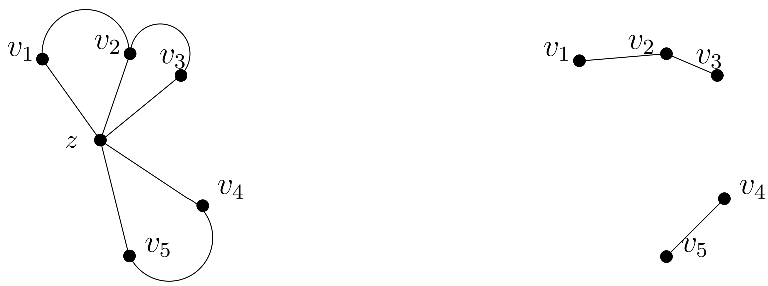

Figure 6: Example for Remark 2.6

Orient_flower

input

$F(z)$ - the flower of z.

At most one cycle in every connected component of $T_f(z)$ is partially oriented.

returns

An orientation of $F(z)$.

begin

while *(there exists a cycle in $F(z)$ which is not completely oriented)*

  while *(there exists a cycle $C \in F(z)$ which is partially oriented)*

    Complete the orientation of C.

  end while

  Arbitrarily choose a cycle $C \in F(z)$ and orient it in an arbitrary direction.

end while

end *Orient_flower*

Figure 7: Orienting a flower

Algorithm Orient_flower (see Figure 7) orients the cycles in $F(z)$. By Lemma 2.5 no contradiction is created. For example applying Algorithm Orient_flower on the flower in Figure 4-left yields the directed graph in Figure 8.

At the beginning all the cycles are completely undirected, so one of the cycles is chosen, for example $C_{\{2,3\}}$. This cycle is oriented $(z \to v_3 \to v_2 \to z)$. Now the cycles $C_{\{1,2\}}$,$C_{\{3,4\}}$ and $C_{\{2,5\}}$ are partially oriented, so one of them is chosen for example $C_{\{3,4\}}$. Since the edge $(z \to v_3)$ is already directed, the orientation of the cycle is $(z \to v_3 \to v_4 \to z)$. The cycles $C_{\{1,2\}}$ and $C_{\{2,5\}}$ are partially oriented, so one of them is chosen for example $C_{\{1,2\}}$. It is oriented $(z \to v_1 \to v_2 \to z)$ to agree with the edge $(v_2 \to z)$. The only cycle which is partially oriented is $C_{\{2,5\}}$, so it is chosen and oriented $(z \to v_5 \to v_2 \to z)$ to agree with the edge $(v_2 \to z)$. Finally $C_{\{5,6\}}$ is partially oriented and it is oriented $(z \to v_5 \to v_6 \to z)$ to agree with the edge $(z \to v_5)$.

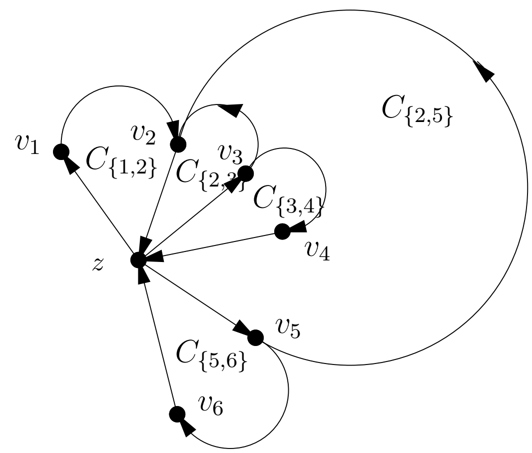

Figure 8: Example for Algorithm Orient_flower

Later in Lemma 2.14 we will show that in every connected component of $T_f(z)$ at most one cycle is partially oriented. Algorithm Orient_by_anchor (see Figure 9) calls Algorithm Orient_flower. It orients all the simple cycles in $G_{\mathcal{C}_z}$ and recursively orients all the other cycles in $G_{\mathcal{C}_z}$. Algorithm Orient_by_anchor is demonstrated in Figure 10. The graph $G_{\mathcal{C}_z}$ is shown on the left side of the figure, the anchor of this graph is the node z. The flower $F(z)$ contains three cycles: $(z, v_5, v_4, v_3, v_2, v_1, z)$, (z, v_5, v_{16}, z) and (z, v_{16}, v_{17}, z). In this example none of the flower's cycles is partially oriented. When Orient_flower is activated the algorithm may choose to orient it $(z \to v_5 \to v_4 \to v_3 \to v_2 \to v_1 \to z)$. Next, the orientation of the cycle $(z \to v_5 \to v_{16} \to z)$ is completed, and last, the orientation of the cycle $z \to v_{17} \to v_{16} \to z$ is completed.

Now Algorithm Orient_by_anchor may:

1. Choose the cycle $(z, v_5, v_4, v_3, v_2, v_7, v_6, v_1, z)$ and direct the path $(v_2 \to v_7 \to v_6 \to v_1)$ to complete its orientation.

2. Choose the cycle $(z, v_5, v_4, v_{12}, v_{11}, v_9, v_3, v_2, v_1, z)$ and direct the path $(v_4 \to v_{12} \to v_{11} \to v_9 \to v_3$ to complete its orientation.

3. Choose the cycle $(z, v_5, v_4, v_3, v_2, v_7, v_8, v_6, v_1, z)$ and direct the path $(v_7 \to v_8 \to v_6)$ to complete its orientation.

Orient_by_anchor

input

A distinguished anchor node z.

A set of cycles $\mathcal{C}_z$ *(connecting every node in* $G_{\mathcal{C}_z}$ *to* z*).*

$G_{\mathcal{C}_z}$ *- an undirected (or mixed) series-parallel graph.*

returns

An orientation of $G_{\mathcal{C}_z}$.

begin

Orient_flower $F(z)$.

for every *(simple cycle* $C \in (\mathcal{C}_z \setminus F(z))$*)*

Complete the orientation of C.

end for

$SA := \emptyset$. *[This will be the set of new anchor nodes]*

for every *(cycle* $C \in \mathcal{C}_z$ *which is still not fully oriented)*

Let $v^*(C)$ *be the anchor node of* C *(see Definition 2.2).*

$SA := SA \cup \{v^*(C)\}$.

end for

for every $(v^* \in SA)$

Let $SC_{v^*} := \{C_{v^*}(u) | v^* \text{ is the anchor node of } C_z(u)\}$.

$[C_{v^*}(u)$ *is the shortest cycle containing* u *and* v^*,

this cycle may contain few simple cycles].

Let G_{v^*} *be the graph induced by the union of the cycles in* SC_{v^*}.

Orient_by_anchor (v^*, SC_{v^*}, G_{v^*}).

end for

end *Orient_by_anchor*

Figure 9: The orientation algorithm

4. Choose the cycle $(z, v_5, v_4, v_{12}, v_{11}, v_{10}, v_9, v_3, v_2, v_1, z)$ and direct the path $(v_{11} \rightarrow v_{10} \rightarrow v_9)$ to complete its orientation.

We note that the cycles $(v_4, v_{12}, v_{13}, v_4)$,$(v_4, v_{14}, v_{13}, v_4)$ and $(v_4, v_{15}, v_{14}, v_4)$ create a flower with the anchor-node v_4. The algorithm will call Orient_by_anchor on this subgraph (which contains in this case just the flower). The flower contains one connected component with exactly one cycle $(v_4, v_{12}, v_{13}, v_4)$ which is partially oriented.

We know by Lemma 2.5 that no contradiction is created while orienting $F(z)$. The next lemmas prove that no contradiction is created for any simple cycle in $G_{\mathcal{C}_z}$.

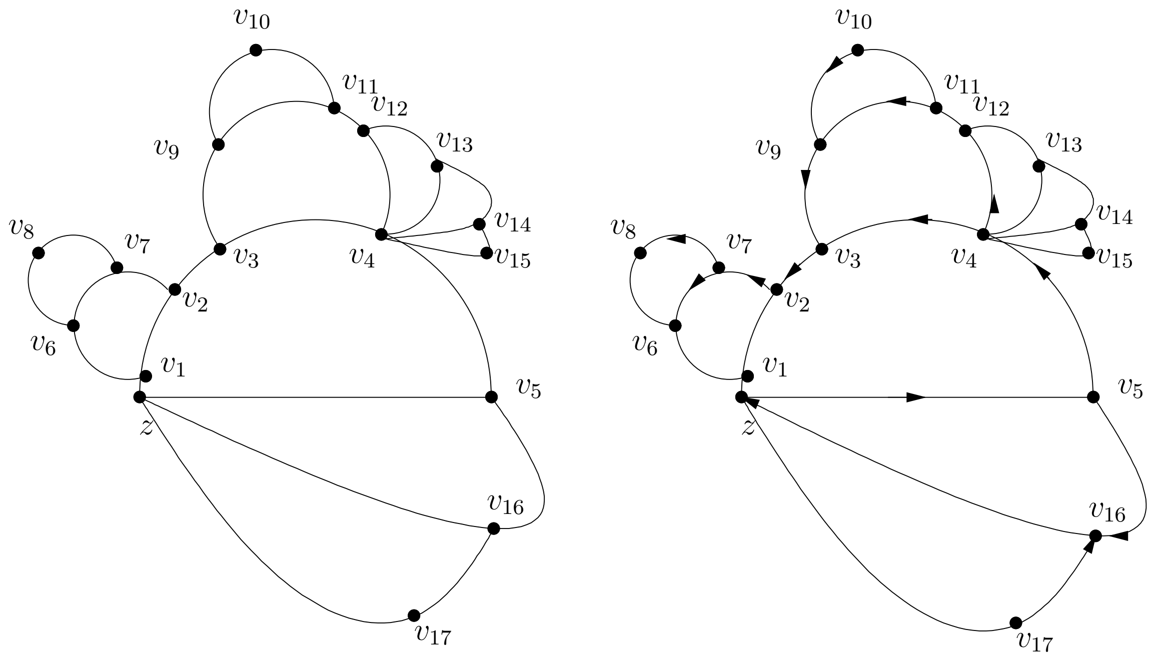

Figure 10: Example for Algorithm Orient_By_anchor

Lemma 2.7 *For a cycle $C \in A_{\{i,j\}} \setminus \{C_{\{i,j\}}\}$:*

- $C_{\{i,j\}} \setminus C$ *is a (not necessarily simple) path (or a cycle) with at most two nodes in* $C_{\{i,j\}} \cap C$.
- $C \setminus C_{\{i,j\}}$ *is a (not necessarily simple) path (or a cycle) with at most two nodes in* $C_{\{i,j\}} \cap C$.

Proof: Let v be a node satisfying $C = C_z(v)$ (C is the shortest cycle containing v and z). We prove the claim by contradiction. Suppose that $C_{\{i,j\}} \setminus C$ is not a path (in this case it contains at least two disjoint paths). Let (P_1, T, P_2) be a subpath of C such that $P_1, P_2 \subseteq C \setminus C_{\{i,j\}}$, $T \subseteq C_{\{i,j\}} \cap C$ and $v \in P_1$, $z \in C_{\{i,j\}} \setminus T$, (see Figure 11).

Let Q_1 (Q_2) be the subpath of $C_{\{i,j\}} \setminus \{z\}$ between the end-points of P_1 (P_2). Without loss of generality we can assume that $C_{\{i,j\}}$ was defined as the shortest cycle containing z and a node in Q_1. Since $C = C_z(v)$, $l(P_2) < l(Q_2)$. But then the cycle $C_{\{i,j\}} \setminus Q_2 \cup P_2$ is a cycle containing z and the nodes in Q_1, which is shorter then $C_{\{i,j\}}$, contradicting the definition of $C_{\{i,j\}}$.

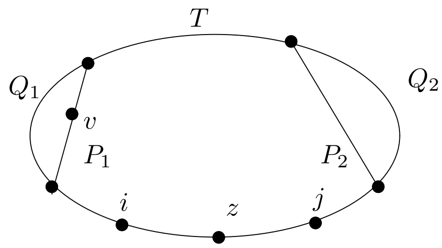

Figure 11: Example for Lemma 2.7, P_1 and P_2 in $C \setminus C_{\{i,j\}}$, Q_1 and Q_2 in $C_{\{i,j\}} \setminus C$

Since $C_{\{i,j\}} \setminus C$ and $C_{\{i,j\}} \cap C$ are both paths, it follows that $C \setminus C_{\{i,j\}}$ is also a path with the same ends as $C_{\{i,j\}} \setminus C$. ∎

Definition 2.8 Given two (not necessarily distinct) nodes $u, v \in V(C_{\{i,j\}})$, define $S_{u,v} = \{C \in A_{\{i,j\}} : C \setminus C_{\{i,j\}}$ is a u-v path (or a u-u cycle if $u = v$)$\}$ (for example in Figure 10 $C_z(v_{10}) \in S_{v_3,v_4}$).

Observation 2.9 *$\{S_{u,v} : u, v \in C_{\{i,j\}}\}$ is a partition of $A_{\{i,j\}} \setminus C_{\{i,j\}}$.*

Lemma 2.10 *If $C' \in A_{\{i,j\}}$ and $C'' \in A_{\{a,b\}}$ for $\{i,j\} \neq \{a,b\}$ then $E(C' \setminus C_{\{i,j\}}) \cap E(C'' \setminus C_{\{a,b\}}) = \emptyset$.*

Proof: Suppose $e \in E(C' \setminus C_{\{i,j\}}) \cap E(C'' \setminus C_{\{a,b\}})$ as in Figure 12-right. Then a K_4 subdivision is created with nodes $\{x, y, i, j\}$. In Figure 12-left we present the special case $|\{i,j\} \cap \{a,b\}| = 1$. ∎

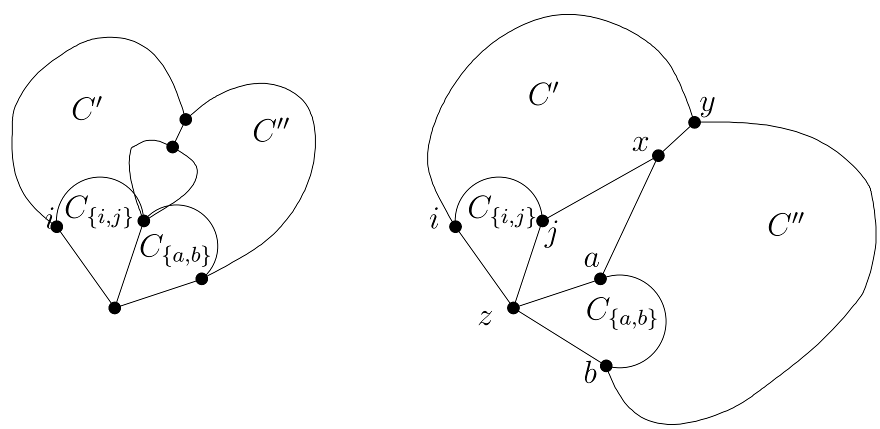

Figure 12: Examples for Lemma 2.10

The first loop of Algorithm Orient_by_anchor orients all the simple cycles is $\mathcal{C}_z$. Lemma 2.13 proves that no contradiction is created when orienting these cycles. We first need the next two lemmas.

Lemma 2.11 *Suppose $C', C'' \in A_{\{i,j\}}$, with $V(C_{\{i,j\}} \setminus C') \cap V(C_{\{i,j\}} \setminus C'') = \emptyset$, then $E(C' \setminus C_{\{i,j\}}) \cap E(C'' \setminus C_{\{i,j\}}) = \emptyset$.*

Proof: By Lemma 2.7 $(C_{\{i,j\}} \setminus C')$, $(C_{\{i,j\}} \setminus C'')$, $(C' \setminus C_{\{i,j\}})$, $(C'' \setminus C_{\{i,j\}})$ are all simple paths. Suppose that $C' \in S_{v_1,v_2}$ and $C'' \in S_{v_3,v_4}$ (it might be that $v_1 = v_2$ or $v_3 = v_4$). See Figure 13 for an example of the graph. Suppose that $E(C' \setminus C_{\{i,j\}}) \cap E(C'' \setminus C_{\{i,j\}})$ contains an edge $\{v_5, v_6\}$. Then, a subdivision of K_4 is created between v_5, v_6, v_2, v_3, contradicting the assumption that this is a series-parallel graph. ■

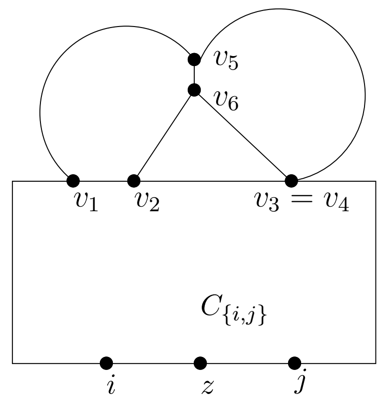

Figure 13: Example for Lemma 2.11

Lemma 2.12 *Let $C', C'' \subset A_{\{i,j\}}$, both simple cycles, $C' \in S_{v_1,v_2}$ with $v_1 \neq v_2$ and $C'' \in S_{v_3,v_4}$ with $v_3 \neq v_4$. Then either $\{v_1, v_2\} = \{v_3, v_4\}$ or $E(C' \setminus C_{\{i,j\}}) \cap E(C'' \setminus C_{\{i,j\}}) = \emptyset$.*

Proof: We denote by $(v_1 - v_2)$ and $(v_3 - v_4)$ the paths on $C_{\{i,j\}}$ that do not contain z, between v_1 and v_2 and between v_3 and v_4, respectively.

Suppose that $E(C' \setminus C_{\{i,j\}}) \cap E(C'' \setminus C_{\{i,j\}}) \neq \emptyset$ (for example they share an edge which touches $C_{\{i,j\}}$). Since $\{v_1, v_2\} \neq \{v_3, v_4\}$ it follows that $(v_1 - v_2) \neq (v_3 - v_4)$. Therefore these paths satisfy one of the following relations:

- $(v_1 - v_2) \cap (v_3 - v_4) = \emptyset$, in this case $E(C' \setminus C_{\{i,j\}}) \cap E(C'' \setminus C_{\{i,j\}}) = \emptyset$ according to Lemma 2.11.
- $(v_1 - v_2) \cap (v_3 - v_4)$ contains exactly one node, as in Figure 14-left.
- $(v_1 - v_2) \cap (v_3 - v_4) = P$, $P \neq C_{\{i,j\}} \setminus C', C_{\{i,j\}} \setminus C''$ as in Figure 14-middle (where $C'' \setminus C_{\{i,j\}}$ is given by the broken line).
- $(v_1 - v_2) \subseteq (v_3 - v_4)$ or $(v_3 - v_4) \subseteq (v_1 - v_2)$, as in Figure 14-right.

In the latter three cases the graph contains a subdivision of K_4. ■

Lemma 2.13 *Algorithm Orient_by_anchor changes all the simple cycles in $\mathcal{C}_z$ into directed cycles, without creating any contradiction.*

Proof: We first note that since $T_f(z)$ is a forest (by Lemma 2.5) no contradiction may occur when Algorithm Orient_flower orients cycles in $F(z)$.

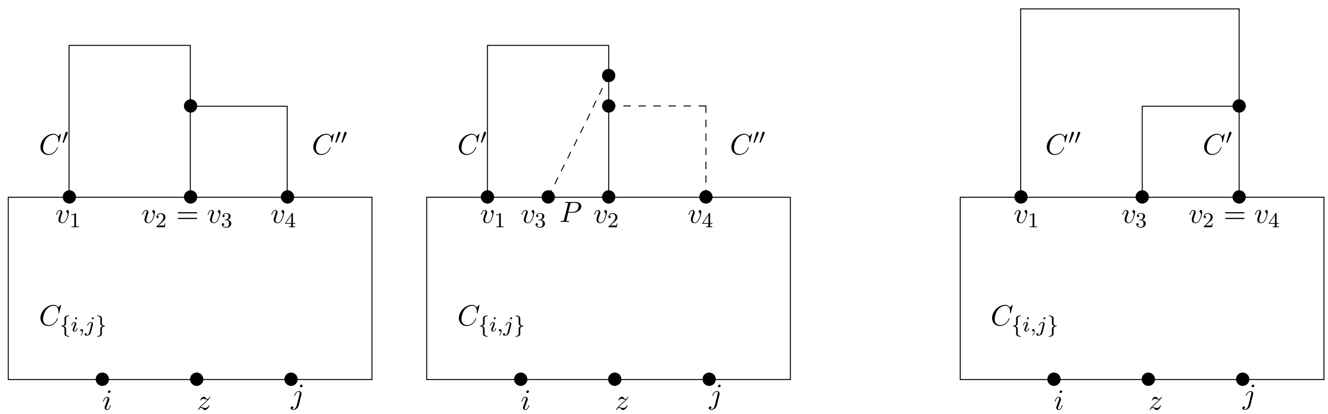

Figure 14: Examples for Lemma 2.12

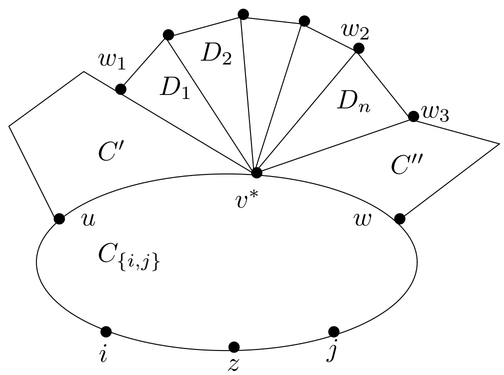

Figure 15: Example for Lemma 2.14

Let C' and C'' be distinct simple cycles in $\mathcal{C}_z \setminus F(z)$. There are $i, j, a, b \in \{1, \ldots, l\}$ such that $C' \in A_{\{i,j\}} \setminus C_{\{i,j\}}$ and $C' \in A_{\{a,b\}} \setminus C_{\{a,b\}}$.

If $\{i, j\} \neq \{a, b\}$ then by Lemma 2.10 $E(C') \cap E(C'') = \emptyset$ and their orientation is independent.

If $\{i, j\} = \{a, b\}$ then $C', C'' \in A_{\{i,j\}} \setminus C_{\{i,j\}}$. If $C_{\{i,j\}} \setminus C' = C_{\{i,j\}} \setminus C'' = (u - v)$ (a simple path between u and v) and without loss of generality assume that in $C_{\{i,j\}}$ the path is oriented $u \to v$, then in C' and in C'' the (u, v) path is directed from u to v, independent of the order by which these paths are oriented. If $C_{\{i,j\}} \setminus C' \neq C_{\{i,j\}} \setminus C''$ then by Lemma 2.12 $E(C_{\{i,j\}} \setminus C') \cap E(C_{\{i,j\}} \setminus C'') = \emptyset$ so orienting one of these paths doesn't affect the other, and again the order of their orientation can be chosen arbitrary. ■

In the first loop of Algorithm Orient_by_anchor all the simple cycles in $G_{\mathcal{C}_z}$ are oriented. After this stage for every anchor-node v^*, in each component $T_f(v^*)$ at most one of its cycles at one of the components is partially oriented (for example in Figure 15 either D_1 or D_n is partially oriented). This ensures that no contradiction will happen when orienting the next flowers. In particular, Lemma 2.14 below shows that the situation shown in Figure 15, where both C' and C'' imply a partial orientation on the graph for the anchor-node v, is not possible.

Lemma 2.14 *After the first loop of Algorithm Orient_by_anchor (in which the simple cycles in $G_{\mathcal{C}_z}$ are oriented) for every anchor-node v^*, each component of $T_f(v^*)$ contains at most one node* *associated with a partially oriented cycle, and if such a node exists then the oriented part of the cycle is a path.*

Proof: After the first loop of Algorithm Orient_by_anchor, a flower $F(z)$ and all the simple cycles in $\mathcal{C}_z \setminus F(z)$ are oriented, for example in Figure 15 the cycles $C_{\{i,j\}}$,C' and C'' are oriented. Next, v^* is identified as an anchor node. and the set SC_{v^*} contains the cycles $D_1, D_2, \ldots, D_n$.

We prove the claim by contradiction, suppose that the two oriented subpaths belongs to two partially oriented cycles in the same component of $T_f(v)$ as D_1 and D_n in Figure 15. In this case the graph contains a subdivision of K_4 (created by the nodes w_1, w_2, w_3 and v^*). ∎

We conclude from Lemma 2.14 that every flower on which the algorithm is activated contains at most one directed path in every component of $T_f(z)$, and can be further directed by Algorithm Orient_flower.

The next corollary follows from Lemmas 2.13 and 2.14:

Corollary 2.15 *Algorithm Orient_by_anchor changes in polynomial time every cycle $C \in \mathcal{C}_z$ into a directed cycle (thus proving Theorems 1.1 and 2.1).*

3 Concluding remarks

Definition 3.1 For $u, v \in V$, $\mathrm{ucl}(u, v)$ is the length of the shortest cycle in G containing u and v. Similarly, for an orientation H of G, $\mathrm{dcl}(u, v)$ is the length of the shortest closed walk in H containing these nodes.

Theorem 3.2 *Let H be an orientation satisfying Theorem 1.1, then* $\mathrm{CD}(H) \leq 2\mathrm{CD}(G)$, *implying* $\rho(G) = 2$ *for series-parallel graphs.*

Proof: According to Theorem 1.1, the length of the shortest oriented walk which contains any two nodes $v, u \in V \setminus \{z\}$ in H is at most $\mathrm{ucl}(v, z) + \mathrm{ucl}(u, z) \leq 2\mathrm{CD}(G)$. Thus, $\mathrm{CD}(H) \leq 2\mathrm{CD}(G)$. ∎

The bound for ρ is tight. Consider again the undirected graph in Figure 2(left), and suppose that each edge length is one. Without loss of generality assume the orientation in Figure 2(right). In this case $\mathrm{ucl}(v, u) = 4$ but $\mathrm{dcl}(v, u) = 8$.

The main problem left open by this paper is whether ρ is bounded for general graphs. Another open problem concerns the existence of solutions with even stronger properties, similar to those proved to exist by Nash-Williams with respect to connectivity. Define $r = \max_{u,v \in V} \frac{\mathrm{dcl}(v,u)}{\mathrm{ucl}(v,u)}$. For a given family of graphs, prove a bound, if one exists, such that for any graph in this family there exists an orientation with a smaller value of r. The next theorem shows that for some series-parallel graphs no orientation can achieve $r \leq 2$.

Theorem 3.3 *There are series-parallel graphs such that $r > 2$ in every orientation of the graph.*

Proof: Consider the undirected graph in Figure 16 (top). Suppose that each curved edge (even if subdivided) is of total length one and each straight edge is of length zero.

In this graph $\mathrm{ucl}(v, u) = 3$ and $\mathrm{ucl}(v, w) = 2$.

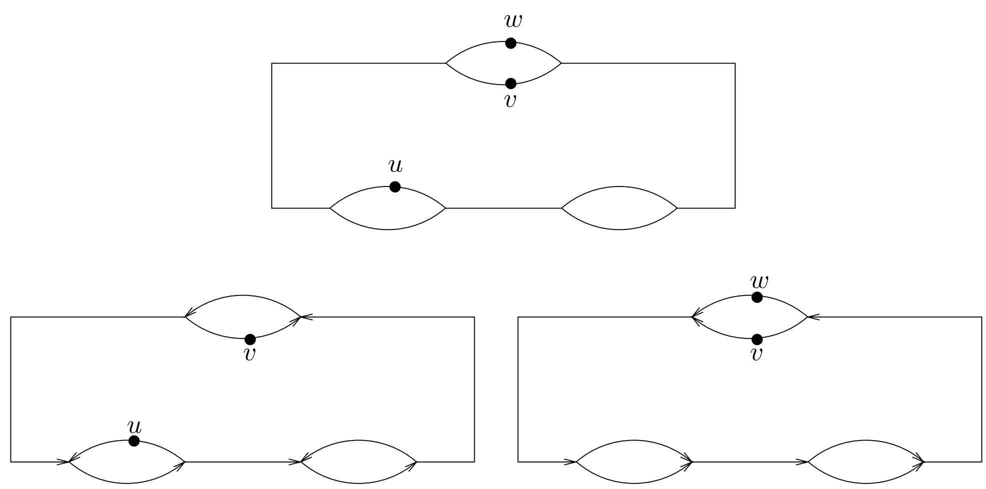

Figure 16: $r > 2$

Without loss of generality we can assume the graph is oriented in one of the orientations shown in Figure 16. In Figure 16 (lower left) $\mathrm{dcl}(v, u) = 7$ so $r > 2$ for this orientation. In Figure 16 (lower right) $\mathrm{dcl}(v, w) = 6$, again giving $r > 2$. ∎

Other obviously interesting open problems concern the complexity of computing minimum cycle-radius and minimum cycle-diameter orientations for special classes of graphs such as planar and series-parallel graphs.

References

[1] E. M. Arkin and R. Hassin. A note on orientations of mixed graphs. *Discrete Applied Mathematics*, **116** 271–278, 2002.

[2] L. Babai. On the diameter of Eulerian orientations of graphs. In *ACM-SIAM Proceedings of the Seventeenth Annual Symposium on Discrete Algorithms, SODA*, 822–831, 2006.

[3] W. W. Bein, P. Brucker and A. Tamir. Minimum cost flow algorithms for series-parallel networks, *Discrete Applied Mathematics* **10** 117-124, 1985.

[4] R. E. Burkard, K. Feldbacher, B. Klinz, and G. J. Woeginger. Minimum-cost strong network orientation problems: classification, complexity, and algorithms. *Networks*, **33** 57–70, 1999.

[5] V. Chvátal and C. Thomassen. Distances in orientations of graphs. *J. Combinatorial Theory Ser. B*, **24** 61–75, 1978.

[6] R. J. Duffin. Topology of series-parallel networks. *J. Math. Anal. Appl.*, **10** 303–318, 1965.

[7] N. Eggemann and S. D. Noble. Minimizing the oriented diameter of a planar graph *Electronic Notes in Discrete Mathematics*, **34** 267–271, 2009.

[8] F. V. Fomin, M. Matamala, E. Prisner, and I. Rapaport. AT-free graphs: linear bounds for the oriented diameter. *Discrete Appl. Math.*, **141** 135–148, 2004.

[9] F. V. Fomin, M. Matamala, and I. Rapaport. Complexity of approximating the oriented diameter of chordal graphs. *J. Graph Theory*, **45** 255–269, 2004.

[10] A. Frank, T. Király, and Z. Király. On the orientation of graphs and hypergraphs. *Discrete Appl. Math.*, **131** 385–400, 2003.

[11] N. Guttmann-Beck and R. Hassin. Minimum diameter and cycle-diameter orientations in planar graphs. arXiv:1105.4770v1 [cs] 2011.

[12] R. Hassin and N. Megiddo. On orientations and shortest paths. *Linear Algebra Appl.*, **114/115** 589–602, 1989.

[13] B. Jackson. Some remarks on arc-connectivity, vertex splitting, and orientation in graphs and digraphs. *J. Graph Theory*, **12** 429–436, 1988.

[14] K. M. Koh and E. G. Tay. Optimal orientations of graphs and digraphs: a survey. *Graphs Combin.*, **18** 745–756, 2002.

[15] A. Medvedovsky, V. Bafna, U. Zwick and R. Sharan. An algorithm for orienting graphs based on cause-effect pairs and its applications to orienting problem networks. In K. A. Crandall and J. Lagergren *Algorithms in Bioinformatics* Proceedings of WABI2008, LNBI **5251** 222-232, 2008.

[16] C. St. J. A. Nash-Williams. Well-balanced orientations of finite graphs and unobtrusive odd-vertex-pairings. In *Recent Progress in Combinatorics (Proc. Third Waterloo Conf. on Combinatorics, 1968)*, pages 133–149. Academic Press, New York, 1969.

[17] A. Schrijver. *Combinatorial Optimization. Polyhedra and Efficiency. Volume B, §61.3.* Springer-Verlag, Berlin, 2003.

[18] W. C. Yen. The edge orientation problem and some of its variants on weighted graphs. *Information Sciences*, **176** 2791–2816, 2006.