

Acyclic orientation of drawings

Citation for published version (APA):

Ackerman, E., Buchin, K., Knauer, C., & Rote, G. (2010). Acyclic orientation of drawings. *Journal of Graph Algorithms and Applications*, 14(2), 367-384.

Document status and date:

Published: 01/01/2010

Document Version:

Publisher's PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:

- A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
- The final author version and the galley proof are versions of the publication after peer review.
- The final published version features the final layout of the paper including the volume, issue and page numbers.

[Link to publication](#)

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license above, please follow below link for the End User Agreement:

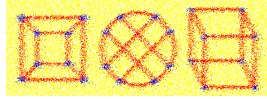
www.tue.nl/taverne

Take down policy

If you believe that this document breaches copyright please contact us at:

openaccess@tue.nl

providing details and we will investigate your claim.



Acyclic Orientation of Drawings

*Eyal Ackerman*¹ *Kevin Buchin*² *Christian Knauer*³
*Günter Rote*⁴

¹Department of Mathematics, Physics, and Computer Science,
University of Haifa at Oranim, Tivon 36006, Israel.

²Department of Mathematics and Computer Science,
Technical University Eindhoven, The Netherlands.

³Institute of Computer Science,
Universität Bayreuth, 95440 Bayreuth, Germany.

⁴Institute of Computer Science,
Freie Universität Berlin, Takustr. 9, 14195 Berlin, Germany.

Abstract

Given a set of pseudosegments in the plane or a topological graph, we ask for an orientation of the pseudosegments or edges which induces an acyclic orientation on the corresponding planar map. Depending on the maximum number of crossings on a pseudosegment or an edge, we provide algorithms and hardness proofs for this problem.

Submitted: February 2010	Accepted: August 2010	Final: August 2010	Published: November 2010
Article type: Regular paper		Communicated by: S.G. Kobourov	

Work by the first author was done while he was visiting the Freie Universität Berlin, and was partly supported by a Marie Curie scholarship. Research by the second author was supported by the Deutsche Forschungsgemeinschaft within the European graduate program “Combinatorics, Geometry, and Computation” (No. GRK 588/2). A preliminary version of this paper appeared in [1].

E-mail addresses: ackerman@sci.haifa.ac.il (Eyal Ackerman) k.a.buchin@tue.nl (Kevin Buchin)
christian.knauer@uni-bayreuth.de (Christian Knauer) rote@inf.fu-berlin.de (Günter Rote)

1 Introduction

Let G be a *topological graph*, that is, a graph drawn in the plane such that its vertices are distinct points, and its edge set is a set of Jordan arcs, each connecting two vertices and containing no other vertex. In this work we further assume that G is a *simple* topological graph, i.e., every pair of its edges intersect at most once, either at a common vertex or at a crossing point.

An *orientation* of (the edges of) a graph is an assignment of a direction to every edge in the graph. We say that an orientation is *acyclic* if the resulting directed graph does not contain a directed cycle. Finding an acyclic orientation of a given undirected (abstract) graph can be easily computed in linear time by performing a depth-first search on the graph and then orienting every *backward* edge from the ancestor to the descendant. However, is it always possible to find an orientation of the edges of a topological graph, such that a traveler on that graph will not be able to return to his starting position even if allowed to move from one edge to the other at their crossing point? Rephrasing it in a more formal way, let $M(G)$ be the planar map induced by G . That is, the map obtained by adding the crossing points of G as vertices, and subdividing the edges of G accordingly. Then we ask for an orientation of the edges of G such that the induced directed planar map $M(G)$ is acyclic.

Clearly, if the topological graph is *x-monotone*, that is, every vertical line crosses every edge at most once, then one can orient each edge from its endpoint with the smaller x -coordinate towards its endpoint with the greater x -coordinate. Traveling on the graph under such orientation, one always increases the value of one's x -coordinate and therefore cannot form a directed cycle. Fig. 1(a) provides an example for a topological graph for which there is no acyclic orientation: The edges e_1, e_2, e_3 cannot be all oriented clockwise or counter-clockwise, so assume w.l.o.g. that e_1 and e_2 are oriented clockwise, while e_3 is oriented counter-clockwise as in Fig. 1(b). To prevent a cycle with e_1 and e_3 , e_6 must be oriented downwards. Similarly, e_5 must be directed leftwards to prevent a cycle with e_2 and e_3 (see Fig. 1(c)). However, this yields the cycle shown in Fig. 1(d). Note that the degree of every vertex in this example is one. This gives rise for considering the orientation problem in the special case the degree of each vertex is one, or in other words, when one looks for an acyclic orientation of a set of *pseudosegments* embedded in the plane. Recall that a set of *pseudosegments* is a set curves such that every pair of curves intersects in at most one point.

It turns out that determining whether a topological graph (resp., a set of curves) has an acyclic orientation depends crucially on the maximum number of times an edge in the graph (resp., a pseudosegment) can be crossed. Given a (simple) topological graph G on n vertices, such that each edge in G is crossed at most once, we show that one can find an acyclic orientation of G in $O(n)$ time. When four crossings per edge are allowed, deciding whether there exists an acyclic orientation becomes NP-complete. Topological graphs with few crossings per edge were considered in several works in the literature [5, 8, 9]. For a set of n pseudosegments in which every pseudosegment is crossed at most k times, we

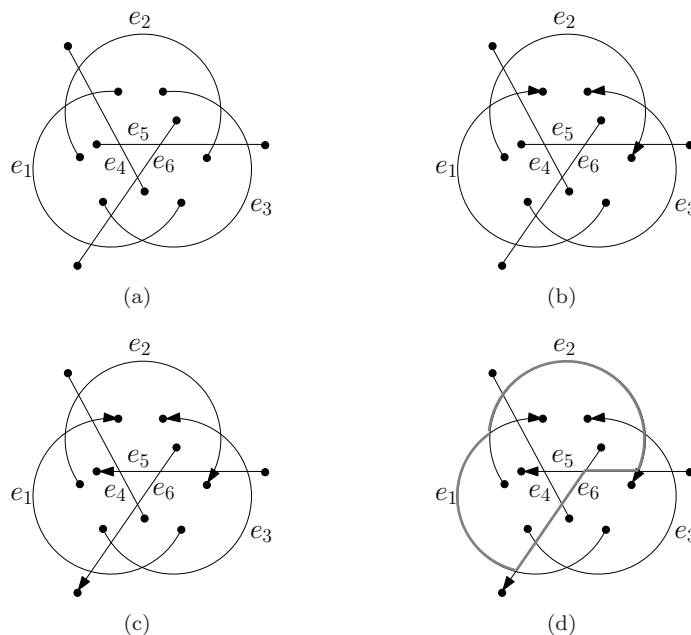


Figure 1: A non-orientable topological graph

describe an $O(n)$ -time orientation algorithm for the case $k \leq 3$. When $k \geq 5$ finding an acyclic orientation of the set of pseudosegments is NP-complete.

The rest of this paper is organized as follows. In Section 2 we study the problem of finding an acyclic orientation for a set of pseudosegments. Then, in Section 3 we consider the more general case where the input is a topological graph. Finally, we give some concluding remarks in Section 4, and mention a few related open problems.

2 Acyclic orientation of a set of pseudosegments in the plane

Throughout this paper we assume the intersections between the pseudosegments are known in advance. Given a set of pseudosegments \mathcal{C} , the vertices of the planar map $M(\mathcal{C})$, induced by \mathcal{C} , are the crossing points between the pseudosegments. The edges of $M(\mathcal{C})$ are segments of the pseudosegments that connect two consecutive crossing points on a pseudosegment. As we have mentioned above, the maximum number of crossings per pseudosegment plays an important role when we ask for an acyclic orientation of a set of pseudosegments. If every pseudosegment is crossed at most once, then $M(\mathcal{C})$ contains no edges, and therefore any orientation of \mathcal{C} is acyclic. If \mathcal{C} is a set of pseudosegments with

at most two crossing points per pseudosegment, then $M(\mathcal{C})$ is a union of cycles and paths and thus finding an acyclic orientation of \mathcal{C} is also easy in this case. Hence, the first non-trivial case is where each pseudosegment is crossed at most three times. In this case we have:

Theorem 1 *Let \mathcal{C} be a set of n pseudosegments, such that each pseudosegment has at most three crossings. Then, there exists an acyclic orientation of \mathcal{C} , and such an orientation can be found in $O(n)$ time.*

This result is proved in Section 2.1, while in Section 2.2 we show:

Theorem 2 *Let \mathcal{C} be a set of pseudosegments, such that each pseudosegment has at most five crossings. Then deciding whether \mathcal{C} has an acyclic orientation is NP-complete.*

2.1 Pseudosegments with at most three crossings per pseudosegment

Proof: [Proof of Theorem 1] Let \mathcal{C} be a set of n pseudosegments, such that each pseudosegment has at most three crossings. In this section we describe an algorithm for obtaining an acyclic orientation of \mathcal{C} . We start by constructing $M(\mathcal{C})$, the planar map induced by \mathcal{C} . Clearly, an (acyclic) orientation of \mathcal{C} induces an (acyclic) orientation of the edges in $M(\mathcal{C})$.

Every connected component of $M(\mathcal{C})$ can be oriented independently, therefore we describe the algorithm assuming $M(\mathcal{C})$ is connected. Suppose \mathcal{C} contains a pseudosegment c which is crossed less than 3 times. By removing c we obtain a set of $n - 1$ pseudosegments in which there must be at least two pseudosegments (the ones crossed by c) which are crossed at most twice. We continue removing the pseudosegments, until none is left. Then we reinsert the pseudosegments in a reverse order (the last to be removed will be the first to be reinserted and so on). During the insertion process we reconstruct $M(\mathcal{C})$ and define a total order of its vertices. For this purpose we store the vertices of $M(\mathcal{C})$ in a data structure suggested by Dietz and Sleator [2]. This data structure supports the following operations, both in $O(1)$ worst-case time:

1. **INSERT**(X, Y): Insert a new element Y immediately after the element X .
2. **ORDER**(X, Y): Compare X and Y .

Note that by inserting Y after X and then switching their labels we can also use this data structure to insert a new element immediately *before* an existing element in a constant time. We also keep a record of the maximal element in the order, **MAX** (that is, we update **MAX** when a new element is added after it or when switching the labels takes place for two elements one of which is **MAX**).

We now describe the way a pseudosegment c is reinserted. If c has no crossing points with pseudosegments that have already been added, then we orient it arbitrarily. Otherwise, for every pseudosegment c' that has already been added

and is crossed by c (recall that there are at most two such pseudosegments) we take the following actions. Let x be the crossing point of c and c' . If c' has no other crossing points, then x is inserted after MAX. In case c' has exactly one other crossing point x' , we insert x after x' when c' is oriented from x' to x , and before x' otherwise. Otherwise, suppose c' has two crossing points x'_1 and x'_2 , such that $x'_1 < x'_2$. Then we insert x before x'_1 if x'_1 is the middle point on c' among the three points; after x'_1 if x is the middle point; and after x'_2 if x'_2 is the middle point. Finally we orient c arbitrarily if it has less than two crossings, or from the smaller crossing to the greater one, in case it has two crossings. We refer to the algorithm described above as Algorithm 1.

Lemma 1 *Let \mathcal{C} be a set of n pseudosegments such that every pseudosegment is crossed at most three times and there is a pseudosegment that is crossed at most twice. Then Algorithm 1 finds an acyclic orientation of \mathcal{C} in $O(n)$ time.*

Proof: Since a total order is defined on the vertices of $M(\mathcal{C})$ and it is easy to verify that every edge is oriented from its smaller vertex to its greater one, it follows that there is no directed cycle in $M(\mathcal{C})$. Computing the connected components of $M(\mathcal{C})$ requires $O(n)$ time. Removing and adding a pseudosegment is performed in constant time, therefore the overall time complexity is $O(n)$. \square

The more complicated case is when all the pseudosegments in \mathcal{C} are crossed exactly three times. The general idea in this case is to:

1. find a set of pseudosegments S that form an undirected cycle in $M(\mathcal{C})$;
2. orient $\mathcal{C} \setminus S$ using Algorithm 1;
3. orient S such that:
 - (a) the pseudosegments in S do not form a directed cycle; and
 - (b) it is impossible to ‘hop’ on S from $\mathcal{C} \setminus S$, ‘travel’ on S , and ‘hop’ off back to $\mathcal{C} \setminus S$.

Henceforth, we assume that every pseudosegment in \mathcal{C} is crossed three times. A crossing point x on a pseudosegment c is *extreme* if in any orientation of c it is either the first or the last crossing point on c . Otherwise x is non-extreme.

Observe that the degree in $M(\mathcal{C})$ of a crossing point is 2, 3, or 4. We tackle the orientation problem based on whether or not there is a crossing point whose degree in $M(\mathcal{C})$ is 3. Suppose x_0 is a crossing point of degree 3, that is, it is the crossing point of two pseudosegments, c_0 and c , such that x_0 is an extreme crossing point on c and a non-extreme crossing point on c_0 . Denote by a_0 and a_1 (resp., a and b) the other crossing points on c_0 (resp., c), see Fig. 2(a).

We proceed by temporarily removing x_0 and looking for an undirected path in $M(\mathcal{C})$ between a_0 and a_1 . If there is no such path, then suppose that there is no path from a_1 to a (the case there is no path from a_0 to a is handled in a similar way). We can solve the orientation problem of \mathcal{C} by solving two sub-problems: first orienting the pseudosegments reachable from a_1 (without going

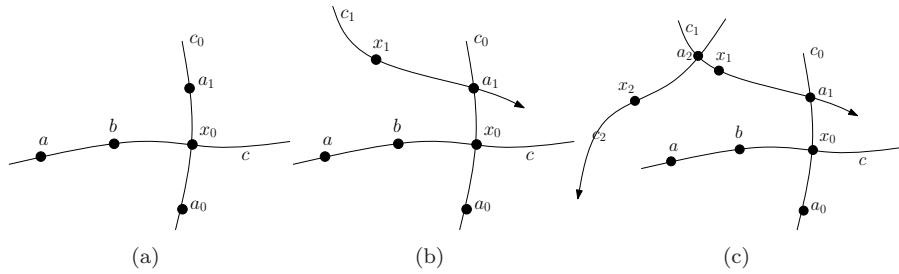


Figure 2: Orienting the pseudosegments where there is a degree 3 crossing point

through x_0), then orienting the rest of the pseudosegments (including c_0). Note that both sub-problems can be solved using Algorithm 1.

Now assume we have found a simple path p between a_1 and a_0 . Denote by c_1, c_2, \dots, c_k the pseudosegments on this path from a_1 to a_0 . Let C be the cycle formed by p , (a_0, x_0) and (x_0, a_1) , and let $S = \{c_0, c_1, \dots, c_k\}$. Our algorithm proceeds by finding an acyclic orientation of $C \setminus S$ (using Algorithm 1) and then orienting the pseudosegments in S . If $c \notin S$ and the direction of c in the acyclic orientation of $C \setminus S$ is from x_0 to a , then we switch the direction of every pseudosegment in $C \setminus S$. It is easy to see that the pseudosegments in $C \setminus S$ still do not form a directed cycle.

Next we provide the details of the orientation of the pseudosegments in S . Denote by a_i the crossing point of c_{i-1} and c_i , $i = 1, \dots, k-1$, and let $a_0 = a_{k+1}$ be the crossing point of c_0 and c_k . Suppose we traverse C starting at x_0 along the pseudosegments c_0, c_1, \dots, c_k , then the pseudosegment c_i is traversed from the point a_i to the point a_{i+1} , $i = 1, \dots, k$. Every pseudosegment has a third crossing point, that we denote by x_i , and refer to as the *connection point* of c_i . Note that x_i may or may not be on C , and that it is possible that $x_i = x_j$ for $i \neq j$. We say that x_i is a *before-connection point* if a_i is between x_i and a_{i+1} on c_i ; x_i is an *after-connection point* if a_{i+1} is between x_i and a_i on c_i ; and we say that x_i is a *middle-connection point* if it is not an extreme crossing point on c_i .

We will orient the pseudosegments in S such that there will be no directed path between two connection points through S (apart from some cases that will be discussed later on). We start by orienting c_1 from x_1 to a_1 (see Fig. 2(b)), thus making it impossible to “walk” from x_0 to x_1 using c_0 and c_1 . Next, we assign an orientation to c_2 . Note that the orientation of c_1 already prevents walking on c_1 and c_2 either from x_1 to x_2 , or from x_2 to x_1 . We assign orientation to c_2 such that both directions are impossible (see Fig. 2(c) for an example). We continue orientating the pseudosegments c_3, \dots, c_k in a similar way, making it impossible to reach x_i from x_{i+1} and the other way around, by using the pseudosegments c_i and c_{i+1} , for $i = 1, 2, \dots, k-1$. Finally, we set an orientation to c_0 as follows: In case x_k is already unreachable from x_0 (using the pseudosegments c_0 and c_k),

we set the orientation of c_0 such that x_0 is unreachable from x_k . Otherwise, we make sure x_k is unreachable from x_0 . Orienting S this way guaranties that for $i = 1, \dots, k - 1$ one cannot reach from x_i to x_{i+1} or vice versa, using c_i and c_{i+1} . It also guaranties that x_1 is unreachable from x_0 using c_0 and c_1 , and that x_k is unreachable from x_0 using c_0 and c_k .

Observation 1 *For every $i = 1, \dots, k$ there is no connection point $x_j \neq x_i$ such that there is a directed path on the pseudosegments in S from x_j to x_i .*

Proof: We prove the claim by induction on number of pseudosegments in the path between the two connection points. If the two connection points are on two crossing (that is, adjacent) pseudosegments, then the algorithm guaranties that there is no path from x_j to x_i (note that $x_i \neq x_0$). Suppose that there are x_i and x_j such that there is a directed path from x_j to x_i on S , and assume w.l.o.g. that c_{j+1} is the next pseudosegment on that path. Note that $c_{j+1} \neq c_0$, for otherwise there is path from x_0 to x_i which is shorter (in the number of traversed pseudosegments) than the path from x_j to x_i . Then, x_{j+1} must be a before-connection point, since otherwise there is a path from x_j to x_{j+1} using c_j and c_{j+1} . However, it follows that there is a path from x_{j+1} to x_i which is shorter than the path from x_j to x_i . \square

Observation 2 *C is not a directed cycle.*

Proof: Assume our orientation results in a directed cycle $a_0 \rightarrow a_1 \rightarrow \dots \rightarrow a_{k+1} = a_0$ (for a directed cycle in the other direction the proof is similar). According to the rules by which c_0 is oriented, it follows that x_k is an after-connection point (otherwise one can walk from x_k to x_0 on c_k and c_0). Considering the orientation of c_k and c_{k-1} and the fact there is no path using them from x_{k-1} to x_k or vice versa, one concludes that x_{k-1} is also an after-connection point. Proceeding in a similar manner implies that x_1 is also an after-connection point. However, in this case c_1 should have been oriented in the reverse direction in order to prevent a path from x_0 to x_1 using c_0 and c_1 . \square

Since C is not a directed cycle and Algorithm 1 finds an acyclic orientation of the pseudosegments in $\mathcal{C} \setminus S$, it remains to verify that there is no cycle that involves pseudosegments from S and from $\mathcal{C} \setminus S$. If there is such a cycle then, when traversing it, one must ‘hop’ on S at some connection point, ‘travel’ on S for a while, and then ‘hop’ off S . However, it follows from Observation 1 that one can ‘hop’ off S only at x_0 . Since c is directed from a to x_0 and x_0 is an extreme crossing point on c , the way from x_0 on c cannot be part of a directed cycle.

Finally, we have to consider the case where every pseudosegment in \mathcal{C} is crossed exactly three times, but there is no crossing point whose degree is three (see Fig. 3 for an example).

In this case we first look for an undirected cycle (there must be one as the degree of every vertex in $M(\mathcal{C})$ is at least 2). During our search, after arriving at a vertex v through one of the two pseudosegments defining v , we leave through

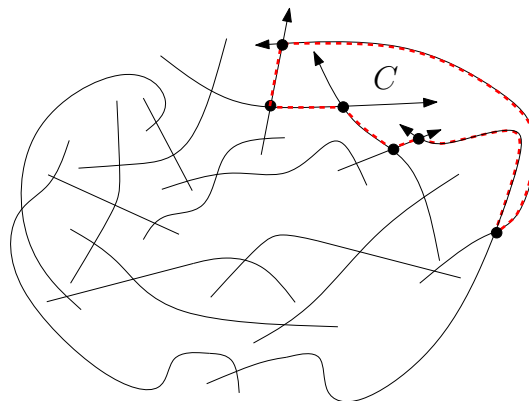


Figure 3: A set of pseudosegments whose planar map contains no degree 3 vertices. We orient it by first finding a cycle C and orienting its pseudosegments in an alternating manner along C . Then the remaining pseudosegments are oriented using Algorithm 1.

the other pseudosegment. Let C be the undirected cycle found, and denote by c_1, c_2, \dots, c_k the pseudosegments forming C (in that order). Again, we first orient the pseudosegments in $C \setminus \{c_1, c_2, \dots, c_k\}$ using Algorithm 1, and then assign orientation to the pseudosegments c_1, c_2, \dots, c_k . Let a_1, a_2, \dots, a_k be the vertices of C , and let x_1, x_2, \dots, x_k be the connection point of c_1, c_2, \dots, c_k , respectively. Note that two crossing points of the same degree cannot be adjacent. Thus, the degree of the points a_1, a_2, \dots, a_k must alternate between 2 and 4, and k must be even. It also follows from the way we search for a cycle, that the connection points x_1, x_2, \dots, x_k are alternating ‘before’ and ‘after’-connection points. By orienting the pseudosegments c_1, c_2, \dots, c_k in an alternating manner (see Fig. 3) we make sure that C is not a directed cycle, and that it is impossible to ‘hop’ on C at some connection point and then ‘hop’ off at another connection point. Therefore, the resulting orientation is acyclic. Let us refer to the algorithm describe above for the case every pseudosegment is crossed exactly three times as Algorithm 2.

Lemma 2 *Let C be a set of n pseudosegments such that every pseudosegment is crossed exactly three times. Then Algorithm 2 finds an acyclic orientation of C in $O(n)$ time.*

Proof: It follows from the correctness of Algorithm 1 and from the discussion above that Algorithm 2 finds an acyclic orientation of C . Note that a cycle is removed only once, and then it is guaranteed that there will be some pseudosegments with less than three crossings (in each connected component), and therefore we can apply Algorithm 1 on the remaining pseudosegments. All the operations concerning finding the cycle, removing it, and orienting the involved

pseudosegments can be performed in $O(n)$. Thus the overall time complexity is linear in the number of pseudosegments. \square

Combining lemmata 1 and 2 completes the proof of Theorem 1. \square

2.2 Pseudosegments with at most five crossings per pseudosegment

In this section we show that deciding whether there exists an acyclic orientation of a set of pseudosegments with at most 5 crossings per pseudosegment is intractable. We will reduce this problem from the problem of 2-coloring 3-uniform hypergraphs. A hypergraph $H = (V, E)$ is 3-uniform if every edge in E contains exactly three vertices. A (legal) 2-coloring of H is a function $f: V \rightarrow \{1, 2\}$ such that no edge in E contains only vertices of the same color. The problem of deciding whether a 3-uniform hypergraph has a legal 2-coloring is NP-complete [7].

Given a 3-uniform hypergraph H we draw in the plane a (not necessarily planar) bipartite graph representing the vertex-edge incidences of H as is shown in Fig. 4(a), with vertices aligned on a horizontal line and the edges aligned on a vertical line. To distinguish the crossings in this drawing from crossings of pseudosegments, we call them *extra-crossings*. The rectangles representing vertices, edges and extra-crossings will be replaced by sets of pseudosegments which we call *vertex-gadgets*, *edge-gadgets* and *crossing-gadgets*. We refer to the part of the gadget inside of its bounding rectangle as *inside*, and to the remainder as *outside*. For each of the gadgets some of the pseudosegments are partially outside of the gadget and are used to connect gadgets; we call these pseudosegments *wires*. We refer to (the orientation of) the wires as *ingoing* or *outgoing* relative to the corresponding gadget. Thus, a wire is always ingoing for one of the gadgets it connects to and outgoing for the other.

The basic idea of the reduction is the following: a vertex-gadget will have the property that in an acyclic orientation of its pseudosegments either all wires are ingoing or all wires are outgoing. Thus, the color of a vertex is represented by the orientation of the wires of its gadget. An edge-gadget will have the property that it can be oriented acyclically if and only if not all wires have the same orientation. A crossing-gadget propagates the orientations, i.e., an ingoing wire on one side should have an outgoing one on the other, and vice-versa. If we would ignore extra-crossings and would let wires between vertex- and edge-gadgets cross, then pseudosegments might have more than 5 crossings, and the extra-crossings might induce unintended oriented cycles. The purpose of crossing-gadgets is to ensure that this does not happen.

The simplest gadget is the edge-gadget shown in Fig. 4(b). The pseudosegments of an edge-gadget form an undirected cycle. There are exactly two cases in which this cycle is also a directed cycle: it might be directed clockwise in which case all wires are ingoing, or it might be directed counterclockwise in which case all wires are outgoing.

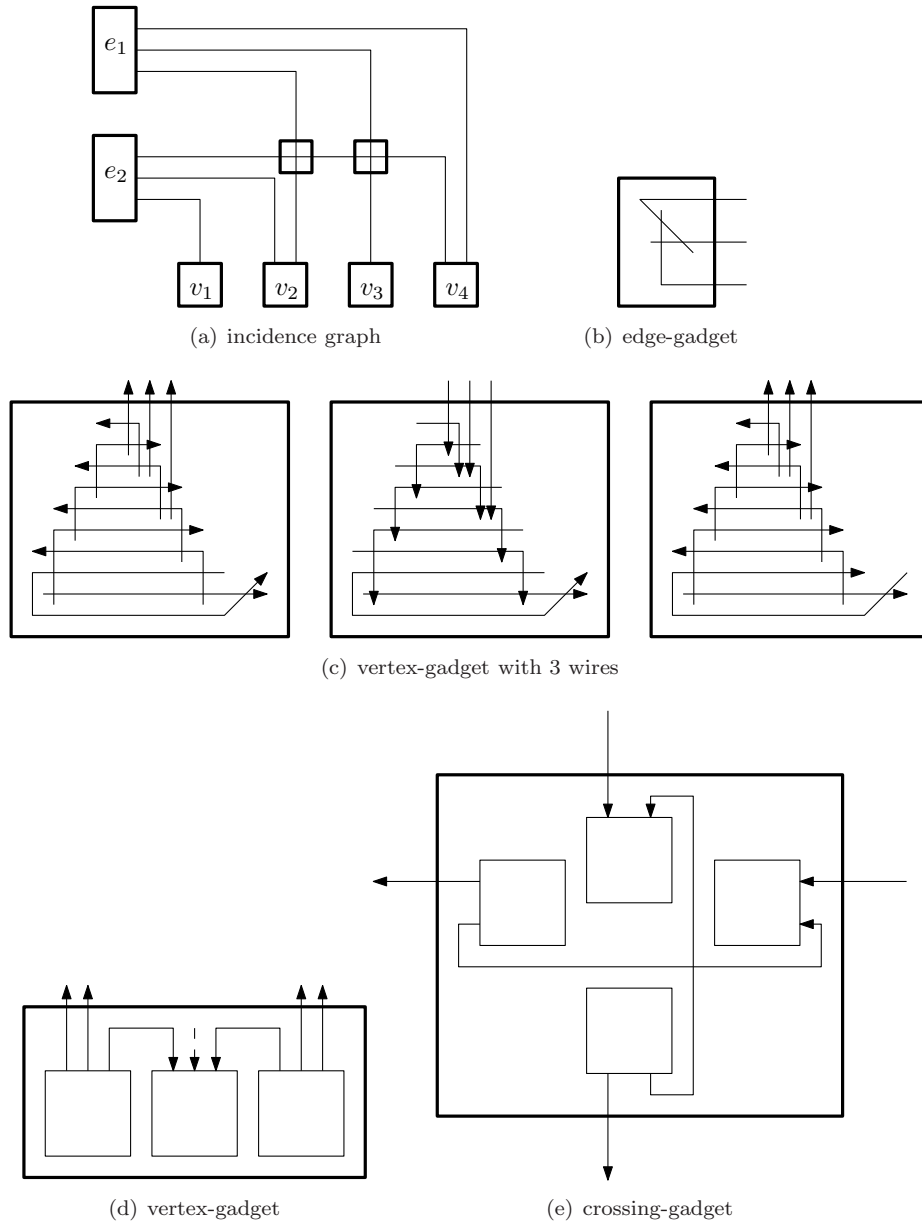


Figure 4: Constructions with pseudosegments in the reduction from 2-coloring 3-uniform hypergraphs

Lemma 3 *The pseudosegments of an edge-gadget form a directed cycle if and only if either all wires are ingoing or all wires are outgoing. Each of the wires has 2 crossings inside the gadget.*

Next we construct the vertex-gadget. We first give a construction for a vertex-gadget having 3 wires in Fig. 4(c). Up to reorienting all pseudosegments, the three orientations shown are exactly the acyclic orientations. This can be easily verified by orienting the pseudosegments in the order bottom to top in the figure; after three (left and middle case) or two (right case) pseudosegments, the previous pseudosegments already force the orientation of the next one and so on. We can connect several of these gadgets as shown in Fig. 4(d) to obtain a vertex-gadget with any number of wires by repeating the construction in the figure.

Lemma 4 *In an acyclic orientation of the pseudosegments of a vertex-gadget either all wires are ingoing or all wires are outgoing. Each of the wires has 2 crossings inside the gadget, and the wires do not cross each other inside the gadget.*

We construct the crossing-gadget by connecting vertex gadgets as shown in Fig. 4(d). All small rectangles in the figure are vertex-gadgets.

Lemma 5 *In an acyclic orientation of the pseudosegments of a crossing-gadget an ingoing wire has an outgoing wire on the opposite side and vice-versa. Each of the wires has 2 crossings inside the gadget, and the wires do not cross each other inside the gadget.*

Next we observe that we do not introduce unintended directed cycles. For a vertex-gadget this is a direct consequence of Lemma 4. For a crossing-gadget this follows from the fact that any such cycle would go through a vertex-gadget in the inside of a crossing-gadget.

Lemma 6 *If a set \mathcal{C} of oriented pseudosegments includes a vertex- or crossing-gadget that has no directed cycle inside, then \mathcal{C} has no directed cycle going through the inside of the gadget.*

Proof: [Proof of Theorem 2] An acyclic orientation can be verified in polynomial time, therefore the problem is in NP. The problem is shown to be NP-hard by reduction from 2-coloring 3-uniform hypergraphs to the acyclic orientation problem using the construction described above. All pseudosegments are polygonal chains of constant complexity. The edge- and the crossing-gadget use a constant number of pseudosegments, and the vertex-gadget of a vertex of degree k has complexity $O(k)$. For a 3-uniform hypergraph the set of pseudosegments can be constructed in polynomial time using polynomial coordinates for the vertices. By construction all pseudosegments have at most 5 crossings and cross pairwise at most once. It remains to prove that the set of pseudosegments has an acyclic orientation if and only if the hypergraph is 2-colorable.

Given a 2-colorable hypergraph $H = (V, E)$ with a 2-coloring $f: V \rightarrow \{1, 2\}$ we orient the pseudosegments in the following way. For a vertex $v \in V$, if

$f(v) = 1$, we orient the wires of the vertex-gadget representing v outgoing, otherwise ingoing. Using an orientation as given in Fig. 4(b,c), we can orient the pseudosegments inside the vertex-gadget acyclically. At crossings we propagate the orientations, i.e., for an ingoing wire we have an outgoing one on the opposite site and vice-versa. Using orientations from Fig. 4(d) (and (a) for the vertex-gadgets), we again obtain an acyclic orientation of the pseudosegments inside the gadget. For an edge $e \in E$, since f is a 2-coloring, e contains vertices of both colors. Therefore, the gadget for e has at least one ingoing and one outgoing wire. By Lemma 3 this induces an acyclic orientation inside the edge-gadget. Thus, we have no directed cycle in any of the gadgets. Furthermore, by Lemma 6 there is no directed cycle using parts inside of a vertex- or crossing-gadget. Thus, to see whether there is any directed cycle we can remove the insides of these gadgets from the corresponding planar map. The remaining parts of the planar map are isolated pseudosegments and acyclically-oriented edge-gadgets. Therefore, the orientation is acyclic.

Now, assume we are given an acyclic orientation for the set of pseudosegments constructed for a hypergraph $H = (V, E)$ and want to construct a 2-coloring of H . At a vertex-gadget either all wires are ingoing or all outgoing by Lemma 4. If for a vertex $v \in V$ all wires at its gadgets are outgoing we set $f(v) = 1$, otherwise we set $f(v) = 2$. Since the orientation is acyclic, the orientation of a wire is propagated at a crossing-gadget by Lemma 5. Thus, a wire at an edge-gadgets is ingoing if it was outgoing at the corresponding vertex-gadget, and outgoing if it was ingoing at the vertex-gadget. Again using that the orientation is acyclic, we can conclude from Lemma 3 that at any edge-gadget there are ingoing and outgoing wires, corresponding to vertices with colors 1 and 2. Thus, f is a legal 2-coloring. □

3 Acyclic orientation of topological graphs

Given a topological graph in which no edge is crossed, one can use the simple algorithm for abstract graphs described in the Introduction to find an acyclic orientation. Thus, the first non-trivial case is when every edge is crossed as most once. In Section 3.1 we show that in this case we have:

Theorem 3 *Let G be a topological graph on n vertices in which every edge is crossed at most once. Then G has an acyclic orientation. Moreover, such an orientation can be found in $O(n)$ time.*

In Section 3.2 we show:

Theorem 4 *Let G be a simple topological graph on n vertices in which every pair of edges crosses at most once and each edge has at most four crossings. Then deciding whether G has an acyclic orientation is NP-complete.*

3.1 Topological graphs with at most one crossing per edge

Before proving Theorem 3 we recall some basic terms and facts from graph theory.

Definition 5 An undirected graph $G = (V, E)$ is biconnected if there is no vertex $v \in V$ such that $G \setminus \{v\}$ is not connected.

Definition 6 A biconnected component (block) of a connected graph G is a maximal set of vertices that induce a biconnected subgraph.

Observation 3 Let G be an undirected graph, let C be a simple cycle in G , and let B be the block containing C . Then for every edge $e \notin C$ that connects two vertices of C , B is also a block of $G \setminus \{e\}$.

Definition 7 Given a biconnected graph $G = (V, E)$ and an edge $(s, t) \in E$, an st-numbering (or st-ordering) of G is a bijection $\ell : V \rightarrow \{1, 2, \dots, |V|\}$ such that: (a) $\ell(s) = 1$; (b) $\ell(t) = |V|$; and (c) for every vertex $v \in V \setminus \{s, t\}$ there are two edges $(v, u), (v, w) \in E$ such that $\ell(v) < \ell(u)$ and $\ell(v) > \ell(w)$.

Given an st-numbering, we will not make a distinction between a vertex and its st-number. An st-numbering of a graph G naturally defines an orientation of the edges of G : direct every edge (u, v) from u to v if $u < v$ and from v to u otherwise.

Lemma 7 ([10, Lemma 1]) Let $G = (V, E)$ be a plane biconnected multi-graph such that $|V| > 2$. Given an st-numbering, let G' denote the corresponding directed plane multi-graph. Let f be a face of G (and G'), and denote by G'_f (resp., G_f) the graph induced by the edges of G' (resp., G) bounding f . Then G'_f has exactly one source and one sink and consists of two directed paths from the source to the sink.

Proof: [Proof of Theorem 3] Let G be a simple topological graph in which every edge is crossed at most once. Denote by n the number of vertices in G , and by m the number of its edges. We will show that Algorithm 3 computes an acyclic orientation of G . Denote by G' the graph obtained after adding the edges in lines 1–3. Note that it is always possible to add the edges listed in line 2 without introducing new crossings. After this step the vertices of each crossing pair of edges lie on a simple 4-cycle. It is enough to verify that each block of G' is acyclicly oriented, since (a) every simple cycle in the underlying abstract graph is contained entirely in some block; and (b) the crossings do not introduce any interaction between different blocks, as all the vertices of a crossing pair of edges lie on a simple 4-cycle and therefore are in the same block. Thus, for the rest of the proof we assume G' is biconnected. We denote by G'' the graph obtained from G' by removing all the pairs of crossing edges. It follows from Observation 3 that G'' is biconnected, therefore, in line (7) an st-numbering of G'' is indeed computed.

Input: A topological graph G with at most one crossing per edge.

Output: An acyclic orientation of G .

```

1: for each pair of crossing edges  $(a, b)$  and  $(c, d)$  do
2:   add each of the edges  $(a, c)$ ,  $(a, d)$ ,  $(b, c)$ , and  $(b, d)$ ;
3: end for
4: compute the blocks of the new graph;
5: for each block  $C$  do
6:   temporarily delete all pairs of crossing edges in  $C$ ;
7:   compute an st-numbering of the remaining subgraph;
8:   reinsert all pairs of crossing edges in  $C$ ;
9:   orient each edge of  $C$  according to the st-numbering;
10: end for
11: remove the edges added in line 2;

```

Algorithm 3: Acyclic orientation of a topological graph with at most one crossing per edge

Clearly, one can obtain an acyclic orientation of an abstract graph by numbering the vertices of the graph and directing every edge from its endpoint with the smaller number to its endpoint with the larger number. Therefore, it is enough to verify that the crossing points do not introduce a *bad* “shortcut”, that is a path from a vertex u to a vertex v such that $v < u$. Let $((a, b), (c, d))$ be a pair of crossing edges. Denote by f the 4-face $a-c-b-d-a$ of G'' . According to Lemma 7 the digraph induced by f and the computed st-numbering has only one source and sink. Therefore, we have to consider only two cases based on whether the sink and the source are adjacent in f . One can easily verify by inspection that in both cases no bad shortcut is formed. Thus Algorithm 3 produces an acyclic orientation.

Note that Algorithm 3 can be implemented to run in time linear in the number of vertices: Finding the blocks of a graph takes $O(n + m)$ time [3], as does the computation of an st-numbering [4]. Therefore the overall running time is $O(n + m)$, however the maximum number of edges in a topological graph in which every edge is crossed at most once is $4n - 8$ [9], thus the time and space complexity of Algorithm 3 is $O(n)$. \square

Remark. We assume that G is a simple topological graph to simplify the presentation of the algorithm and its proof, however, it is actually not needed. An almost exact algorithm and proof will work for topological graphs in which every edge is crossed at most once, but two edges may cross and also share a common vertex.

In the next section we show that deciding whether there exists an acyclic orientation of a topological graph with at most four crossings per edge is NP-complete.

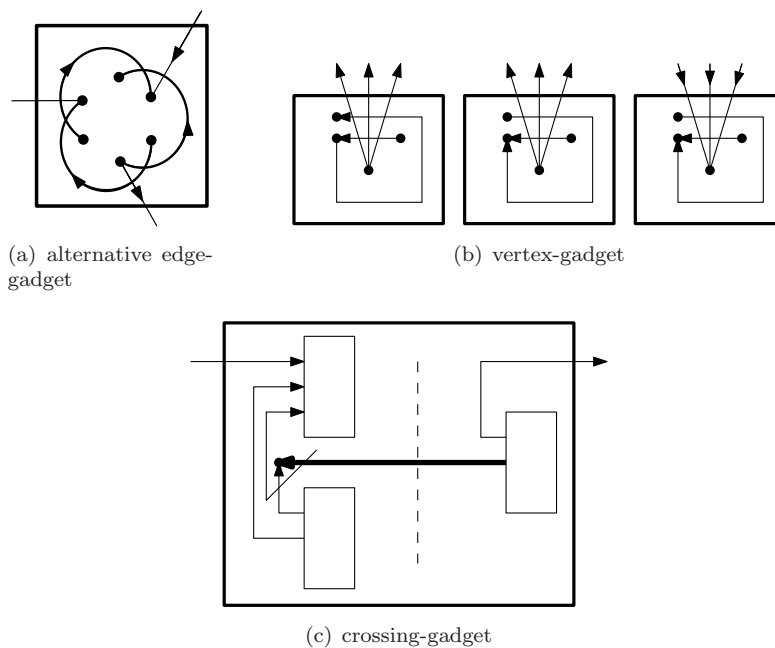


Figure 5: Constructions with graph edges in in the reduction from 2-coloring 3-uniform hypergraphs

3.2 Topological graphs with at most four crossings per edge

Proof: [Proof of Theorem 4] An acyclic orientation can be verified in polynomial time, therefore the problem is in NP. As for the case of a set of pseudosegments with at most 5 crossings per edge, we reduce from 2-coloring of 3-uniform hypergraphs. The reduction is the same as for pseudosegments, except for the vertex- and the crossing-gadget. We use the same edge-gadget but note that an edge-gadget with only one inside crossing per wire is possible (Fig. 5(a), see Observation 4), arcs are used to highlight the symmetry.

The vertex-gadget with three wires with all possible acyclic orientations (up to reorienting all edges) is shown in Fig. 5(b). To see that these are indeed all acyclic orientations, first note that the figure includes (up to reorienting both edges) all orientations of the two non-wire edges. In the first case (left figure) an ingoing wire would yield a counter-clockwise cycle with these two edges. In the second case (middle and right figure) any two wires form triangles with both of these edges. For one of the edges the corresponding triangle edge is oriented from left to right and for the other of the edges from right to left. Therefore, if one of the wires is ingoing and one is outgoing, for one of the non-wire edges we get a directed cycle. Thus, either all wires are outgoing or all are ingoing.

For vertex-gadgets with more wires we connect several of these in the same way as for pseudosegments. For the crossing-gadget we use the construction from Fig. 5(c) twice, once as shown and once rotated by 90 degrees. The edge that is drawn fat and its counterpart in the rotated version are used for the actual crossing. The remaining proof is then the same as for pseudosegments. \square

4 Discussion

We considered the problem of finding an acyclic orientation for a given topological graph or a set of pseudosegments in the plane. For topological graphs with at most one crossing per edge we showed an algorithm for finding an acyclic orientation in linear time. It follows from our results that when the maximum number of crossings per edge is at least four, deciding whether an acyclic orientation of the graph exists is NP-complete. An obvious open question is what happens when the maximum number of crossings per edge is two or three.

Observation 4 *There is a simple topological graph G , such that each edge has at most three crossings, and there is no acyclic orientation of G .*

Proof: Such a graph can be constructed with the gadgets of the NP-hardness proof in the case of at most four crossings per edge. The vertex-gadget (Fig. 5(b)) with only three wires uses at most three crossings per edge and two crossings for the wires. The wires all have the same orientation. The alternative edge-gadget (Fig. 5(a)) has at most three crossings per edge, and at most one crossing per wire.

To see that the edge-gadget works, consider the following argument. The three arcs form an undirected cycle. Therefore, to avoid a directed cycle one must be directed in the opposite direction of the other two. In the example of the figure this is the right arc, which is oriented counter-clockwise. As shown in the figure, this forces one wire to be ingoing and another one to be outgoing. The third can be chosen arbitrarily. Connecting one vertex-gadget to one edge-gadget yields a graph with at most three crossings per edge and no acyclic orientation. \square

It follows from Observation 4 that there are topological graphs with at most three crossings per edge that do not have an acyclic orientation. However, deciding whether a topological graph with at most three crossings per edge has an acyclic orientation is open. The situation is worse for topological graphs with at most two crossings per edge: So far we were unable to find an example which has no acyclic orientation, or to prove that every such graph is acyclic-orientable.

A special case is where all the vertices in the topological graph have degree 1. This case corresponds to asking the acyclic orientation question for a set of pseudosegments. Clearly, if the problem can be solved (or decided) for topological graphs with at most k crossings per edge, then it can be solved for pseudosegments with at most k crossings per pseudosegment. It would be interesting to

determine whether there is a construction that provides a reduction from topological graphs with at most k crossings per edge to a set of pseudosegments with at most k' crossings per pseudosegment.

For pseudosegments with at most three crossings per pseudosegment we provided a linear time algorithm that finds an acyclic orientation. For five crossings per pseudosegment we showed that the problem becomes NP-complete. A set of pseudosegments with at most four crossings per pseudosegment might not have an acyclic orientation, as Fig. 1 implies. However, the decision problem for such sets of pseudosegments is also open. Two other interesting open questions are: (1) What happens if we only require acyclic *faces*? and (2) What happens if we look for an orientation such that for every pair of vertices, u, v , in the induced planar map there is a directed path from u to v or vice versa?

For both graphs and sets of pseudosegments, one can look at the optimization problem where one wishes to find an orientation that minimizes the number of directed cycles. Is it possible to approximate the optimal solutions, and does the number of crossings per edge/pseudosegment also play a role when it comes to the approximation ratio one can obtain?

Our original motivation for considering the acyclic orientation of pseudosegments does not concern pseudosegments in the plane but in space. An important property [6, Lemma 10] of arrangements of double pseudolines in the Möbius band can be formulated in terms of the acyclicity of the 1-skeleton of a certain arrangement of oriented curves in a cylinder. It would be interesting to obtain results concerning acyclic orientations of curves on other surfaces. Also it would be interesting to study acyclic orientations of graphs with more general dependencies between or constraints on the orientation of edges.

Acknowledgments. We thank Michel Pocchiola for suggesting the problem concerning the acyclicity of pseudosegments in the plane. We also thank Scot Drysdale, Frank Hoffmann, and Klaus Kriegel for helpful discussions, and anonymous referees for several useful remarks.

References

- [1] E. Ackerman, K. Buchin, C. Knauer, and G. Rote. Acyclic orientation of drawings. In *Proc. 10th Scandinavian Workshop on Algorithm Theory (SWAT)*, volume 4059 of *Lecture Notes in Computer Science*, pages 268–279, Riga, Latvia, July 2006. Springer.
- [2] P. Dietz and D. Sleator. Two algorithms for maintaining order in a list. In *Proc. 19th Ann. ACM Symp. on Theory of Computing (STOC)*, pages 365–372, NYC, NY, 1987.
- [3] S. Even. *Graph Algorithms*. Computer Science Press, 1979.
- [4] S. Even and R. E. Tarjan. Computing an st-numbering. *Theoretical Computer Science*, 2(3):339–344, 1976.
- [5] A. Grigoriev and H. L. Bodlaender. Algorithms for graphs embeddable with few crossings per edge. *Algorithmica*, 49(1):1–11, 2007.
- [6] L. Habert and M. Pocchiola. Arrangements of double pseudolines. In *Proc. 20th ACM Symp. on Computational Geometry (SoCG)*, pages 314–323, New York, NY, USA, 2009. ACM.
- [7] L. Lovász. Coverings and colorings of hypergraphs. In *Proc. 45th South-eastern Conf. Combinatorics, Graph Theory, Computing*, pages 3–12, 1973.
- [8] J. Pach, R. Radoičić, G. Tardos, and G. Tóth. Improving the crossing lemma by finding more crossings in sparse graphs. *Discrete Comput. Geom.*, 36(4):527–552, 2006.
- [9] J. Pach and G. Tóth. Graphs drawn with few crossings per edge. *Combinatorica*, 17(3):427–439, 1997.
- [10] R. Tamassia and I. G. Tollis. A unified approach to visibility representations of planar graphs. *Discrete Comput. Geom.*, 1(4):321–341, 1986.