Splitting Clusters to Get C-Planarity*

Patrizio Angelini, Fabrizio Frati, and Maurizio Patrignani

{angelini,frati,patrigna}@dia.uniroma3.it

Abstract. In this paper we introduce a generalization of the c-planarity testing problem for clustered graphs. Namely, given a clustered graph, the goal of the SPLIT-C-PLANARITY problem is to split as few clusters as possible in order to make the graph c-planar. Determining whether zero splits are enough coincides with testing c-planarity. We show that SPLIT-C-PLANARITY is NP-complete for c-connected clustered triangulations and for non-c-connected clustered paths and cycles. On the other hand, we present a polynomial-time algorithm for flat c-connected clustered graphs whose underlying graph is a biconnected series-parallel graph, both in the fixed and in the variable embedding setting, when the splits are assumed to maintain the c-connectivity of the clusters.

1 Introduction

Let $C(G,T)$ be a clustered graph and suppose that a c-planar drawing of C is impossible (or very difficult) to find. A natural question is whether C admits a drawing where each cluster is represented by a small set of connected regions instead of a single connected region of the plane. We formalize this concept by introducing the *split* operation, that replaces a cluster μ of T with two clusters μ_1 and μ_2 with the same parent as μ, and distributes the children of μ between μ_1 and μ_2. We search for the minimum number of splits turning C into a c-planar clustered graph. Formally, the corresponding decision problem is as follows:

Problem: SPLIT-C-PLANARITY
Instance: A clustered graph $C = (G,T)$ and an integer $k \geq 0$.
Question: Can $C(G,T)$ be turned into a c-planar clustered graph $C(G,T')$ by performing at most k split operations?

SPLIT-C-PLANARITY is motivated not only by the practical need of drawing non-c-planar clustered graphs, but also by its implications on the c-planarity theory. In fact, the long-standing problem of testing c-planarity [8] is a particular case of SPLIT-C-PLANARITY, where zero splits are allowed. Therefore, SPLIT-C-PLANARITY extends the c-planarity testing problem to a more general setting, where we are able to show the NP-hardness even for flat clustered graphs whose underlying graphs are paths or cycles.

Hence, following a strategy that is analogous to the one used in the literature for the c-planarity testing problem, we focus on peculiar classes of clustered graphs.

* This work is partially supported by the Italian Ministry of Research, Grant number RBIP06BZW8, FIRB project "Advanced tracking system in intermodal freight transportation".

D. Eppstein and E.R. Gansner (Eds.): GD 2009, LNCS 5849, pp. 57–68, 2010.

Table 1. Time complexity of SPLIT-C-PLANARITY for non-c-connected graphs

Graph Family	Fixed Embedding Setting	Variable Embedding Setting
Paths, cycles, trees, & outerplanar graphs	NP-hard (Th. 5)	NP-hard (Th. 5)
Series-parallel graphs	NP-hard (Th. 5)	NP-hard (Th. 4)
General graphs	NP-hard (Th. 1)	NP-hard (Th. 1)

Table 2. Time complexity of SPLIT-C-PLANARITY for c-connected graphs

Graph Family	Fixed Embedding Setting	Variable Embedding Setting
Paths, cycles, & trees	$\Theta(1)$ (trivial)	$\Theta(1)$ (trivial)
Outerplanar graphs	?	$\Theta(1)$ (trivial)
Series-parallel graphs*	Polynomial (Th. 2)	Polynomial (Th. 3)
Series-parallel graphs	?	?
General graphs	NP-hard (Th. 1)	NP-hard (Th. 1)

* Flat hierarchy, biconnected underlying graph, c-connectivity preserved.

Restrictions on the c-planarity testing problem that have been considered in the literature include: (i) assuming that each cluster induces a small number of connected components [8,4,11,10,1,2,12] (in particular, the case in which the graph is *c-connected*, that is, each cluster induces one connected component, has been deeply investigated); (ii) considering only *flat* hierarchies, where all clusters different from the root of T are children of the root [3,6]; (iii) focusing on particular families of underlying graphs [3,13]; and (iv) fixing the embedding of the underlying graph [6,12].

We show that SPLIT-C-PLANARITY is NP-hard even for flat c-connected clustered graphs whose underlying graph is triconnected (hence even for flat c-connected embedded clustered graphs). On the other hand, we show that SPLIT-C-PLANARITY is polynomial-time solvable for flat c-connected clustered graphs whose underlying graph is a biconnected series-parallel graph (both if the underlying graph has fixed or variable embedding) if the splits are assumed to preserve the c-connectivity of the graph.

Tables 1 and 2 summarize the time complexity of SPLIT-C-PLANARITY. Observe that, being acyclic, every c-connected clustered tree is trivially c-planar. Also, in an outerplanar embedding of any outerplanar graph no cycle contains a vertex in its interior. Therefore, every c-connected clustered outerplanar graph is c-planar.

The rest of the paper is organized as follows. In Sect. 2 we introduce some preliminaries; in Sect. 3 we prove the NP-hardness of SPLIT-C-PLANARITY for flat c-connected clustered triangulations; in Sect. 4 we show a polynomial-time algorithm for SPLIT-C-PLANARITY on flat c-connected biconnected clustered series-parallel graphs; in Sect. 5 we show the NP-hardness of SPLIT-C-PLANARITY for flat non-c-connected clustered paths and cycles; in Sect. 6 we conclude and present some open problems.

2 Background

We refer to [5] for basic definitions about graphs and embeddings, and to [8,4,11,3,10,1,6,2,13,12] for basic definitions about clustered graphs and c-planar drawings.

A *series-parallel graph* is inductively defined as follows. An edge (u,v) is a series-parallel graph with *poles* u and v. Denote by u_i and v_i the poles of a series-parallel graph G_i. A *series composition* of a sequence $G_1, \ldots, G_k$ of series-parallel graphs, with $k \geq 2$, is a series-parallel graph with poles $u = u_1$ and $v = v_k$ such that v_i and u_{i+1} have been identified, for each $i = 1, \ldots, k-1$. A *parallel composition* of a set $G_1, \ldots, G_k$ of series-parallel graphs, with $k \geq 2$, is a series-parallel graph with poles $u = u_1 = \cdots = u_k$ and $v = v_1 = \cdots = v_k$. The SPQ-tree of a series-parallel graph G is the tree representing the series and parallel compositions of G. Let G be a series-parallel graph with poles u and v and with a fixed plane embedding $\mathcal{E}_o$. The *leftmost path* (resp. *rightmost path*) of G is the path $(w_1 = u, w_2, \ldots, w_k = v)$ (resp. $(z_1 = u, z_2, \ldots, z_h = v)$) such that: (i) w_2 follows w_1 (resp. z_2 precedes z_1) in the counter-clockwise order of the vertices incident to the outer face of $\mathcal{E}_o$; (ii) edge (w_i, w_{i+1}) follows (w_{i-1}, w_i) (resp. (z_i, z_{i+1}) precedes (z_{i-1}, z_i)) in the counter-clockwise order of the edges incident to w_i (resp. incident to z_i). The leftmost and rightmost paths of G are also called *extreme paths* of G.

3 General C-Connected Clustered Graphs

We show the NP-hardness of SPLIT-C-PLANARITY for flat c-connected clustered graphs whose underlying graph is triconnected. This is done by means of a reduction from HAMILTONIAN-CIRCUIT [9], which takes as an input a triconnected, planar, and cubic graph $G(V, E)$ and asks whether a simple cycle exists in G traversing each node $v \in V$ exactly once. Given an instance of HAMILTONIAN-CIRCUIT, consider a planar drawing of it and the dual graph G' of G (see Fig. 1(a)). Observe that, since G is cubic, G' is a triangulation. Construct an instance $\langle C(G'', T), k\rangle$ of SPLIT-C-PLANARITY as follows. Graph G'' is obtained by adding to G' a node v_i in each face f_i and by connecting v_i to the three vertices incident to f_i (see Fig. 1(b)). Tree T has height two and has a cluster μ_i for each added vertex v_i and a cluster μ_0 containing all the vertices of G'. The value of k is set to one. We make use of the following result appeared in [7]:

Lemma 1. *(Feng [7]) Let $C(G, T)$ be a clustered graph where G is a triangulation. Then C is c-planar only if C is c-connected.*

Lemma 2. *Instance G of* HAMILTONIAN-CIRCUIT *admits a solution if and only if the corresponding instance $\langle C(G'', T), 1\rangle$ of* SPLIT-C-PLANARITY *does.*

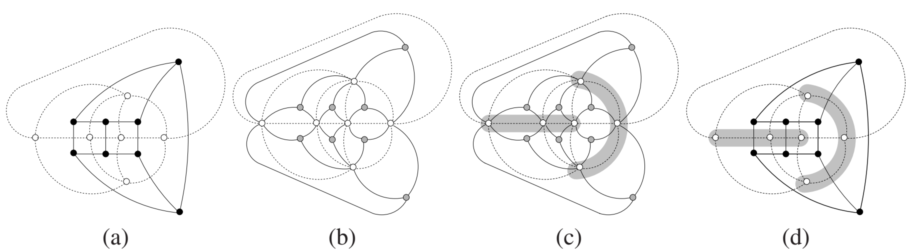

Fig. 1. (a) A planar graph G (black vertices) and its dual graph G' (white vertices). (b) Graph G'' (the vertices added to G' are drawn gray). (c) A split of cluster μ_0 turning G'' into a c-planar clustered graph. (d) The corresponding Hamiltonian circuit on G (thick edges).

Proof: Suppose $\langle C(G'', T), 1\rangle$ admits a solution. Since G'' is triconnected, in any planar drawing of G'' each vertex v_i inserted into an internal face f_i of G' is inside a cycle of vertices belonging to cluster μ_0. Hence, $C(G'', T)$ is not c-planar, and at least one split of cluster μ_0 has to be performed in order to turn $C(G'', T)$ into a c-planar graph. Suppose that a split of cluster μ_0 into two clusters μ_a and μ_b exists such that the obtained clustered graph $C(G'', T')$ is c-planar (see Fig. 1(c)). The split is a bipartition of the vertices of G' into V_a and V_b. By Lemma 1, the two graphs induced by V_a and V_b are connected. Hence, the edges between V_a and V_b form a cutset. A cutset in G' corresponds to a cycle $\mathcal{C}$ in G [14, pg. 16]. Since $C(G'', T')$ is c-planar, each vertex v_i inserted into a face f_i of G' is adjacent both to a vertex in μ_a and to a vertex in μ_b. This is equivalent to saying that $\mathcal{C}$ traverses each vertex of G exactly once (see Fig. 1(d)).

Suppose that a Hamiltonian circuit $\mathcal{C}$ exists in G. Split μ_0 so that nodes internal to $\mathcal{C}$ belong to μ_a and nodes external to $\mathcal{C}$ belong to μ_b. The obtained graph $C(G'', T')$ is c-planar. In fact, $\mathcal{C}$ determines a cutset in G', hence μ_a and μ_b induce connected graphs. Further, since $\mathcal{C}$ is Hamiltonian, the graphs induced by μ_a and μ_b are acyclic. □

Since $\langle C(G'', T), 1\rangle$ can be constructed in polynomial time and since the problem is easily seen to be in NP, the following holds.

Theorem 1. SPLIT-C-PLANARITY *is NP-complete when the input graph is a flat c-connected clustered graph and* $k = 1$.

4 Series-Parallel C-Connected Clustered Graphs

In this section, we show that SPLIT-C-PLANARITY is polynomial-time solvable if: (i) the input graph is a flat c-connected clustered graph whose underlying graph is a biconnected series-parallel graph, and (ii) the splits have to maintain the c-connectivity of the input graph. Observe that the reduction shown in Sect. 3 proves that SPLIT-C-PLANARITY is NP-complete if: (i) the input graph is a flat c-connected clustered graph, and (ii) the splits have to maintain the c-connectivity of the input graph (namely, such a condition is always met when splitting clusters of a clustered triangulation). Throughout this section, we assume that every set of splits turning a c-connected clustered graph into a c-planar clustered graph maintains the c-connectivity of the graph.

4.1 Series-Parallel Graphs with Fixed Embedding

We show a polynomial-time algorithm that, given a flat c-connected clustered graph $C(G, T)$, where G is a biconnected series-parallel graph with fixed planar embedding $\mathcal{E}$, computes the minimum number of splits turning C into a c-planar clustered graph. The algorithm performs a bottom-up visit of the SPQ-tree $\mathcal{T}$ of G, rooted at any P-node corresponding to a parallel composition of two series-parallel graphs B_1 and B_2, where B_1 is an edge e and B_2 is the rest of the graph. Topologically, such a choice corresponds to assuming that e is on the outer face of a plane embedding $\mathcal{E}_o$ corresponding to the planar embedding $\mathcal{E}$. However, there are $O(n)$ ways of making such a choice, hence the test is repeated a linear number of times. Throughout this subsection, we assume

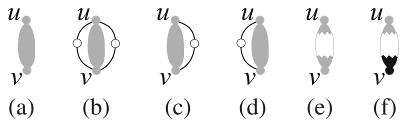

Fig. 2. Representation of a node t of $\mathcal{T}$ satisfying (a) Condition A, (b) Condition B, (c) Condition C, (d) Condition D, (e) Condition E, and (f) Condition F

that $\mathcal{E}$ is fixed and that e is on the outer face of $\mathcal{E}_o$. We denote by $\mu(u)$ the only cluster different from the root of T containing vertex u.

For each node t of $\mathcal{T}$ corresponding to a series-parallel graph B with poles u and v, the algorithm computes six labels $\alpha(t), \beta(t), \gamma(t), \delta(t), \epsilon(t)$, and $\phi(t)$. Such labels represent the minimum number of splits on C turning $(B, T'[B])$ (that is, the clustered graph whose cluster hierarchy is the tree obtained from T by performing the splits on C and by restricting to the clusters containing vertices of B) into a c-planar clustered graph satisfying, respectively, the following conditions (see Fig. 2):

- *Condition A:* all the vertices of B belong to $\mu(u) = \mu(v)$;
- *Condition B:* $\mu(u) = \mu(v)$, there exists a path between u and v whose vertices all belong to $\mu(u)$, and $p_r(B)$ and $p_l(B)$ contain vertices not belonging to $\mu(u)$;
- *Condition C:* $\mu(u) = \mu(v)$, there exists a path between u and v whose vertices all belong to $\mu(u)$, $p_r(B)$ contains vertices not belonging to $\mu(u)$, and all the vertices of $p_l(B)$ belong to $\mu(u)$;
- *Condition D:* $\mu(u) = \mu(v)$, there exists a path between u and v whose vertices all belong to $\mu(u)$, all the vertices of $p_r(B)$ belong to $\mu(u)$, and $p_l(B)$ contains vertices not belonging to $\mu(u)$;
- *Condition E:* $\mu(u) = \mu(v)$ and there exists no path between u and v whose vertices all belong to $\mu(u)$; and,
- *Condition F:* $\mu(u) \neq \mu(v)$.

When $(B, T'[B])$ satisfies a certain condition, we equivalently say that t satisfies the same condition. In general, it could be not possible to make t satisfy a certain condition with any set of splits. For example, if $\mu(u) \neq \mu(v)$, no set of splits makes u and v belong to the same cluster, hence labels $\alpha(t), \beta(t), \gamma(t), \delta(t)$, and $\epsilon(t)$ have no meaning for t. In such cases, we set the corresponding labels to ∞.

We observe the following lemmata:

Lemma 3. *Consider any set of splits turning $C(G, T)$ into a c-planar clustered graph $C'(G, T')$. Then, $(B, T'[B])$ satisfies exactly one of Conditions A, B, C, D, E, and F.*

Lemma 4. *If $(B, T'[B])$ satisfies Condition A, B, C, D, or F, then $(B, T'[B])$ is a c-connected clustered graph. Also, if $(B, T'[B])$ satisfies Condition E, then each cluster in $T'[B]$ induces one connected component in B, except for $\mu(u)$, which induces two connected components, one containing u, and the other containing v.*

We now sketch how to compute $\alpha(t), \beta(t), \gamma(t), \delta(t), \epsilon(t)$, and $\phi(t)$. In the base case, t is an edge (u, v) and the six labels can be easily computed. Namely, if u and v belong

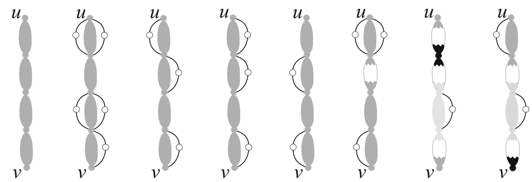

Fig. 3. Constraints on the children of t, if t is an S-node satisfying Condition x. If $x = A$, then all the t_i satisfy Condition A (a). If $x = B$, then either there exists t_i satisfying Condition B and all other t_j satisfy Condition A, B, C, or D (b), or there exist t_i satisfying Condition C, t_j satisfying Condition D, and all other t_l satisfy Condition A, C, or D (c). If $x = C$, then there exists t_i satisfying Condition C and all other t_j satisfy Condition A or C (d). If $x = D$, then there exists t_i satisfying Condition D and all other t_j satisfy Condition A or D (e). If $x = E$, then u and v belong to the same cluster in the input clustered graph and either there exists t_i satisfying Condition E and all other t_j satisfy Condition A, B, C, or D (f), or there exist t_i and t_j satisfying Condition F and all other t_l satisfy Condition A, B, C, D, or F (g). If $x = F$, then there exists t_i satisfying Condition F and all other t_j satisfy Condition A, B, C, D, or F (h).

to distinct clusters, then $\alpha(t) = \beta(t) = \gamma(t) = \delta(t) = \phi(t) = \infty$, and $\epsilon(t) = 0$. If u and v belong to the same cluster, then $\alpha(t) = 0$, $\beta(t) = \gamma(t) = \delta(t) = \epsilon(t) = \infty$, and $\phi(t) = 1$.

Consider a node t of $\mathcal{T}$ corresponding to a series-parallel graph B. Let $t_1, \dots, t_k$ be the children of t, corresponding to series-parallel graphs $B_1, \dots, B_k$. Let u_i and v_i be the poles of B_i. Inductively suppose that the labels of $t_1, \dots, t_k$ have been computed.

The main idea is that if a set S of splits makes $(B, T'[B])$ satisfy Condition A, B, C, D, E, or F, then several constraints on the conditions that are satisfied by the children of t can be deduced, also based on whether t is an S-node or a P-node.

As an example, if t is a P-node satisfying Condition C, then either t_i exists satisfying Condition C or not. If such a t_i exists, then all the t_j with $j < i$ satisfy Condition A and all the t_j with $j > i$ satisfy Condition E; namely, if any t_j with $j < i$ satisfies Condition B or C, then $(B, T'[B])$ is not c-planar, as it contains a cycle, whose vertices belong to the same cluster, enclosing a vertex not belonging to such a cluster; if any t_j with $j < i$ satisfies Condition D or E, then either $(B, T'[B])$ is not c-planar or t does not satisfy Condition C; no t_j satisfies Condition F because $\mu(u) = \mu(v)$; finally, if any t_j with $j > i$ satisfies Condition A, B, C, or D, then $(B, T'[B])$ is not c-planar. If no t_i satisfies Condition C, then a sequence of consecutive t_j, including t_1, satisfy Condition A, and all other t_j, including t_k, satisfy Condition E. See Figs. 3 and 4.

As a result of the above argumentations, a set of k-tuples is associated to Condition x, where $x \in \{A, B, C, D, E, F\}$, for each node t of $\mathcal{T}$ with k children. Each tuple is such that if t_i satisfies the condition indicated at the i-th item of the tuple, for each i, then t satisfies Condition x. Then, the minimum number of splits turning $(B, T'[B])$ into a c-planar clustered graph satisfying Condition x is the minimum among the values associated with the tuples, where the value associated with each tuple is obtained by summing up the labels corresponding to the conditions of the tuple, paying attention to those splits counted more than once in different nodes t_i. We get the following:

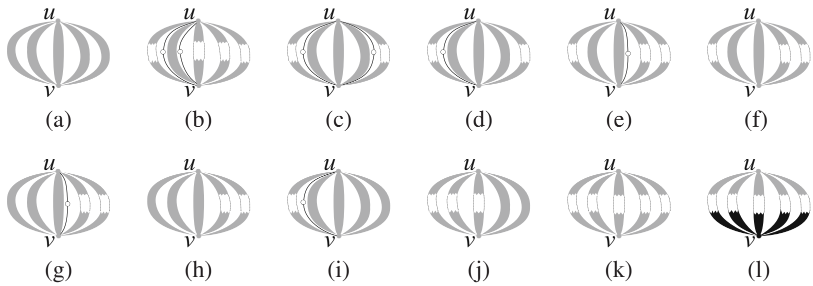

Fig. 4. Constraints on the children of t, if t is a P-node satisfying Condition x. If $x = A$, then all the t_i satisfy Condition A (a). If $x = B$, then either there exists t_i satisfying Condition B and all other t_j satisfy Condition E (b), or there exists t_i satisfying Condition D, t_j satisfying Condition C, with $j > i$, and all the t_l satisfy Condition E, if $l < i$ and $l > j$, or Condition A, if $i < l < j$ (c), or there exists t_i satisfying Condition D, all the t_l satisfy Condition E, if $l < i$ and if $l > y$, for some $i \leq y < k$, and all the t_l satisfy Condition A, if $i < l \leq y$ (d), or there exists t_i satisfying Condition C, all the t_l satisfy Condition E, if $l > i$ and if $l < x$, for some $1 < x \leq i$, and all the t_l satisfy Condition A, if $x \leq l < i$ (e), or all the t_l satisfy Condition E, if $l < x$ and if $l > y$, for some $1 < x \leq y < k$, and all the t_l satisfy Condition A, if $x \leq l \leq y$ (f). If $x = C$, then either there exists t_i satisfying Condition C, all the t_j with $j > i$ satisfy Condition E, and all the t_j with $j < i$ satisfy Condition A (g), or all the t_j satisfy Condition A, with $1 \leq j \leq y$ for some $1 \leq y < k$, and all the t_j satisfy Condition E, with $j > y$ (h). If $x = D$, then either there exists t_i satisfying Condition D, all the t_j with $j < i$ satisfy Condition E, and all the t_j with $j > i$ satisfy Condition A (i), or all the t_j satisfy Condition A, with $x \leq j \leq k$ for some $1 < x \leq k$, and all the t_j satisfy Condition E, with $j < x$ (j). If $x = E$, then all the t_i satisfy Condition E (k). If $x = F$, then all the t_i satisfy Condition F (l).

Theorem 2. *Let $C(G, T)$ be a flat c-connected clustered graph whose underlying graph G is an n-vertex biconnected series-parallel graph with a fixed planar embedding $\mathcal{E}$. The minimum number of splits turning C into a c-planar clustered graph while maintaining the c-connectivity of every cluster can be computed in $O(n^4)$ time.*

4.2 Series-Parallel Graphs with Variable Embedding

We sketch how to extend the result of Sect. 4.1 to the variable embedding scenario.

As in the fixed embedding case, we perform a bottom-up visit of the rooted SPQ-tree $\mathcal{T}$ of G, while computing some labels for each node t of $\mathcal{T}$. However, in this case, we have to determine some embeddings of the series-parallel graph B corresponding to t. For each node t of $\mathcal{T}$, we compute *five* labels $\alpha(t), \beta(t), \gamma\delta(t), \epsilon(t)$, and $\phi(t)$. Labels $\alpha(t), \epsilon(t)$, and $\phi(t)$ have the same meaning as in the fixed embedding case. Label $\beta(t)$ represents the minimum number of splits turning $(B, T[B])$ into a c-planar clustered graph $(B, T'[B])$ containing a path between u and v whose vertices all belong to $\mu(u)$, and having vertices not belonging to $\mu(u)$ on both extreme paths of a computed planar embedding. Label $\gamma\delta(t)$ represents the minimum number of splits turning $(B, T[B])$ into a c-planar clustered graph $(B, T'[B])$ containing a path between u and v whose vertices all belong to $\mu(u)$ and having vertices not belonging to $\mu(u)$ on exactly one extreme path of a computed planar embedding. Observe that labels $\beta(t)$ and $\gamma\delta(t)$ replace

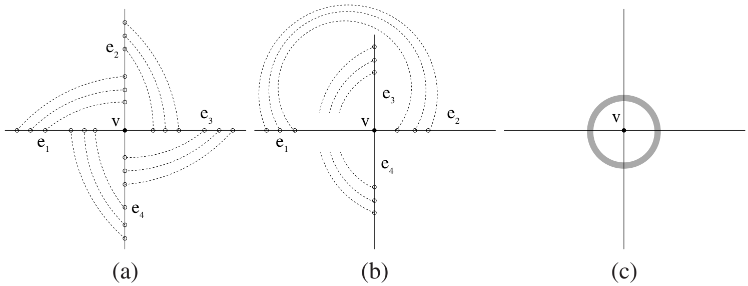

Fig. 5. (a) A pinwheel gadget of size three. Dashed lines join vertices of the same cluster. (b) An illustration for the proof of Lemma 5. (c) A symbolic representation of the pinwheel gadget.

labels $\beta(t)$, $\gamma(t)$, and $\delta(t)$ of the fixed embedding scenario, as in the variable embedding setting it is not known *a priori* which are the rightmost and the leftmost path of t.

Theorem 3. *Let $C(G,T)$ be a flat c-connected clustered graph whose underlying graph G is an n-vertex biconnected series-parallel graph. The minimum number of splits turning C into a c-planar clustered graph while maintaining the c-connectivity of every cluster at each split can be computed in $O(n^4)$ time.*

5 Non-C-Connected Clustered Graphs

We open this section by showing that, given a flat non-c-connected clustered graph $C(G,T)$, where G is a biconnected series-parallel graph, it is NP-hard to find the minimum number of splits turning C into a c-planar clustered graph. Namely, we perform a reduction from NAE3SAT [9], which takes in input a collection of clauses, each consisting of three literals, and asks whether a truth assignment to the variables exists such that each clause has at least one true literal and at least one false literal.

Given a clustered graph $C(G,T)$ and a vertex v of G with four incident edges e_1, e_2, e_3, and e_4, we introduce a gadget that forces such edges to appear in this circular order around v in any c-planar drawing of any clustered graph obtained from C with less than σ splits. We construct around v a *pinwheel gadget* of size σ by inserting, in each edge e_i, 2σ vertices $v_{i,j}$, with $j = 1, \dots, 2\sigma$. For each pair (e_i, e_{i+1}) we add σ child-clusters to the root of T and assign $v_{i,j}$ and $v_{i+1,j+\sigma}$ to the same cluster, for $j = 1, \dots, \sigma$. Figure 5 provides an example for $\sigma = 3$.

Lemma 5. *Let $C(G,T)$ be a clustered graph containing a pinwheel gadget of size σ around a vertex v. Any c-planar drawing of a clustered graph obtained from C with less than σ splits preserves the circular order of the edges around v, up to a reversal.*

Proof: Suppose, for a contradiction, that there exists a c-planar drawing of a clustered graph obtained from C with less than σ splits such that the order of the edges around v is e_1, e_3, e_2, and e_4, the other cases being analogous. Consider the σ clusters involving vertices of both e_1 and e_2. Since less than σ splits are allowed, at least one of such

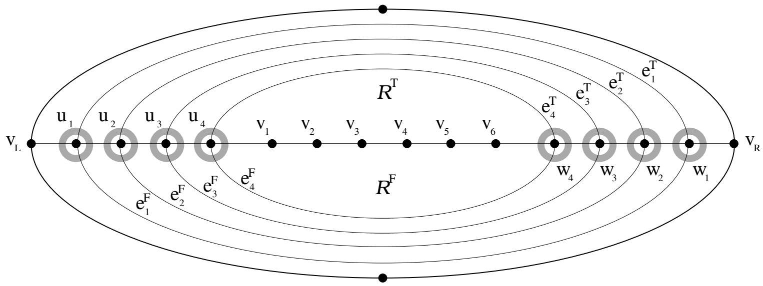

Fig. 6. An illustration for the construction of instance $\langle C(G_\varphi, T_\varphi), k_\varphi\rangle$ of SPLIT-C-PLANARITY corresponding to an instance φ of NAE3SAT with four variables and six clauses

clusters is not split. Hence, the region of the plane delimited by the border of such a cluster, by e_1, and by e_2 either encloses vertices $v_{3,j}$, with $j = 1, \ldots, 2\sigma$, and does not enclose vertices $v_{4,j}$, with $j = 1, \ldots, 2\sigma$, or vice versa. It follows that all the σ clusters involving vertices of both e_3 and e_4 are split, contradicting the hypothesis. □

Given an instance φ of NAE3SAT with n variables and c clauses we construct the corresponding instance $\langle C_\varphi(G_\varphi, T_\varphi), 2c\rangle$ of SPLIT-C-PLANARITY as follows. Graph G_φ contains a cycle $\mathcal{C}$ with two notable vertices v_L and v_R (see Fig. 6), and a path $(v_L, u_1, u_2, \ldots, u_n, v_1, v_2, \ldots, v_c, w_n, w_{n-1}, \ldots, w_1, v_R)$. Observe that, in any planar embedding of G_φ, such a path, together with $\mathcal{C}$, determines two regions (both inside or both outside $\mathcal{C}$) that we arbitrarily denote by $\mathcal{R}^T$ and $\mathcal{R}^F$. G_φ also contains two edges $e_i^T = (u_i, w_i)$ and $e_i^F = (u_i, w_i)$, for each $i = 1, \ldots, n$. Denote $u_0 = v_L$, $u_{n+1} = v_1$, $w_0 = v_R$, and $w_{n+1} = v_c$. For $i = 1, \ldots, n$, two pinwheel gadgets of size $2c + 1$ are inserted around u_i and w_i so that the circular order of the edges around u_i and w_i is (u_{i-1}, u_i), e_i^T, (u_i, u_{i+1}), e_i^F, and (w_{i-1}, w_i), e_i^F, (w_i, w_{i+1}), e_i^T, respectively. Figure 6 shows an example with $n = 4$ and $c = 6$. The insertion of the pinwheel gadgets turns e_i^T and e_i^F into two paths, that we denote by p_i^T and p_i^F, respectively. Observe that, by Lemma 5, in any c-planar embedding of a clustered graph obtained from C_φ with less than $2c + 1$ splits, if p_i^T lies into $\mathcal{R}^T$ ($\mathcal{R}^F$), then p_i^F lies into $\mathcal{R}^F$ ($\mathcal{R}^T$).

For each clause j, we introduce two clusters $\nu_{j,1}$ and $\nu_{j,2}$. Also, we define two *literal gadgets* $l^{\notin}(j)$ and $l^{\in}(j)$ as follows. Gadget $l^{\notin}(j)$ is a sequence of three vertices v_a, v_b, and v_c belonging to clusters $\nu_{j,1}$, $\nu_{j,2}$, and $\nu_{j,1}$, respectively (see variable x_1 of Fig. 7(a)). Gadget $l^{\in}(j)$ contains a sequence of four vertices v_d, v_e, v_f, and v_g, plus two additional vertices v_h and v_i attached to both v_d and v_e. While v_d and v_e are assigned to the root of T_φ, v_f belongs to $\nu_{j,2}$ and v_g, v_h, and v_i belong to $\nu_{j,1}$. Finally, two pinwheel gadgets of size $2c + 1$ are inserted around v_d and v_e so that, in any c-planar drawing of a clustered graph obtained from C_φ with less than $2c + 1$ splits, v_h and v_i are on opposite sides with respect to edge (v_d, v_e) (see variable x_2 of Fig. 7(a)).

For each variable x_i, with $i = 1, \ldots, n$, and for each clause c_j, with $j = 1, \ldots, c$, we insert into p_i^T (p_i^F) gadget $l^{\in}(j)$ if x_i ($\overline{x}_i$, respectively) is a literal of c_j and gadget $l^{\notin}(j)$ otherwise, in such a way that the gadgets for clauses $c_1, c_2, \ldots, c_c$ appear in this order from u_i to w_i in p_i^T and p_i^F.

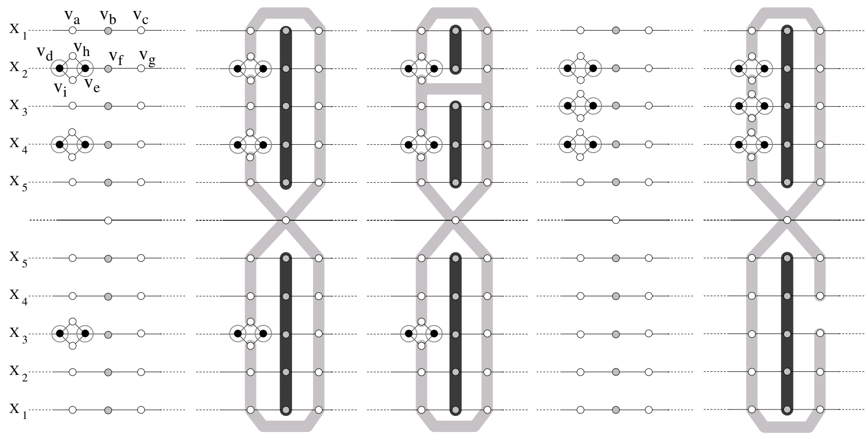

Fig. 7. (a) Configuration of a clause $(x_2 \vee \overline{x}_3 \vee x_4)$. (b) and (c) show drawings with two split. (d) A configuration of a clause with all true literals. Any c-planar drawing of it needs three split (e).

We assign to the root of T_φ vertices v_L, v_R, u_i and w_i, with $i = 1, \ldots, n$. Vertex v_j, for $j = 1, \ldots, c$, is assigned to $\nu_{j,1}$.

Lemma 6. *Instance* φ *of* NAE3SAT*, with* n *variables and* c *clauses, admits a solution if and only if instance* $\langle C_\varphi(G_\varphi, T_\varphi), 2c\rangle$ *of* SPLIT-C-PLANARITY *admits a solution.*

Proof sketch: Suppose φ admits a solution and consider an assignment of truth values to the variables that satisfies φ. If variable x_i is TRUE (FALSE), then draw p_i^T into $\mathcal{R}^T$ ($\mathcal{R}^F$) and p_i^F into $\mathcal{R}^F$ ($\mathcal{R}^T$). Observe that, for each clause c_j no three $l^\in(j)$ are in the same region. Figure 7(b) shows a portion of a c-planar drawing of a clustered graph obtained from C_φ with two splits per clause. Hence, $\langle C_\varphi, 2c\rangle$ admits a solution.

Suppose $\langle C_\varphi, 2c\rangle$ admits a solution. In order to obtain a c-planar clustered graph from C_φ, at least two splits are needed for $\nu_{j,1}$ and $\nu_{j,2}$ as a whole (see Figs 7(b) and 7(c)); further, if the literal gadgets $l^\in(j)$ of clause c_j are all three in the same region, then at least three splits are needed for $\nu_{j,1}$ and $\nu_{j,2}$ as a whole (see Fig. 7(e)). It follows that, since only $2c$ splits turn C_φ into a c-planar graph, there exists a truth assignment such that each clause has a TRUE and a FALSE literal. □

Since $\langle C_\varphi, 2c\rangle$ can be constructed in polynomial time and since the problem is easily seen to be in NP, the following holds.

Theorem 4. SPLIT-C-PLANARITY *is NP-complete when the input is a flat non-c-connected clustered series-parallel graph.*

By modifying the above reduction, it is possible to show that SPLIT-C-PLANARITY is NP-complete even for a non c-connected clustered tree, path, or cycle.

Namely, we introduce the *open pinwheel gadget* of size σ, whose vertices have degree at most two, to replace a pinwheel gadget of size σ in the reduction from

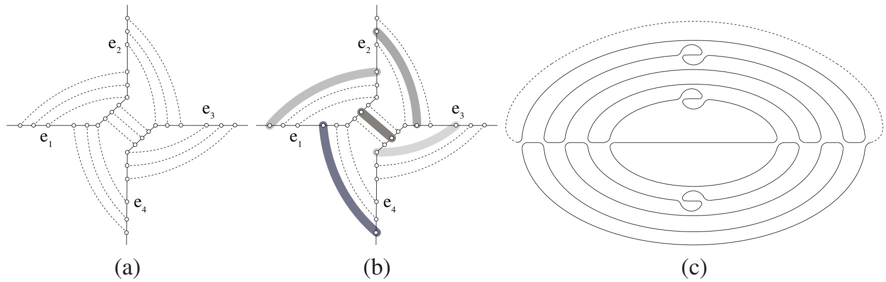

Fig. 8. (a) An open pinwheel gadget. (b) A picture for the proof of Lemma 7. (c) A picture for the proof of Theorem 5 (the literal gadgets of only one cluster are shown).

NAE3SAT. Such a gadget is obtained from a pinwheel gadget around vertex v by removing v and joining edges e_1 and e_2 and edges e_3 and e_4 (or edges e_1 and e_4 and edges e_2 and e_3) with a path of σ vertices belonging to clusters $\mu_1, \ldots, \mu_\sigma$ (see Fig. 8).

Lemma 7. *Let $C^*(G^*, T^*)$ be the clustered graph obtained from $C_\varphi(G_\varphi, T_\varphi)$ by replacing each pinwheel gadget of size σ with an open pinwheel gadget of the same size. Then, C^* can be turned into a c-planar clustered graph with less than σ splits if and only if C_φ can be turned into a c-planar clustered graph with less than σ splits.*

Theorem 5. *Problem* SPLIT-C-PLANARITY *is NP-complete when the input graph is a non-c-connected cycle or path.*

Proof sketch: Construct instance $\langle C_\varphi(G_\varphi, T_\varphi), 2c\rangle$ corresponding to the instance φ of NAE3SAT with c clauses as in the reduction used in Theorem 4. Add an edge connecting v_L with v_R and add two pinwheel gadgets around v_L and v_R. Observe that all the vertices have degree two or four and that all the vertices of degree four have a pinwheel gadget around them. Replace each pinwheel gadget with an open pinwheel gadget of the same size in such a way that the underlying graph G^* of the obtained clustered graph $C^*(G^*, T^*)$ is a cycle, as shown in Fig. 8(c). By Lemma 7, any c-planar drawing of a clustered graph obtained from C^* with less than $2c$ splits corresponds to a c-planar drawing of a clustered graph obtained from C_φ with less than $2c$ splits, and vice versa. Lemma 6 ensures that instance φ admits a solution if and only if instance $\langle C^*(G^*, T^*), 2c\rangle$ of SPLIT-C-PLANARITY admits a solution.

To prove that the problem is NP-complete also for paths it suffices to turn G^* into a path by "opening" edge (v_L, v_R) (dashed edge of Fig. 8(c)). $\square$

Theorem 5 implies that SPLIT-C-PLANARITY is NP-complete when the input is a non-c-connected tree both in the fixed and in the variable embedding setting.

6 Conclusions

In this paper we introduced the SPLIT-C-PLANARITY problem, which takes as an input a clustered graph $C(G, T)$ and an integer $k \geq 0$ and asks whether C can be turned into a c-planar clustered graph $C'(G, T')$ by performing at most k cluster splits.

We proved that SPLIT-C-PLANARITY is NP-hard, even for non-c-connected clustered paths and cycles, and for c-connected clustered triangulations. Further, SPLIT-C-PLANARITY is not fixed-parameter tractable with respect to k, as it is NP-hard even with $k = 1$. However, it could still be the case that SPLIT-C-PLANARITY is fixed-parameter tractable with respect to k, when the underlying graph of the input clustered graph is a path, a cycle, or a graph in a similarly simple graph family. Namely, the reduction we presented in the c-connected case uses a constant k, but deals with triconnected graphs, while the reduction we presented for the non-c-connected case deals with paths and cycles, but uses a k which is function of the size of the problem.

We proved that for flat clustered graphs whose underlying graph is a biconnected series-parallel graph SPLIT-C-PLANARITY is polynomial-time solvable, if the splits are assumed to maintain the c-connectivity of the clusters. We believe the following extensions of such a result to be interesting: (i) non-flat clustered graphs; (ii) simply-connected series-parallel graphs; (iii) splits not maintaining the c-connectivity.

References

1. Cornelsen, S., Wagner, D.: Completely connected clustered graphs. J. Discrete Algorithms 4(2), 313–323 (2006)
2. Cortese, P.F., Di Battista, G., Frati, F., Patrignani, M., Pizzonia, M.: C-planarity of c-connected clustered graphs. J. Graph Alg. Appl. 12(2), 225–262 (2008)
3. Cortese, P.F., Di Battista, G., Patrignani, M., Pizzonia, M.: Clustering cycles into cycles of clusters. J. Graph Alg. Appl. 9(3), 391–413 (2005)
4. Dahlhaus, E.: A linear time algorithm to recognize clustered graphs and its parallelization. In: Lucchesi, C.L., Moura, A.V. (eds.) LATIN 1998. LNCS, vol. 1380, pp. 239–248. Springer, Heidelberg (1998)
5. Di Battista, G., Eades, P., Tamassia, R., Tollis, I.G.: Graph Drawing: Algorithms for the Visualization of Graphs. Prentice-Hall, Englewood Cliffs (1999)
6. Di Battista, G., Frati, F.: Efficient c-planarity testing for embedded flat clustered graphs with small faces. In: Hong, S.-H., Nishizeki, T., Quan, W. (eds.) GD 2007. LNCS, vol. 4875, pp. 291–302. Springer, Heidelberg (2008)
7. Feng, Q.: Algorithms for Drawing Clustered Graphs. Ph.D. thesis, The University of Newcastle, Australia (1997)
8. Feng, Q., Cohen, R.F., Eades, P.: Planarity for clustered graphs. In: Spirakis, P.G. (ed.) ESA 1995. LNCS, vol. 979, pp. 213–226. Springer, Heidelberg (1995)
9. Garey, M.R., Johnson, D.S.: Computers and Intractability: A Guide to the Theory of NP-Completeness. W.H. Freeman, New York (1979)
10. Goodrich, M.T., Lueker, G.S., Sun, J.Z.: C-planarity of extrovert clustered graphs. In: Healy, P., Nikolov, N.S. (eds.) GD 2005. LNCS, vol. 3843, pp. 211–222. Springer, Heidelberg (2006)
11. Gutwenger, C., Jünger, M., Leipert, S., Mutzel, P., Percan, M., Weiskircher, R.: Advances in c-planarity testing of clustered graphs. In: Goodrich, M.T., Kobourov, S.G. (eds.) GD 2002. LNCS, vol. 2528, pp. 220–235. Springer, Heidelberg (2002)
12. Jelinek, V., Jelinkova, E., Kratochvil, J., Lidicky, B.: Clustered planarity: Embedded clustered graphs with two-component clusters. In: Tollis, I.G., Patrignani, M. (eds.) GD 2008. LNCS, vol. 5417, pp. 121–132. Springer, Heidelberg (2009)
13. Jelinkova, E., Kara, J., Kratochvil, J., Pergel, M., Suchy, O., Vyskocil, T.: Clustered planarity: Small clusters in eulerian graphs. In: Hong, S.-H., Nishizeki, T., Quan, W. (eds.) GD 2007. LNCS, vol. 4875, pp. 303–314. Springer, Heidelberg (2008)
14. Nishizeki, T., Chiba, N.: Planar Graphs: Theory and Algorithms. Ann. Discrete Math., vol. 32. North-Holland, Amsterdam (1988)