Dipartimento di Informatica e Automazione
Via della Vasca Navale, 79
00146 Roma, Italy

Finding Bimodal and Acyclic Orientations of Mixed Planar Graphs is NP-Complete

Maurizio Patrignani

RT-DIA-188-2011 **Aug 2011**

Università Roma Tre,
Via della Vasca Navale, 79
00146 Roma, Italy

This work was partially supported by MIUR of Italy under project AlgoDEEP prot. 2008TF-BWL4 and was written while the author was visiting the Visualization and Visual Computing Research Group at Peking University, Beijing, China. Part of the research was conducted in the framework of ESF project 10-EuroGIGA-OP-003 GraDR "Graph Drawings and Representations".

ABSTRACT

We investigate the computational complexity of the following problem. Given a "mixed" planar graph, i.e., a planar graph where some edges are directed, orient the remaining edges in such a way that the whole graph is acyclic and admits a bimodal planar embedding, that is, a planar embedding where edges entering (exiting) each vertex appear consecutively in its circular adjacency list. We show that this extendability problem is NP-complete by first determining its complexity in the fixed embedding setting and then extending the result to the variable embedding setting.

1 Introduction

A flourishing line of research investigates combinatorial problems where the solution is partially given in advance and the task is to extend such a partial solution to a complete one. In some cases the partial solution can be efficiently exploited to find a complete one. More often it makes hard an otherwise easy problem.

These extendability problems have been studied in various settings. As an example, determining the chromatic number of a perfect graph is polynomial, but becomes NP-complete if only four vertices are already colored [12]. Another example is provided by edge colorings, where the famous König-Hall theorem ensures that cubic bipartite graphs are always 3-edge colorable while the same problem becomes NP-complete if some edges are colored in advance [8].

Regarding the planarity of graphs, a notoriously linear problem, in some cases the corresponding extendability problem is also linear and in some others it becomes NP-hard. Namely, the planarity of partially drawn graphs can be tested in linear time [1] while the same problem is known to be NP-hard for drawings where edges are constrained to be straight-line segments [15].

The corresponding extendability problem for upward drawings of directed graphs (digraphs) has been recently addressed in [3]. It is known that deciding whether a planar digraph admits an upward planar drawing (the so-called upward planarity testing problem) is NP-complete in the variable embedding setting [9, 10] while it is polynomial for embedded planar digraphs [2]. If the input graph is undirected, finding an orientation which is compatible with an upward drawing of it is trivially linear and it is polynomial finding one with the minimum number of sources and sinks for 1-connected embedded planar graphs [6].

If a plane graph is only partially directed, finding an orientation for the remaining edges which is compatible with a planar upward drawing of it is a problem of unknown complexity and has been pointed out by several researchers [3, 11]. In [3] an ILP formulation has been proposed to solve this problem while in [7] a heuristic that attempts to compute an upward drawing of the graph with few crossings is described.

An orientation of a plane graph is *acyclic* if it does not induce any directed cycle, while it is *bimodal* if for each vertex of the graph all entering (and, hence, also all exiting) edges appear consecutively in its circular adjacency list. Acyclicity and bimodality are two necessary conditions for a plane directed graph to admit a planar upward drawing [2]. Hence, studying the problem of finding an acyclic and bimodal orientation for the edges of a partially directed graph is a possible strategy to approach the problem of finding an orientation that makes the graph upward planar.

In this paper we show that the following problem is NP-complete.

Problem: BIMODAL AND ACYCLIC ORIENTATION OF A MIXED PLANAR GRAPH (BAOMPG)

Instance: A planarly embedded *mixed graph* $G = (V, E_d, E_u)$ where edges in E_d are directed and edges in E_u are undirected.

Question: Does there exist an acyclic orientation of the edges in E_u that is compatible with a *bimodal* embedding of G, that is, an embedding of G such that all edges entering (and, hence, also all edges exiting) each vertex $v \in V$ appear consecutively in the circular adjacency list of v?

In Section 2 by reducing Planar-3SAT we show that BAOMPG is NP-hard in the fixed embedding setting, where the planar embedding of the input graph G can not be changed. In Section 3 we extend such result to the variable embedding setting. As the problem is clearly in NP (all possibile orientations of edges in E_u can be non-deterministically generated and the resulting directed graph tested for acyclicity and bimodality in linear time), this leads to the following theorem.

Theorem 1 BAOMPG *is NP-complete both in the fixed and in the variable embedding settings.*

In order to further investigate the problem, in Section 4 we drop the acyclicity requirement and show that the following problem is NP-hard both in the fixed and in the variable embedding settings.

Problem: Mixed Planar Graph Bimodality (MPGB)

Instance: A planar *mixed graph* $G = (V, E_d, E_u)$ where edges in E_d are directed and edges in E_u are undirected.

Question: Does there exist an orientation of the edges in E_u that is compatible with a *bimodal* embedding of G, that is, an embedding of G such that all edges entering (and, hence, also all edges exiting) each vertex $v \in V$ appear consecutively in the circular adjacency list of v?

Again, this brings to the following theorem.

Theorem 2 MPGB *is NP-complete both in the fixed and in the variable embedding settings.*

We observe that orienting the undirected edges of a mixed plane or planar graph such that the resulting digraph is acyclic can be done in linear-time. In fact a directed cycle in the primal graph is equivalent to a directed cut in its dual. Hence, the problem is equivalent to orienting the undirected edges of a mixed graph in such a way that the resulting digraph is strongly connected, which was addressed in [4, 16].

Section 5 contains our conclusions and open problems.

2 BAOMPG **NP-Hardness in the Fixed Embedding Setting**

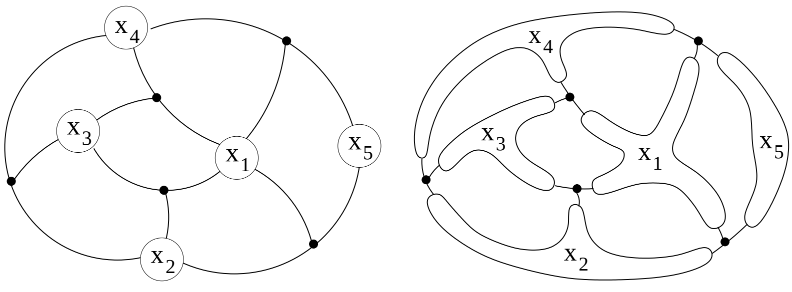
In order to show the NP-hardness of the Bimodal and Acyclic Orientation of a Mixed Planar Graph problem we produce a reduction from the Planar 3SAT (P3SAT) problem, which is NP-complete [13]. The problem is formally stated as follows:

Problem: **Planar 3-Satisfiability (P3SAT)**

Instance: A collection of clauses, where each clause consists of exactly three literals. Moreover, the bipartite graph $G(V_A, V_B, E)$, where nodes in V_A correspond to the variables while nodes in V_B correspond to the clauses and edges in E connect clauses to the variables of the literals they contain, is planar (see Fig. 1.a).

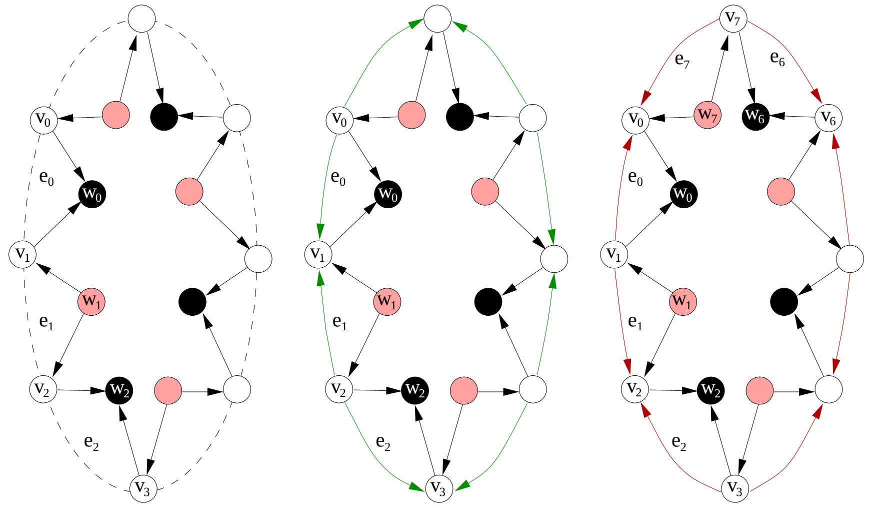
Question: Does there exist a truth assignment to the variables so that each clause has at least one true literal?

We recall that in a P3SAT instance each clause can be assumed to be composed of exactly three literals of different variables [14].



(a) (b)

Figure 1: (a) A planar embedding of graph $G(V_A, V_B, E)$ for the P3SAT instance $\phi = (x_1 \vee \neg x_2 \vee x_3) \wedge (x_2 \vee \neg x_3 \vee \neg x_4) \wedge (\neg x_1 \vee x_3 \vee \neg x_4) \wedge (x_1 \vee \neg x_4 \vee x_5) \wedge (x_1 \vee x_2 \vee \neg x_5)$. (b) The same instance where variables are replaced with cycles.



(a) (b) (c)

Figure 2: A variable gadget (a) and a true (b) and a false (c) orientation of it. Dashed edges are undirected edges in the input graph. The variable gadget is true if the edges of the cycle are oriented counter-clockwise in correspondence to sinks (black vertices) and clockwise in correspondence to sources (red vertices).

Starting from an instance of the P3SAT problem, consisting of the set of clauses $C_1, \ldots, C_m$, each one having three literals from the Boolean variables $x_1, \ldots, x_n$, and a drawing of the graph $G(V_A, V_B, E)$, an instance of the POPGP problem can be constructed as follows. We replace each vertex in V_A (i.e., corresponding to a variable) with partially-oriented cycles and attach vertices in V_B (i.e., corresponding to clauses) to such cycles (see Fig. 1.b). Details on the definitions of partially-oriented cycles are provided in Section 2.1, while the connections of cycles to the clause vertices is described in Sections 2.2 and 2.3. The overall proof of correctness is given in Section 2.4.

2.1 The Variable Gadget

For each variable x_i of the P3SAT instance we build a *variable gadget* depicted in Fig. 2(a). The variable gadget is composed of a cycle with $2k$ vertices $v_0, v_1, \ldots, v_{2k-1}$, where k is the degree of variable x_i, joined by undirected edges. Also, for each undirected edge $e_i = (v_i, v_{(i+1) \bmod k})$, with i even (odd, respectively), a vertex w_i is placed in the internal portion of the plane and connected with two directed edges (v_i, w) and $(v_{(i+1) \bmod k}, w)$ ((w, v_i) and $(w, v_{(i+1) \bmod k})$, respectively). Hence, vertices w_i are alternatively sinks (even i) and sources (odd i) in the mixed graph G.

Given an orientation for the undirected edges of G, we say that a variable gadget is *true* (*false*, respectively) if each edge e_i is directed counter-clockwise for even i and clockwise for odd i (clockwise for even i and counter-clockwise for odd i, respectively). See Figs. 2(b) and 2(c) for examples of true and false variable gadgets.

We have the following lemma.

Lemma 1 *In any bimodal orientation of the undirected edges of G, a variable gadget is either true or false.*

Proof: Consider edge $e_0 = (v_0, v_1)$ in Fig. 2(a). Suppose that in the bimodal orientation e_0 is directed counter-clockwise from v_0 to v_1 as in Fig. 2(b). The direction of the three edges (v_0, v_1), (v_1, w_0), and (w_1, v_1) which are adjacent in the circular list of v_1 forces edge e_1 to be directed clockwise from v_2 to v_1 in order for the orientation to be bimodal. Analogously, the clockwise orientation of e_1 and the bimodality of v_2 forces edge e_2 to be directed counter-clockwise from v_2 to v_3. It follows that each e_i is directed counter-clockwise when i is even and clockwise when i is odd, i.e., the variable gadget is true.

Converserly, suppose that in the bimodal orientation e_0 is directed clockwise from v_1 to v_0 as in Fig. 2(c). Analogous considerations allow to estabilish that e_7 is directed counter-clockwise. In turn, this implies that e_6 is directed clockwise, and so on. Hence, in this second case each e_i is directed clockwise when i is even and counter-clockwise when i is odd, i.e., the variable gadget is false. $\square$

A variable gadget will be connected to the rest of the graph by attaching edges to the vertices on the external boundary of it. Regarding the orientation of such edges the following lemma holds.

Lemma 2 *Consider any bimodal orientation of the undirected edges of G and suppose that a variable gadget is true. Any edge attached to a vertex v_i on the external boundary of the variable gadget is forced to be exiting v_i (entering v_i, respectively) if i is even (i is odd, respectively). Conversely, suppose that the variable gadget is false. Any edge attached to a vertex v_i on the external boundary of the variable gadget is forced to be entering v_i (exiting v_i, respectively) if i is even (i is odd, respectively).*

Proof: The statement immediately follows by observing Figs. 2(b) and 2(c) and considering the bimodality of the vertex v_i. $\square$

2.2 The Clause Gadget

The clause gadget for a clause $C = l_1 \vee l_2 \vee l_3$ is depicted in Fig. 3(a) and is composed of a cycle of three undirected edges e_a, e_b and e_c connecting three vertices v_1, v_2, and v_3.

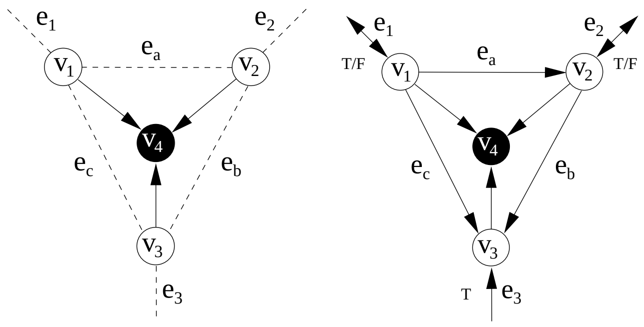


Figure 3: (a) The clause gadget. (b) In all bimodal and acyclic orientations of the clause gadget one among e_1, e_2, and e_3 must be exiting (the corresponding literal must be true).

A fourth vertex v_4 is placed in the interior side of the cycle and connected with three directed edges from v_1, v_2, and v_3. Three undirected edges e_1, e_2, e_3, exiting v_1, v_2, and v_3, respectively, connect the clause gadget to the three variable gadgets of the literals l_1, l_2, and l_3 of C_i as it is specified in Section 2.3. Given an orientation for the undirected edges of the clause gadget, we say that one of the edges v_1, v_2, and v_3 is *true* if it is directed entering the clause gadget, *false* otherwise.

Lemma 3 *In any bimodal and acyclic orientation of the undirected edges of the clause gadget one among* e_1*,* e_2*, and* e_3 *is true, while the other two may be either true or false.*

Proof: As the orientation is acyclic at least one among v_1, v_2, and v_3 has two entering edges among e_a, e_b, and e_c. Suppose without loss of generality that vertex v_3 has entering edges e_c and e_b (see Fig. 3(b)). From the bimodality of v_3 follows that e_3 is directed entering v_3, and hence it is true. Both the orientations of e_1 and e_2 are compatible with a bimodal and acyclic orientation of the clause gadget. $\square$

Observe that, in order for the orientation of the whole graph to be acyclic it is necessary to check that no directed cycles uses the sequence e_1, e_a, e_2. This will be ensured by suitably attaching the clause gadgets to the variable gadgets as described in Section 2.3.

2.3 The Connections among Gadgets

Consider a clause $C = l_1 \vee l_2 \vee l_3$ and the variable gadgets for the corresponding variables x_1, x_2, and x_3 (see Fig. 4). If l_i is the directed literal of variable x_i, we attach with an undirected edge e_i the vertex v_i of the clause gadget with a vertex v_p, with p even, of the variable gadget for x_i. Lemma 2 ensures that if the variable gadget for x_i is true, then e_i will exit v_p, and hence enter v_i (e_i is true). Analogously, if l_i is the negated literal of variable x_i, we attach with an undirected edge e_i the vertex v_i of the clause gadget with a vertex v_q, with q odd, of the variable gadget for x_i. Again, Lemma 2 ensures that if the variable gadget for x_i is false, then e_i will exit v_p, and hence enter v_i (e_i is true).

Theorem 3 of Section 2.4 ensures that any sequence of edges traversing a clause gadget (such as e_1, e_a, and e_2 of Fig. 4) can not be part of a directed cycle of the graph.

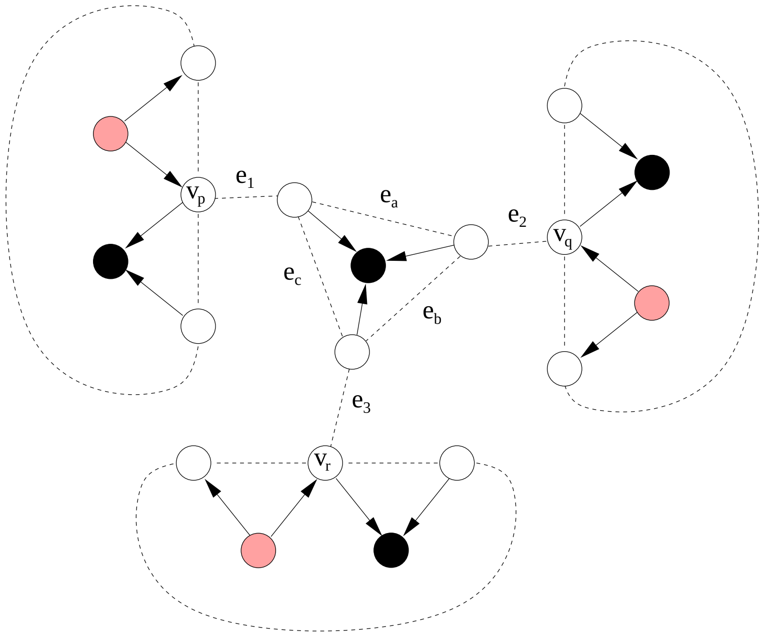


Figure 4: The connections among variable gadgets and clause gadgets.

2.4 Proof of Correctness

Theorem 3 *The* Bimodal and Acyclic Orientation of a Mixed Planar Graph *problem is NP-hard in the fixed embedding setting.*

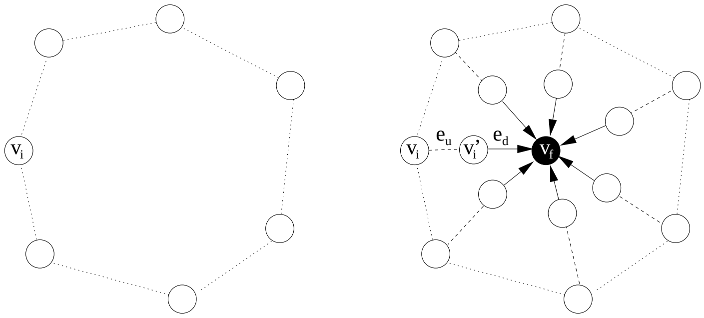
Proof: Suppose that the P3SAT instance admits a truth assignment such that each clause has at least one true literal. An orientation for the edges of G can be found as follows. Give to each variable gadget the true or false orientation depending on the truth value of the corresponding variable. Orient the edges attaching to the variable gadgets the orientation prescribed by Lemma 2. Select one true literal for each clause and orient the edges of the clause gadget in such a way that they enter the vertex corresponding to such a literal. It can be seen that the orientation is bimodal. In order to show that it is also acyclic, we show that by recursively removing sources and sinks we obtain the empty graph. First, observe that when the sources and sinks are removed from the interior part of each variable gadget, the boundary of the variable gadget is composed by alterante sinks and sources. Second, observe that when the central sink vertex of each clause gadget is removed, the vertex corresponding to the true literal is also a sink, and all the vertices of the clause can be removed yielding an empty graph.

Conversely, suppose that there exists an acyclic and bimodal orientation of the edges of G. An assignment of truth values to the variables $x_1, x_2, \ldots, x_n$ satisfying the corresponding P3SAT instance can be found as follows. From the orientation of each variable gadget a truth value for the corresponding variable can be obtained. The true literal of each clause can be obtained by observing the direction of the edges of the corresponding clause gadget.

Since, starting from a P3SAT instance, the construction of the corresponding BAOMPG instance can be done in polynomial time, the statement follows. □

3 Extension to the Variable Embedding Setting

The extension to the variable embedding setting simply consists of adding edges to the graph G constructed as described in Section 2 in order to obtain a graph G' which admits a single embedding up to a reversal of all its adjacency lists.



(a) (b)

Figure 5: The *triangulation* operation: (a) A face f with more than three vertices. (b) The same face after vertex v_f has been inserted and joined to the boundary of f with paths of length 2.

To pursue this target, we define the *triangulation* operation, which we perform on each face of the graph with more than three vertices (see Fig. 5). Namely, given a face f of G with more than three vertices, we insert into f a vertex v_f and join each vertex v_i on the boundary of f to v_f with a path of length two composed by one undirected edge $e_u = (v_i, v_i')$ and one directed edge $e_d = (v_i', v_f)$.

Lemma 4 *The graph G' obtained from G by triangulating each face of degree greater than three admits a single embedding up to a reversal of its adjacency lists.*

Proof: The statement follows from the observation that G' is a subdivision of a triangulated graph. □

Theorem 4 *The* Bimodal and Acyclic Orientation of a Mixed Planar Graph *problem is NP-hard in the variable embedding setting.*

Proof: By Theorem 3 graph G admits an orientation that is acyclic and bimodal if and only if the corresponding P3SAT instance admits a solution. Graph G' is obtained from G by triangulating each face of degree greater than three. We show that G' admits an acyclic and bimodal orientation if and only if G admits one. Suppose G' admits an acyclic and bimodal orientation. By Lemma 4 graph G' admits a single embedding up to a reversal of its adjacency lists, and such embedding by construction corresponds to the embedding of G described in Section 2. Remove all vertices and edges introduced to triangulate the faces of G to obtain an orientation for G which is acyclic and bimodal.

Conversely, suppose that G admits an acyclic and bimodal orientation. As any vertex v_f inserted during triangulation is a sink, it can not be traversed by any directed cycle. Hence, G' also is acyclic. Observe that undirected edges (v_i, v_i') introduced during triangulation can always be directed in such a way to leave vertex v_i bimodal. Hence there exists an acyclic and bimodal orientation for G'. □

4 Mixed Planar Graph Bimodality is NP-Hard

In this section we modify the reduction of Section 2 in order to show that asking for bimodality is sufficient to make the orientation of a mixed graph NP-hard. Again, we first consider the fixed embedding setting and then extend to the variable embedding setting.

Observe that in the reduction described in Section 2 the acyclicity requirements is only used for the clause gadget (see Lemma 3). Hence, the clause gadget is the only one that needs to be changed to show the NP-hardness of the MPGB problem.

4.1 The Clause Gadget for MPGB

In order to describe the clause gadget for MPGB we first introduce the *2SAT-gadget* shown in Fig. 6. The 2SAT-gadget is attached to the rest of the graph through three edges e_4, e_5, and e_8. Intuitively, the purpose of the 2SAT-gadget is check if the truth vaules encoded in the directions of e_4 and e_5 are both false, and encode the result in the direction of e_8.

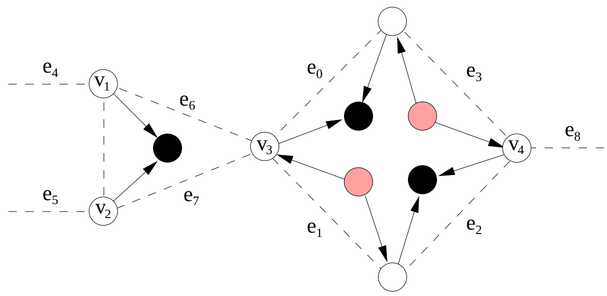


Figure 6: The 2SAT-gadget. If edges e_4 and e_5 are exiting the 2SAT-gadget, edge e_8 is entering it.

The main component of the 2SAT-gadget is a cycle of four edges e_0, e_1, e_2, and e_3, which form a construction analogous to that of the variable gadget described in Section 2.1 (actually, it is a small variable gadget built using $k = 1$). Analogously to the variable gadget, we say that the 2SAT-gadget is *true* (*false*, respectively) if each edge e_i is directed counter-clockwise for even i and clockwise for odd i (clockwise for even i and counter-clockwise for odd i, respectively). By Lemma 1 in any bimodal orientation of the undirected edges of G, the 2SAT-gadget is either true or false, and consequently (by Lemma 2) we have that edges e_6, e_7 and e_8 are either all entering v_3 and v_4 or all exiting v_3 and v_4. Edges e_4 and e_5 are not directly attached to v_3. The purpose of vertices v_1 and v_2 and of the edge between them is stated by the following lemma.

Lemma 5 *In any bimodal orientation of the undirected edges of the 2SAT-gadget, if edges e_4 and e_5 are directed exiting v_1 and v_2 edge e_8 is directed entering v_4, otherwise e_8 may have both the orientations.*

Proof: Suppose that e_4 and e_5 are directed exiting v_1 and v_2 (see Fig. 7(a)). Without loss of generality suppose that edge (v_1, v_2) is directed towards v_2. From the bimodality of v_2 follows that e_7 is directed entering v_3. Lemma 2 implies that e_6 is directed entering v_3 and e_8 is directed entering v_4.

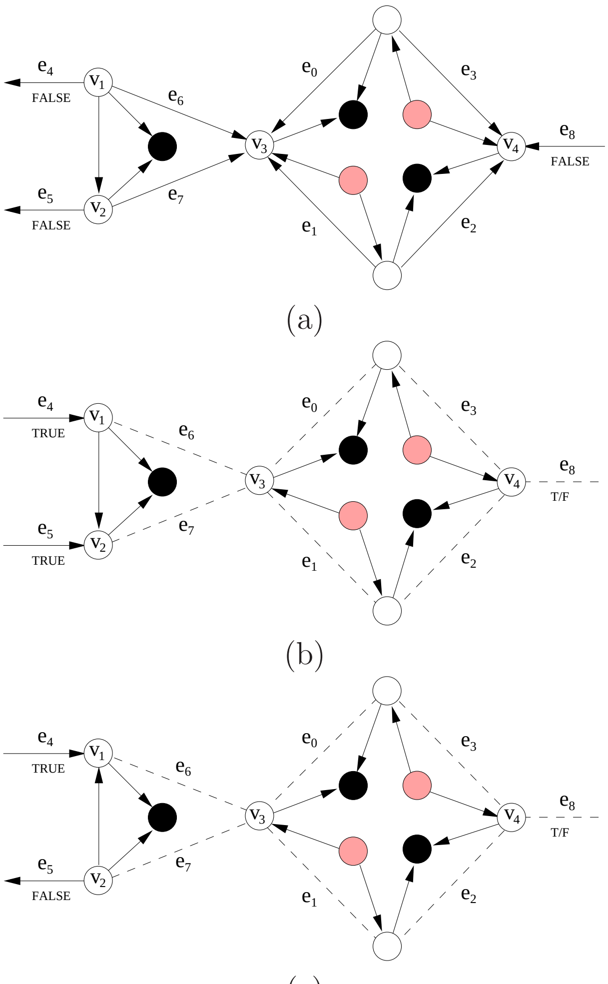


Figure 7: (a) The orientation of the 2SAT-gadget when edges e_4 and e_5 are exiting the gadget. Both the orientations are possible for edge e_8 when edges e_4 and e_5 are oriented entering the gadget (b) or one is entering and one is exiting (c).

Otherwise, suppose that e_4 and e_5 are directed both exiting v_1 and v_2 (see Fig. 7(b)). Whatever is the orientation of edge (v_1, v_2), edges e_6 and e_7 may be oriented both entering or exiting v_3 and, correspondingly, e_8 may be oriented exiting or entering v_4.

Finally, suppose that e_4 and e_5 are directed one exiting and one entering v_1 and v_2 (see Fig. 7(c)). If edge (v_1, v_2) is directed towards v_1, both the orientation for e_6 and e_7 are compatible with a bimodal orientation of the 2SAT-gadget and, again, edge e_8 may have both the orientations. □

The clause gadget for MPGB is depicted in Fig. 8 and is composed by two 3SAT-gadgets connected together. Observe that edge e_{TRUE} is always directed exiting the gadget.

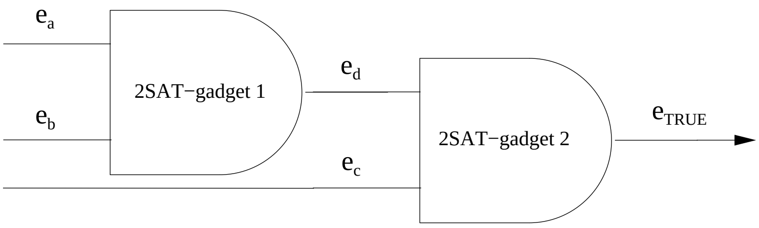


Figure 8: The clause gadget for MPGB.

The following lemma trivially descend from Lemma 5.

Lemma 6 *In any bimodal orientation of the undirected edges of the clause gadget of Fig. 8, at least one among the edges* e_a*,* e_b*, and* e_c *is directed entering the clause gadget.*

Lemma 6 gives us the basic tool for constructing an instance of Mixed Planar Graph Bimodality starting from an instance of P3SAT. As in the construction described in Section 2, we introduce a variable gadget for each variable and a clause gadget for each clause $C_j = l_1 \vee l_2 \vee l_3$. Let x_i be the variable of literal l_i, with $i = 1, 2, 3$. We suitably connect the variable gadget with one of the three edges e_a, e_b, or e_c of the clause gadget corresponding to C_j in such a way that if the truth value of the variable gadget makes l_i true then the corresponding edge enters the clause gadget. Also, since the embedding can be constrained as described in Section 3 we have the following theorem.

Theorem 5 *The* Mixed Planar Graph Bimodality *problem is NP-hard both in the fixed and in the variable embedding settings.*

5 Conclusions and Open Problems

We proved that orienting the undirected edges of a mixed plane graph such that the resulting orientation is acyclic and bimodal (or simply bimodal) is an NP-complete problem. The two problems are NP-complete even in the variable embedding setting where the embedding of the graph can be changed. We recall that both the problems are polynomial-time solvable if the input mixed graph $G = (V, E_u, E_d)$ is undirected (i.e., a mixed graph with $E_d = \emptyset$). Also, the property of being acyclic and bimodal is easy to chech on a directed plane graph (i.e., a mixed graph with $E_u = \emptyset$).

A problem that remains open is determining the complexity of orienting the undirected edges of a mixed plane (or planar) graph such that the resulting directed graph is upward planar. In fact, the constraints imposed by an upward embedding [2, 5, 6, 3] can not be directly enforced on the constructions described in Sections 2 and 4.

Acknowledgments

I am in debt with Carla Binucci, Walter Didimo, and Jan Kratochvíl for introducing this problem to me. I expecially thank Carla Binucci and Walter Didimo for interesting conversations and for encouraging this research.

References

[1] Patrizio Angelini, Giuseppe Di Battista, Fabrizio Frati, Vit Jelinek, Jan Kratochvil, Maurizio Patrignani, and Ignaz Rutter. Testing planarity of partially embedded graphs. In *Symposium On Discrete Algorithms (SODA '10)*, pages 202–221, 2010.

[2] Paola Bertolazzi, Giuseppe Di Battista, Giuseppe Liotta, and Carlo Mannino. Upward drawings of triconnected digraphs. *Algorithmica*, 12(6):476–497, 1994.

[3] Carla Binucci and Walter Didimo. Upward planarity testing of embedded mixed graphs. In *GD '11*, LNCS, 2011. To appear.

[4] Fan R. K. Chung, Michael R. Garey, and Robert E. Tarjan. Strongly connected orientations of mixed multigraphs. *Networks*, 15(4):477–484, 1985.

[5] W. Didimo, F. Giordano, and G. Liotta. Upward spirality and upward planarity testing. *SIAM J. Discrete Math.*, 23(4):1842–1899, 2009.

[6] Walter Didimo and Maurizio Pizzonia. Upward embeddings and orientations of undirected planar graphs. *J. Graph Algorithms Appl.*, 7(2):221–241, 2003.

[7] Markus Eiglsperger, Michael Kaufmann, and Frank Eppinger. An approach for mixed upward planarization. *J. Graph Algorithms Appl.*, 7(2):203–220, 2003.

[8] Jiří Fiala. NP-completeness of the edge precoloring extension problem on bipartite graphs. *J. Graph Theory*, 43(2):156–160, 2003.

[9] Ashim Garg and Roberto Tamassia. Upward planarity testing. *Order*, 12:109–133, 1995.

[10] Ashim Garg and Roberto Tamassia. On the computational complexity of upward and rectilinear planarity testing. *SIAM J. Comput.*, 31(2):601–625, 2001.

[11] Jan Kratochvíl. Personal communication, 2011.

[12] Jan Kratochvíl and András Sebö. Coloring precolored perfect graphs. *J. Graph Theory*, 25:207–215, 1997.

[13] David Lichtenstein. Planar formulae and their uses. *SIAM J. Comput.*, 11:185–225, 1982.

[14] Anthony Mansfield. Determining the thickness of graphs is NP-hard. *Proc. Math. Cambridge Philos. Soc.*, 93:9–23, 1983.

[15] Maurizio Patrignani. On extending a partial straight-line drawing. *International Journal of Foundations of Computer Science (IJFCS), Special Issue on Graph Drawing*, 17(5):1061–1069, 2006.

[16] S. Raghavan. A note on Eswaran and Tarjan's algorithm for the strong connectivity augmentation problem. In Bruce Golden, S. Raghavan, and Edward Wasil, editors, *The Next Wave in Computing, Optimization, and Decision Technologies, vol. 29 of Oberations Research*, pages 19–26. 2005.