

On Embedding a Cycle in a Plane Graph^{*}

(Extended Abstract)

Pier Francesco Cortese, Giuseppe Di Battista,
Maurizio Patrignani, and Maurizio Pizzonia

Università Roma Tre
{cortese, gdb, patrigna, pizzonia}@dia.uniroma3.it

Abstract. Consider a planar drawing Γ of a planar graph G such that the vertices are drawn as small circles and the edges are drawn as thin strips. Consider a cycle c of G . Is it possible to draw c as a non-intersecting closed curve inside Γ , following the circles that correspond in Γ to the vertices of c and the strips that connect them? We show that this test can be done in polynomial time and study this problem in the framework of clustered planarity for highly non-connected clustered graphs.

1 Introduction

Let Γ be a planar drawing of a planar graph G and c be a cycle composed of vertices and edges of G . We deal with the problem of testing if c can be drawn on Γ without crossings.

Of course, if the vertices of G are drawn as points, the edges as simple curves, and the drawing of c must coincide with the drawing of its vertices and edges, then the problem is trivial. In this case c can be drawn without crossings if and only if it is simple.

We consider the problem from a different point of view. Namely, we suppose that the vertices of G are drawn in Γ as “small circles” and the edges as “thin strips”. Hence, c can pass several times through a vertex or through an edge without crossing itself. In this case even a non-simple cycle can have a chance to be drawn without crossings.

The problem, in our opinion, is interesting in itself. However, we study it because of its meaning in the field of *clustered planarity* [11, 10].

Clustered planarity is a classical Graph Drawing topic (see [4] for a survey). A *cluster* of a graph is a non empty subset of its vertices. A *clustered graph* $C(G, T)$ is a graph G plus a rooted tree T such that the leaves of T are the vertices of G . Each node ν of T corresponds to the cluster $V(\nu)$ of G whose

^{*} Work partially supported by European Commission - Fet Open project DELIS - Dynamically Evolving Large Scale Information Systems - Contract no 001907, by “Project ALGO-NEXT: Algorithms for the Next Generation Internet and Web: Methodologies, Design, and Experiments”, MIUR Programmi di Ricerca Scientifica di Rilevante Interesse Nazionale, and by “The Multichannel Adaptive Information Systems (MAIS) Project”, MIUR-FIRB.

vertices are the leaves of the subtree rooted at ν . The subgraph of G induced by $V(\nu)$ is denoted as $G(\nu)$. An edge e between a vertex of $V(\nu)$ and a vertex of $V - V(\nu)$ is *incident* to ν . Graph G and tree T are called *underlying graph* and *inclusion tree*, respectively. A clustered graph is *connected* if for each node ν of T we have that $G(\nu)$ is connected.

In a *drawing* of a clustered graph vertices and edges of G are drawn as points and curves as usual [8], and each node ν of T is a simple closed region $R(\nu)$ such that: (i) $R(\nu)$ contains the drawing of $G(\nu)$; (ii) $R(\nu)$ contains a region $R(\mu)$ if and only if μ is a descendant of ν in T ; and (iii) any two regions $R(\nu_1)$ and $R(\nu_2)$ do not intersect if ν_1 is not a descendant or an ancestor of ν_2 . Consider an edge e and a node ν of T . If e is incident on ν and e crosses the boundary of $R(\nu)$ more than once, we say that edge e and region $R(\nu)$ have an *edge-region crossing*. Also, edge e and region $R(\nu)$ have an *edge-region crossing* if e is not incident on ν and e crosses the boundary of $R(\nu)$. A drawing of a clustered graph is *c-planar* if it does not have edge crossings and edge-region crossings. A clustered graph is *c-planar* if it has a c-planar drawing. C-planarity testing algorithms for connected clustered graphs are shown in [13, 11, 6]. A planarization algorithm for connected clustered graph is shown in [7].

However, the complexity of the c-planarity testing for a non connected clustered graph is still unknown. A contribution on this topic has been given by Gutwenger et al. who presented a polynomial time algorithm for c-planarity testing for *almost connected* clustered graphs [12].

Another contribution studying the interplay between c-planarity and connectivity has been presented in [3] by Cornelsen and Wagner. They show that a *completely connected* clustered graph is c-planar if and only if its underlying graph is planar. A completely connected clustered graph is so that not only each cluster is connected but also its complement is connected.

A clustered graph $C(G, T)$ is *flat* if all the leaves of T have distance two from the root. This implies that all the non-root clusters have depth 1 in T . Hence, in a flat clustered graph $C(G, T)$ a *graph of the clusters* $G^1(C)$ can be identified. Vertices of $G^1(C)$ are the children of the root of T and an edge (μ, ν) exists if and only if an edge of G exists incident to both μ and ν .

Flat clustered graphs offer a way to deepen our insight into the properties of non-connected c-planar clustered graphs. In fact, by changing the families of the graphs G and $G^1(C)$, c-planarity problems of increasing complexity can be identified. The works in [2, 1] by Biedl, Kaufmann, and Mutzel can be interpreted as a linear time c-planarity test for non connected flat clustered graphs with exactly two clusters.

A *clustered cycle* is a flat clustered graph whose underlying graph is a cycle. In [5] it is shown that for a clustered cycle $C(G, T)$ where $G^1(C)$ is also a cycle, the c-planarity testing and embedding problem can be solved in linear time.

A *rigid clustered cycle* is a clustered cycle C in which $G^1(C)$ has a prescribed planar embedding. In this paper we tackle the c-planarity testing and embedding problem for rigid clustered cycles. Namely, consider again the problem stated at the beginning of this section according to the above definitions. The cycle is

the underlying graph of a flat clustered graph and the nodes of the graph are the clusters. If you are able to find a drawing of the cycle without intersections you are also able to find a c-planar embedding for the rigid clustered cycle and vice versa.

In this paper we present the following results. We develop a new theory for dealing with rigid clustered cycles, based on operations that preserve their c-planarity (Section 3). We show that the c-planarity of a rigid clustered cycle can be tested in polynomial time (Section 4). As a side effect we also solve in polynomial time the cycle drawing problem stated at the beginning of the section. If the rigid clustered cycle is c-planar we also show a simple method for computing a planar embedding of it (Section 5). Section 2 contains basic definitions, while conclusions and open problems are in Section 6.

2 Basic Definitions

We assume familiarity with connectivity and planarity of graphs [9, 8].

In the following we need a slightly wider definition of clustered cycle in which $G^1(C)$ is allowed to have multiple edges between two nodes. We define a *clustered cycle* $C(G, G^1, \Phi_V, \Phi_E)$, where G^1 is a graph, possibly with multiple edges, G is a cycle, Φ_V maps each vertex of G to a vertex of G^1 , and Φ_E maps each edge of G between vertices $v_1 \in \mu_1$ and $v_2 \in \mu_2$, where $\mu_1 \neq \mu_2$, to an edge of G^1 between vertices μ_1 and μ_2 .

In the following, to avoid ambiguities, we denote G^1 as $G^1(C)$, its edges will be called *pipes* while its vertices will be called *nodes* or *clusters*.

Given a cluster $\mu \in G^1(C)$, we denote by $deg(\mu)$ the number of pipes that are adjacent to μ in $G^1(C)$, where multiple pipes count for their multiplicity. The *size* of a pipe of $G^1(C)$ is the number of edges of G it contains.

It is easy to see that a path in G whose vertices belong to the same cluster can be collapsed into a single vertex without affecting the c-planarity property of the clustered cycle. Hence, in the following we consider only clustered cycles where consecutive vertices belong to distinct clusters. We call *cusp* a vertex v of G whose incident edges e_1 and e_2 are such that $\Phi_E(e_1) = \Phi_E(e_2)$.

Given a rigid clustered cycle C the *embedding* Λ of C is the specification, for each pipe a in $G^1(C)$ and for each end node μ of a , of the total ordering $\lambda_\mu(a)$ of the edges contained in a when turning around μ clockwise. An embedding of a clustered cycle is c-planar if there exists a planar drawing of C that respects such embedding. If an embedding is c-planar, for each pipe $a = (\mu, \nu)$, we have that $\lambda_\mu(a) = \bar{\lambda}_\nu(a)$, where $\bar{\lambda}_\nu(a)$ denotes the reverse of $\lambda_\nu(a)$.

3 Fountain Clusters

Consider a clustered cycle C and one of its clusters $\mu = \{v_1, \dots, v_q\}$. For each v_i let w_i and z_i be its neighbors. Cluster μ is a *fountain cluster* if there exists a cluster ν different from μ such that for each v_i we have that $w_i \in \nu$ or $z_i \in \nu$ (see Fig. 1 for an example). We call *base* of μ the pipe of $G^1(C)$ between μ and ν .

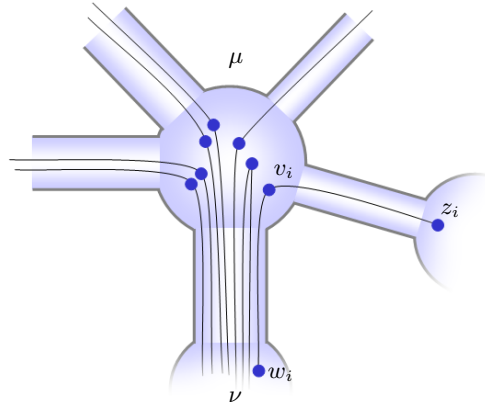


Fig. 1. A fountain cluster

A *fountain clustered cycle* is a clustered cycle in which each cluster is a fountain cluster.

Let μ be a fountain cluster and let b be a base of μ . The following properties hold:

Property 1. Cluster μ has a second base $b' \neq b$ if and only if $\text{deg}(\mu) = 2$ and no cusps belongs to μ . Otherwise μ has a single base.

Property 2. The edges incident to a cusp v of μ belong to b .

Property 3. Let a be a pipe incident to μ . If a is also a base for μ then $\text{size}(a) = \text{size}(b)$, otherwise $\text{size}(a) < \text{size}(b)$.

3.1 Cluster Expansion

Given a cluster μ of C , we call *cluster expansion* of μ the following operation (see Fig. 2), that produces the clustered cycle C' .

Let a_1, \dots, a_k be the pipes incident to μ , where $k = \text{deg}(\mu)$. Let v a vertex belonging to μ , and let e_i and e_j be the edges incident to v , where $e_i \in a_i$ and $e_j \in a_j$, respectively. Note that if v is a cusp, then $a_i = a_j$.

Cluster μ is replaced in C' with k new clusters μ_1, \dots, μ_k , each one incident to pipes a_1, \dots, a_k , respectively. All the other clusters of C are unchanged in C' . Each non-cusp vertex v in μ having edges $e_i \in a_i$ and $e_j \in a_j$ is represented in C' by two new vertices v' and v'' , with $\Phi_V(v') = \mu_i$ and $\Phi_V(v'') = \mu_j$. A new pipe (μ_i, μ_j) is inserted (if not already present) and a new edge (v', v'') is added such that $\Phi_E(v', v'') = (\mu_i, \mu_j)$. Each cusp vertex v having its edges in pipe a_i stays unchanged in C' , and belongs to cluster μ_i .

Note that a cluster μ_i produced by the cluster expansion is a fountain cluster with base a_i . Hence, after one expansion the number of non-fountain clusters of C' is not greater than the number of non-fountain clusters of C . Also, before applying the cluster expansion, μ could be the end node of multiple pipes. After the cluster expansion these multiple pipes are eliminated.

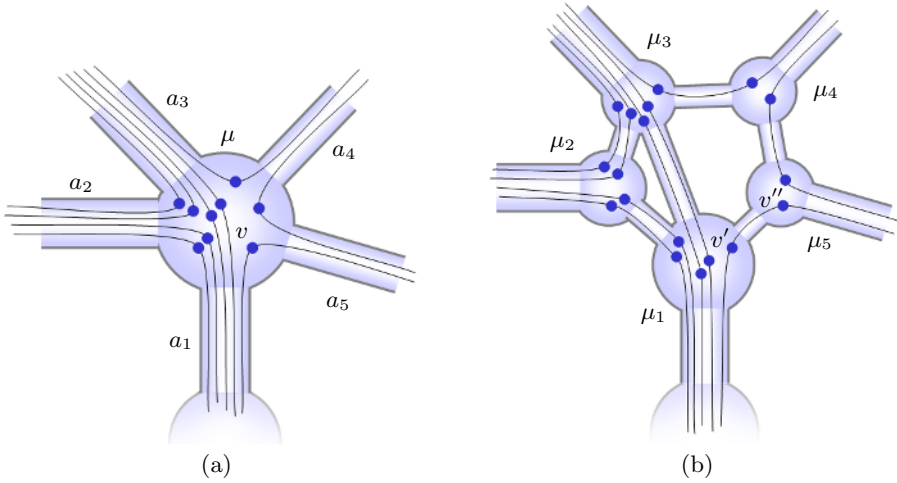


Fig. 2. An example of cluster expansion: (a) A non-fountain cluster μ . (b) The result of the cluster expansion.

Up to now, the expansion operation has been defined without considering the embedding of C and C' . If C is embedded (rigid) it is easy to extend the definition of cluster expansion considering also embedding issues. Namely, we embed the new pipes around the new nodes with the same order the old edges had in C . Note that, even if the starting embedding is planar, the resulting embedding may be not planar due to the new pipes inserted among the clusters μ_1, \dots, μ_k .

Given a rigid clustered cycle C , a cluster expansion of one of its clusters μ is *feasible* if the embedding induced on $G^1(C')$ is planar, that is, if C' is a rigid clustered cycle.

Lemma 1. *Given a rigid clustered cycle C , if a cluster expansion of one of its clusters μ is not feasible, then C is not c-planar.*

Proof. If the cluster expansion of μ is not feasible, then the induced embedding on $G^1(C')$ contains a crossing, that is, it contains two pipes (μ_i, μ_h) and (μ_j, μ_l) , with $i < j < h < l$. This implies that there exist two paths of G , one traversing clusters ν_i, μ, ν_h and the other traversing ν_j, μ, ν_l . Since the embedding of μ is fixed, this two paths cannot be drawn without intersections. \square

A *cluster expansion* operation on a clustered cycle C is done performing a cluster expansion for each non-fountain cluster of C . A cluster expansion of a rigid clustered cycle is *feasible* if all the required cluster expansions are feasible, that is if the result is a rigid clustered cycle.

Property 4. The cluster expansion of a clustered cycle produces a fountain clustered cycle.

Lemma 2. *Let C be a rigid clustered cycle and let μ be a cluster of C . Let C' be the result of a feasible cluster expansion applied to μ . C is c -planar iff C' is c -planar.*

Proof sketch. Suppose that C is c -planar, and let Γ be a c -planar embedding of C . A c -planar embedding Γ' of C' can be computed as follows. For each pipe that is present both in C and in C' , including pipes a_1, \dots, a_k incident to μ , we assume that the order of edges in Γ' is the same as in Γ . The order of the edges inside the pipes added among nodes μ_1, \dots, μ_k is determined by their order in the bases a_1, \dots, a_k . Hence, the c -planarity of Γ' follows from the c -planarity of Γ .

Suppose now that C' is c -planar, and let Γ' be a c -planar embedding of C' . A c -planar embedding Γ of C can directly be obtained from Γ' . Since all pipes of C are also present in C' , the order of their edges can be assumed to be the same as in Γ' . Consider edge e of pipe (μ_i, μ_j) in Γ' . The path e_i, e, e_j of Γ' , where $e_i \in a_i$ and $e_j \in a_j$ corresponds to path e_i, e_j in Γ . Hence, the c -planarity of Γ' implies the c -planarity of Γ . \square

By repeatedly applying Lemma 2 we have:

Lemma 3. *Let C be a rigid clustered cycle and let C' be a feasible cluster expansion of C . C is c -planar iff C' is c -planar.*

3.2 Pipe Contraction

We call a pipe b between two fountain clusters μ and ν *contractible* if (i) b is the only pipe between μ and ν , (ii) b is a base for both μ and ν , and (iii) b is the only base for one of them.

We define the *pipe contraction* operation on a contractible pipe b as follows. The pipe contraction produces a clustered cycle C' starting from a clustered cycle C by replacing μ, ν , and b , with a new cluster μ' , which is adjacent to all the clusters which μ and ν were adjacent to.

If μ and ν were adjacent to the same cluster ρ , μ' is doubly adjacent to ρ ; that is, the pipe contraction may introduce multiple pipes incident to μ' .

Each edge e_{in} entering μ or ν belongs to a path $p_C = e_{in}, v, e_1, v_1, \dots, e_k, v_k, e_{out}$, where e_{out} is the first edge exiting μ or ν and $\Phi_E(e_i) = b, i = 1, \dots, k$. Since b is a base for both μ and ν , $k \geq 1$. Path p_C is replaced by $p_{C'} = e_{in}, v_{\mu'}, e_{out}$, with $\Phi_V(v_{\mu'}) = \mu'$.

An example of pipe contraction is shown in Fig. 3. Note that the new cluster μ' is, in general, not a fountain cluster. If C has a prescribed embedding we assume that the result has also a prescribed embedding in which the circular order of the pipes around μ' is the same as the circular order they have in C around the subgraph composed of μ, ν , and b .

Lemma 4. *Let C be a fountain clustered cycle and C' be obtained from C by applying a pipe contraction operation. C is c -planar iff C' is c -planar.*

Proof sketch. Suppose that C is c -planar, let Γ be a c -planar drawing of C , we show how to build a c -planar drawing Γ' of C' by slightly modifying Γ . Namely,

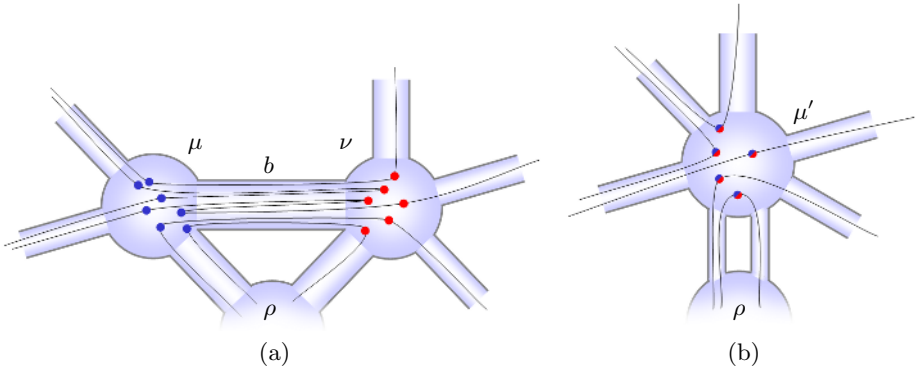


Fig. 3. An example of pipe contraction: (a) pipe b before contraction; (b) The result of the contraction of b

region $R(\mu')$ is the union of $R(\mu)$, $R(\nu)$, and the stripe corresponding to b . (Observe that $R(\mu')$ is connected.) Each path $p_C = e_{in}, v, e_1, v_1, \dots, e_k, v_k, e_{out}$ of C , with $\Phi_E(e_i) = b$, is replaced by $p_{C'} = e_{in}, v_{\mu'}, e_{out}$, where $v_{\mu'}$ replaces v , and all vertices v_i , with $i = 1, \dots, k$, are removed joining their incident edges. It is easy to see that the obtained drawing is a c -planar drawing of C' .

Suppose now that C' is c -planar, and let Γ' be a c -planar drawing of C' . We provide a c -planar drawing Γ of C by suitably modifying Γ' . We take region $R(\mu) = R(\mu')$. Observe that in Γ' all the pipes that were incident to ν are consecutively attached to the border of $R(\mu')$. Hence, it is possible to add two arbitrarily thin stripes, corresponding to b and $R(\nu)$, along the border of $R(\mu')$ in such a way to intersect those pipes only (see Fig 4.b).

Now, consider the edges entering $R(\mu')$ that were incident to μ before contraction in counterclockwise order. Let e_{in} be the current edge and $p_{C'} = e_{in}, v_{\mu'}, e_{out}$

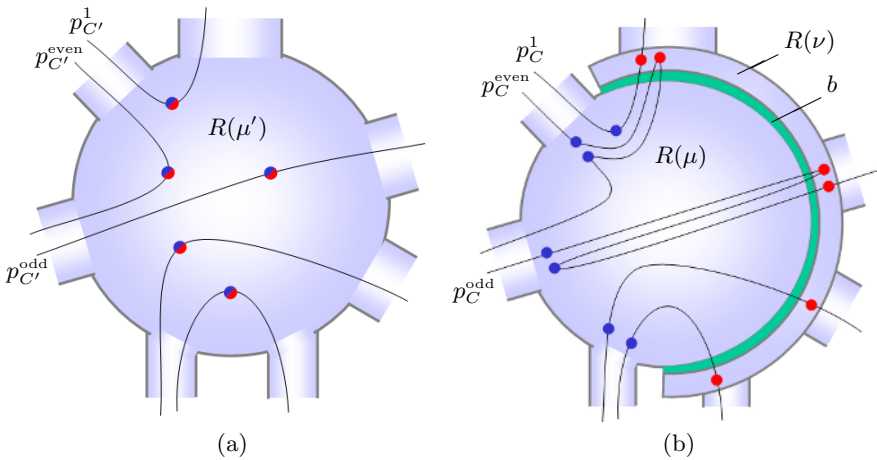


Fig. 4. A drawing Γ' of C' (a) and the corresponding drawing Γ of C (b)

be the path of C' that replaced $p_C = e_{in}, v, e_1, v_1, \dots, e_k, v_k, e_{out}$. (Remember that $k \geq 1$.) If $k = 1$, it is easy to obtain a drawing of $p_C = e_{in}, v, e_1, v_1, e_{out}$ starting from the drawing of $p_{C'} = e_{in}, v_{\mu'}, e_{out}$ by replacing $v_{\mu'}$ with v and splitting e_{out} with a vertex v_1 in such a way that v_1 is into $R(\mu)$ (see paths $p_{C'}^1$ and p_C^1 of Fig 4 for an example).

Analogously, if k is odd (e_{out} was incident to ν) it is possible to draw $p_C = e_{in}, v, e_1, v_1, \dots, e_k, v_k, e_{out}$ in a thin stripe along the drawing of $p_{C'} = e_{in}, v_{\mu'}, e_{out}$ (see paths $p_{C'}^{odd}$ and p_C^{odd} of Fig 4 for an example). If k is even, then both e_{in} and e_{out} were incident to μ in C . In this case the drawing of $p_{C'} = e_{in}, v_{\mu'}, e_{out}$ does not immediately provide a drawing of $p_C = e_{in}, v, e_1, v_1, \dots, e_k, v_k, e_{out}$, which can be built as follows. Vertex v is placed into $R(\mu)$ as edge e_{in} crosses the border of $R(\mu)$. Edge e_1 follows clockwise the border of $R(\mu)$ till the previous edge e'_{in} entering $R(\mu)$ is found (or $R(\nu)$ is reached). Since edges e_{in} are considered in counterclockwise order and since b was a base for both μ and ν , path $p'_{C'}$, starting with edge e'_{in} , always has vertex v' into $R(\mu)$ and v'_1 into $R(\nu)$. Therefore, edge e_1 can be drawn arbitrarily near to path $p'_{C'}$ and can be terminated with v_1 placed into $R(\nu)$. Edges e_i , with $i = 2, \dots, k$, can be drawn in an arbitrarily thin stripe adjacent to e_1 , positioning v_i alternately into $R(\mu)$ and $R(\nu)$. Finally, edge e_{out} can follow path $p_{C'}$ to exit $R(\nu)$ (see paths $p_{C'}^{even}$ and p_C^{even} of Fig 4 for an example). \square

4 C-Planarity Testing of Clustered Cycles

In this section we describe a c-planarity testing algorithm for rigid clustered cycles. The following lemmas state properties of clustered cycles which are needed to prove the correctness of the algorithm.

Lemma 5. *Let C be a fountain clustered cycle such that $G^1(C)$ is not a simple cycle and has not multiple pipes. There exists at least one contractible pipe b^* in $G^1(C)$.*

Proof sketch. Consider a pipe $b = (\mu, \nu)$ of maximum size. Since b is the pipe of maximum size for both μ and ν , by Property 3, b is the base for both. If one between μ and ν (say μ) has degree different from two then, by Property 1, μ admits a single base and the statement holds with $b^* = b$. Otherwise, suppose that both μ and ν have degree two and that both have two bases. Let b_1 be the second base of μ . Due to Property 3, $size(b_1) = size(b)$. Therefore b_1 is also a base for its incident cluster $\mu_1 \neq \mu$. If b_1 is the only base for μ_1 then the statement holds with $b^* = b_1$, otherwise μ_1 has a second base $b_2 \neq b_1$, with $size(b_2) = size(b_1)$, and we apply the same argument to b_2 . Since $G^1(C)$ is not a simple cycle the current pipe b_i is different from b and there exists at least a j for which b_j is the only base for μ_j . \square

We introduce a quantity that will be used to analyze the algorithm both in terms of correctness and in terms of time complexity. Intuitively, it is an indicator of the structural complexity of $G^1(C)$. We denote by $\mathcal{E}(C)$ the following quantity:

$$\mathcal{E}(C) = \sum_{a \in \{\text{pipes of } G^1(C)\}} (\text{size}(a))^2.$$

We now concentrate on a pair of consecutive contraction-expansion operations and show how \mathcal{E} changes.

Lemma 6. *Let C be a fountain clustered cycle and let $b = (\mu, \nu)$ be a contractible pipe which is the only base for μ . Let C^* be the clustered cycle obtained by applying a pipe contraction to b followed by a cluster expansion of the obtained cluster μ' . We have that $\mathcal{E}(C^*) < \mathcal{E}(C)$.*

Proof. Let C' be the clustered cycle generated by the pipe contraction applied to b . C' contains all the pipes of C with the exception of b , then $\mathcal{E}(C') = \mathcal{E}(C) - (\text{size}(b))^2$. Clustered cycle C^* has the same pipes of C' plus a set of new pipes a_1, \dots, a_k . If $k = 0$ then $\mathcal{E}(C^*) = \mathcal{E}(C') < \mathcal{E}(C)$. If $k = 1$ then $\text{deg}(\mu') = \text{deg}(\mu) = \text{deg}(\nu) = 2$. Since b is the only base for μ by Properties 1 and 2, b contains edges incident to cusps which are not present in a_1 . Therefore $\mathcal{E}(C^*) < \mathcal{E}(C)$. Suppose $k > 2$. We have that $\mathcal{E}(C^*) = \mathcal{E}(C') + \sum_{j=1}^k (\text{size}(a_j))^2 = \mathcal{E}(C) - (\text{size}(b))^2 + \sum_{j=1}^k (\text{size}(a_j))^2$. Observe that each edge contained in the pipes a_1, \dots, a_k is generated by the split of a vertex in μ' , and that the number of vertices in μ' is at most $\text{size}(b)$. Then, $\sum_{j=1}^k \text{size}(a_j) \leq \text{size}(b)$. Hence, $\sum_{j=1}^k (\text{size}(a_j))^2 < (\text{size}(b))^2$, and the statement follows. \square

Lemma 7. *A clustered cycle C whose graph of the clusters $G^1(C)$ is a path is c-planar.*

Proof sketch. Let μ_1, \dots, μ_m be the nodes of $G^1(C)$ in the order in which they appear in the path. A planar embedding of C can be built as follows. Traverse the cycle G starting from a vertex in μ_1 . Each edge e belonging to pipe $a = (\mu_i, \mu_j)$ is inserted at the last position of $\lambda_{\mu_i}(a)$ and at the first position of $\lambda_{\mu_j}(a)$. When the path comes back to μ_1 for the last time it can be connected to the starting point preserving c-planarity. \square

We are now ready to introduce the c-planarity testing algorithm for a rigid clustered cycle C . First, the algorithm performs a cluster expansion for each non-fountain cluster. If one of such expansions is not feasible, then, according to Lemma 1, C is not c-planar. If all the expansions are feasible, according to Property 4, we obtain a fountain clustered cycle C^f , which is c-planar iff C is c-planar. If the clusters of C^f form a cycle, then the c-planarity can be easily tested using the results described in [5]. If $G^1(C^f)$ is a path, then Lemma 7 states that C^f is c-planar. If the clusters of C^f form neither a cycle nor a path, then Lemma 5 ensures that there exists a contractible pipe $b^* = (\mu, \nu)$. Perform a contraction operation on b^* . Perform a cluster expansion on the resulting cluster. These last two steps are performed until the clusters of the clustered cycle form a cycle, or a path, or a cluster expansion fails. Note that a pipe contraction may temporarily generate multiple pipes; however, the subsequent cluster expansion produces a new clustered cycle which has no multiple pipes. The algorithm, called *ClusteredCyclePlanarityTesting*, is formally described below.

Algorithm *ClusteredCyclePlanarityTesting***input** A rigid clustered cycle C **output** True if C is c-planar, false otherwise

```

for all non-fountain clusters  $\mu$  in  $C$  do
  perform a cluster expansion of  $\mu$ 
  if the cluster expansion of  $\mu$  is not feasible then
    return false
  end if
end for
{at this point  $C$  is a fountain clustered cycle}
while  $C$  is not a cycle or a path do
  let  $b$  be a contractible pipe of  $C$ 
  apply a pipe contraction to  $b$ , obtaining cluster  $\mu'$ .
  perform a cluster expansion of  $\mu'$ 
  if the cluster expansion of  $\mu'$  is not feasible then
    return false
  end if
end while
{at this point  $C$  is a cycle or a path}
if  $C$  is a cycle then
  return the result of the c-planarity testing on  $C$ 
else
  return true
end if

```

Theorem 1. *There exists a polynomial time algorithm to test if a rigid clustered cycle is c-planar.*

Proof. First, we prove that algorithm `ClusteredCyclePlanarityTesting` can be always executed in a polynomial number of steps. Let C be a rigid clustered cycle whose underlying cycle is G and be n the number of vertices of G . In the first phase of the algorithm a cluster expansion is performed for all the non-fountain clusters. Each cluster expansion can be performed in polynomial time. At the end of this phase the number of vertices is at most $2n$. Suppose that \overline{E} is the value of $\mathcal{E}(C)$ at the end of this phase. We have that $\overline{E} = O(n^2)$.

By Lemma 6 each pair of pipe contraction and cluster expansion decreases $\mathcal{E}(C)$ of at least one unit. Hence, the body of the **while** cycle is executed at most \overline{E} times. Also, a contractible pipe always exists (see Lemma 5) and can be determined in constant time using a suitable data structure that contains the candidate bases and that is updated after each operation. This proves that algorithm `ClusteredCyclePlanarityTesting` terminates in polynomial time.

Second, we prove that algorithm `ClusteredCyclePlanarityTesting` gives the correct result. Lemmas 2, 3, and 4 guarantee that the cluster expansion and pipe contraction operations can be applied without modifying the c-planarity

property of the graph, while if a cluster expansion is not feasible the graph is not c -planar. If none of the cluster expansions fails, either the algorithm produces a k -cluster cycle and applies the c -planarity testing algorithm shown in [5], or produces a clustered path, which by Lemma 7 is always c -planar. Also, (see the above discussion) the algorithm always terminates. \square

5 Computing C -Planar Embeddings of Clustered Cycles

In this section we show how to build an embedding for a c -planar rigid clustered cycle. We assume that Algorithm `ClusteredCyclePlanarityTesting`, described in Section 4, has been applied, and that each step of the algorithm has been recorded. The clustered cycle C_{end} obtained at the last step of the execution of that algorithm is such that $G^1(C_{end})$ is a cycle or a path. A c -planar embedding of C_{end} can be easily computed by using the results described in [5], if $G^1(C_{end})$ is a cycle, or by using the technique introduced in the proof of Lemma 7, if $G^1(C_{end})$ is a path.

The embedding of the input clustered cycle can be obtained by going through the transformations operated by Algorithm `ClusteredCyclePlanarityTesting` in reverse order starting from a c -planar embedding of C_{end} . Algorithm `ClusteredCyclePlanarityTesting` performs two kind of operations: pipe contraction and cluster expansion.

For each cluster expansion on a clustered cycle C , which produces a cluster cycle C' , the embedding of C is directly obtained from the embedding of C' as described in the proof of Lemma 2 since all pipes in C' are also in C and their embedding do not change.

For each pipe contraction on a clustered cycle C , which produces a cluster cycle C' , only part of the embedding of C can be directly obtained from the embedding of C' since C has one more pipe (the contracted one) with respect to C' . The proof of Lemma 4 describes how to compute a c -planar embedding of C starting from a c -planar embedding of C' .

From the above discussion and from the fact that `ClusteredCyclePlanarityTesting` has a polynomial time complexity we can state the following result.

Theorem 2. *Given a c -planar rigid clustered cycle, a c -planar embedding of it can be computed in polynomial time.*

6 Conclusions

In this paper we addressed the problem of drawing, without crossings, a cycle in a planar embedded graph and have shown that the problem can be solved in polynomial time.

If we interpret the problem and the result from the clustered planarity perspective it turns out that we have identified a new family of flat clustered graphs that are highly non-connected and whose c -planarity can be tested in polynomial time. This might be useful for deepening the insight into the general problem of

testing the c -planarity of non-connected clustered graphs, whose computational complexity is still unknown.

However, we point out that a trivial generalization of the result to flat clustered graphs whose underlying graph is a general graph fails. In fact, it is easily to find clustered graphs which are not c -planar while all cycles of their underlying graphs are separately c -planar.

References

1. T. C. Biedl. Drawing planar partitions III: Two constrained embedding problems. Tech. Report RRR 13-98, RUTCOR Rutgers University, 1998.
2. T. C. Biedl, M. Kaufmann, and P. Mutzel. Drawing planar partitions II: HH-Drawings. In *Workshop on Graph-Theoretic Concepts in Computer Science (WG'98)*, volume 1517, pages 124–136. Springer-Verlag, 1998.
3. S. Cornelsen and D. Wagner. Completely connected clustered graphs. In *Proc. 29th Intl. Workshop on Graph-Theoretic Concepts in Computer Science (WG 2003)*, volume 2880 of *LNCS*, pages 168–179. Springer-Verlag, 2003.
4. P. F. Cortese and G. Di Battista. Clustered planarity. In *SCG '05: Proceedings of the twenty-first annual symposium on Computational geometry*, pages 32–34, New York, NY, USA, 2005. ACM Press.
5. P. F. Cortese, G. Di Battista, M. Patrignani, and M. Pizzonia. Clustering cycles into cycles of clusters. In János Pach, editor, *Proc. Graph Drawing 2004 (GD'04)*, volume 3383 of *LNCS*, pages 100–110. Springer-Verlag, 2004.
6. E. Dahlhaus. Linear time algorithm to recognize clustered planar graphs and its parallelization. In C.L. Lucchesi, editor, *LATIN '98, 3rd Latin American symposium on theoretical informatics, Campinas, Brazil, April 20–24, 1998*, volume 1380 of *LNCS*, pages 239–248, 1998.
7. G. Di Battista, W. Didimo, and A. Marcandalli. Planarization of clustered graphs. In *Proc. Graph Drawing 2001 (GD'01)*, *LNCS*, pages 60–74. Springer-Verlag, 2001.
8. G. Di Battista, P. Eades, R. Tamassia, and I. G. Tollis. *Graph Drawing*. Prentice Hall, Upper Saddle River, NJ, 1999.
9. S. Even. *Graph Algorithms*. Computer Science Press, Potomac, Maryland, 1979.
10. Q. W. Feng, R. F. Cohen, and P. Eades. How to draw a planar clustered graph. In Ding-Zhu Du and Ming Li, editors, *Proc. COCOON'95*, volume 959 of *LNCS*, pages 21–30. Springer-Verlag, 1995.
11. Q. W. Feng, R. F. Cohen, and P. Eades. Planarity for clustered graphs. In P. Spirakis, editor, *Symposium on Algorithms (Proc. ESA '95)*, volume 979 of *LNCS*, pages 213–226. Springer-Verlag, 1995.
12. C. Gutwenger, M. Jünger, S. Leipert, P. Mutzel, M. Percan, and René Weiskircher. Advances in C -planarity testing of clustered graphs. In Stephen G. Kobourov and Michael T. Goodrich, editors, *Proc. Graph Drawing 2002 (GD'02)*, volume 2528 of *LNCS*, pages 220–235. Springer-Verlag, 2002.
13. T. Lengauer. Hierarchical planarity testing algorithms. *J. ACM*, 36(3):474–509, 1989.