Completely Connected Clustered Graphs *

Sabine Cornelsen[1] and Dorothea Wagner[2]

[1] University of Konstanz, Department of Computer & Information Science
[2] University of Karlsruhe, Department of Computer Science

Abstract. Planar drawings of clustered graphs are considered. We introduce the notion of completely connected clustered graphs, i.e. hierarchically clustered graphs that have the property that not only every cluster but also each complement of a cluster induces a connected subgraph. As a main result, we prove that a completely connected clustered graph is c-planar if and only if the underlying graph is planar. Further, we investigate the influence of the root of the inclusion tree to the choice of the outer face of the underlying graph and vice versa.

1 Introduction

A frequently used method for visualizing a clustered structure of a graph is to draw every cluster as a simple closed region (e.g. an axis-parallel rectangle) bounded by a simple closed curve. Algorithms and data structures for representing clusterings in general graphs according to this approach can, for example, be found in [1, 14, 19–21]. Feng et al. [11] defined planarity – called c-planarity – for clustered graphs. Since then algorithms for drawing c-planar clustered graphs with respect to different aesthetic criteria were developed [6–10, 17].

For connected clustered graphs, i.e. for clustered graphs in which every cluster induces a connected subgraph, Feng et al. [11] gave a c-planarity criterion. They used this criterion to test in quadratic time, whether a connected clustered graph is c-planar. A linear time algorithm that solves this problem was given by Dahlhaus [4]. A first attempt for testing whether certain not necessarily connected clustered graphs are c-planar was done by Gutwenger et al. [13]. As far as we know, the complexity status for testing whether an arbitrary clustered graph is c-planar is still open.

Motivated by drawings of minimal cuts [2, 3], we consider completely connected clustered graphs, i.e. clustered graphs in which not only every cluster, but also the complement of each cluster is connected. This graph class has also applications in triangulating c-planar clustered graphs [15]. Very surprisingly it turns out that a completely connected clustered graph is c-planar if only the underlying graph is planar. In this paper, we consider varying roots of the inclusion tree. Originally, this was also motivated by drawings of cuts but turned out to be a useful proof technique as well.

The contribution of this paper is as follows. In Section 2, we shortly summarize the definitions and results about c-planar graphs that we will use in this

* This work was partially supported by the DFG under grant BR 2158/1-1.

paper. Section 3 introduces completely connected clustered graphs. We first show that c-planarity does not depend on the choice of the root of the inclusion tree. Then, we apply this result to show that a completely connected clustered graph with underlying planar graph is c-planar. The dependence between the outer face of the underlying graph on one hand and the root of the inclusion tree on the other hand is examined in Sect. 4. Finally, we investigate in Sect. 5, whether c-planar clustered graphs can be augmented to completely connected c-planar clustered graphs and whether arbitrary completely connected clustered graphs have a completely connected c-planar clustered subgraph.

2 Preliminaries

A hierarchically clustered graph (G, T, r) – or *clustered graph*, for simplicity – as introduced by Feng et al. [11] consists of

- an (undirected) graph $G = (V, E)$,
- a tree T, and
- an inner vertex r of T such that
- the set of leaves of T is exactly V.

G is called the *underlying graph* and T the *inclusion tree* of (G, T, r). To distinguish vertices of the inclusion tree from vertices in the underlying graph, inner vertices of T are called *nodes*. We denote the tree T rooted at r by (T, r). Each node ν of T represents the *cluster* $V_r(\nu)$ of leaves in the subtree of (T, r) rooted at ν. Let S be any subset of V. By $G(S)$, we denote the subgraph of G induced by S and by $G - S$ we denote the subgraph of G induced by $V \setminus S$. An edge e of G is said to be *incident* to S, if e is incident to a vertex in S and a vertex in $V \setminus S$. If $v_1, v_2 \in V$ are two vertices then $G + \{v_1, v_2\}$ denotes the graph $(V, E \cup \{v_1, v_2\})$. A clustered graph (G, T, r) is *connected*, if each cluster induces a connected subgraph of G. A *c-planar drawing* of a clustered graph (G, T, r) consists of

- a planar drawing of the underlying graph G and
- an inclusion representation[1] of the rooted tree (T, r) such that
- each edge crosses the boundary of the drawing of a node of T at most once.

Note that the vertices of G are the leaves of T and thus have the same drawing. A clustered graph is *c-planar* if it has a c-planar drawing. A clustered graph with planar underlying graph does not have to be c-planar. An example is given in Fig. 1d. Feng et al. [11] characterized c-planar connected clustered graphs as follows.

[1] In an inclusion representation of a rooted tree (T, r), each node of T is represented by a simple closed region bounded by a simple closed curve. The drawing of a node or leaf ν of T is contained in the interior of the region representing a node μ of T if and only if μ is contained in the path from ν to r in T. The drawings of two nodes μ and ν are disjoint if neither μ is contained in the path from ν to r nor ν is contained in the path from μ to r in T.

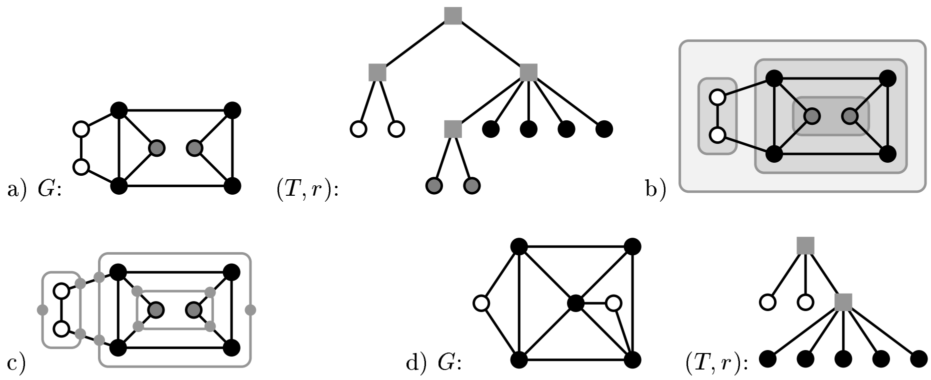

Fig. 1. a) A c-planar clustered graph that is not connected. b) A c-planar drawing of the clustered graph in a. c) The auxiliary graph associated with the c-planar drawing in b. Boundary cycles are grey. d) A connected clustered graph that is not c-planar. The correspondence of vertices of G and leaves of T is indicated by corresponding colors.

Theorem 1 **([11]).** *A connected clustered graph* (G, T, r) *is c-planar if and only if there exists a* c-planar embedding *of* G *for* (G, T, r)*, i.e. a planar embedding of* G *together with a fixed outer face such that for each node* ν *of* T *all vertices of* $V \setminus V_r(\nu)$ *are in the outer face of the drawing of* $G(V_r(\nu))$*.*

With $H \subseteq G$ we denote that a graph H is a spanning subgraph of the graph G. We call a clustered graph (H, T, r) a *subgraph* of the clustered graph (G, T, r), if $H \subseteq G$.

Theorem 2 **([11]).** *A clustered graph is c-planar if and only if it is a subgraph of a c-planar connected clustered graph.*

3 Planarity is Sufficient

A clustered graph (G, T, r) is *completely connected* if and only if for each inner node ν of T both, $G(V_r(\nu))$ and $G - V_r(\nu)$, are connected. An example of a completely connected clustered graph is shown in Fig. 2, while the clustered graph in Fig. 1d is connected but not completely connected.

Remark 1. Let (G, T, r) be a clustered graph. The following statements are equivalent.

1. (G, T, r) is completely connected.
2. (G, T, ν) is connected for every inner node ν of T.
3. (G, T, ν) is completely connected for every inner node ν of T.

In the remainder of this section, we show that a completely connected clustered graph is c-planar if and only if the underlying graph is planar. First, we prove that it does not depend on the choice of the root of the inclusion tree whether a clustered graph is c-planar or not.

For an easier discussion, we associate an *auxiliary graph* $G_{\mathcal{D}}$ with a c-planar drawing $\mathcal{D}$ of the clustered graph (G,T,r). It is the auxiliary graph that was introduced for constructing bend-minimum orthogonal drawings of clustered graphs [2, 17]. Let V' be the set of points, in which drawings of edges and boundaries of drawings of clusters intersect. Then the vertex set of $G_{\mathcal{D}}$ is $V \cup V'$. The edge set of $G_{\mathcal{D}}$ contains two types of edges. For an edge $e = \{v, w\}$ of G, let $v_1, \ldots, v_k$ be the points in $\mathcal{D}(e) \cap V'$ in the order they occur in the drawing of e from v to w. Then $G_{\mathcal{D}}$ contains the edges $\{v, v_1\}, \{v_1, v_2\}, \ldots, \{v_k, w\}$. Let $\nu \neq r$ be a node of T. Let $v_1, \ldots, v_k$ be the points in $\partial\mathcal{D}(\nu) \cap V'$ in the order they occur in the boundary $\partial\mathcal{D}(\nu)$ of the drawing of ν. Then $G_{\mathcal{D}}$ contains the edges $\{v_1, v_2\}, \ldots, \{v_{k-1}, v_k\}, \{v_k, v_1\}$. The cycle $v_1, \ldots, v_k$ of $G_{\mathcal{D}}$ is called the *boundary cycle* of ν. (To avoid loops and parallel edges, additional vertices of degree two may be inserted into boundary cycles). We interpret the c-planar drawing $\mathcal{D}$ of (G,T,r) also as a planar drawing of $G_{\mathcal{D}}$. An example of such an auxiliary graph can be found in Fig. 1c.

Lemma 1. *Let (G,T,r) be a c-planar clustered graph and ν a node of T. Then (G,T,ν) is c-planar.*

Proof. Let $\mathcal{D}_r$ be a c-planar drawing of (G,T,r) and let G_r be the auxiliary graph of $\mathcal{D}_r$. By Theorem 2, we may assume that G_r is connected. Let $P : \nu = \nu_1, \nu_2, \ldots, \nu_\ell = r$ be the path in T between ν and r. Then

$$V_\nu(\mu) = \begin{cases} V \setminus V_r(\nu_{i-1}) & \text{if } \mu = \nu_i, i = 2, \ldots, \ell \\ V_r(\mu) & \text{if } \mu \text{ is not in P} \end{cases}$$

for a node $\mu \neq \nu$ of T. Thus for any choice of the outer face, there is a boundary cycle C_μ in G_r that separates $V_\nu(\mu)$ and $V \setminus V_\nu(\mu)$. More precisely, C_μ is the boundary cycle of μ if μ is not in P or C_μ is the boundary cycle of ν_{i-1} if $\mu = \nu_i, i = 2, \ldots, \ell$. We show now that the outer face can be chosen such that $V_\nu(\mu)$ is always contained in the simple region bounded by C_μ.

Let C be the boundary cycle of ν. Let f be a face of $\mathcal{D}_r$ that is contained in the simple region bounded by C, but incident to some edge in C. Let $\mathcal{D}_\nu$ be a drawing of G_r that has the same embedding as $\mathcal{D}_r$, but outer face f. Let $\mu_1, \ldots, \mu_k$ be the adjacent nodes of ν, such that μ_1 is on the path from ν to r in T. Then for $i = 2, \ldots, k$ it holds that $V_\nu(\mu_i) = V_r(\mu_i)$ is still inside the boundary cycle of μ_i and that $V_\nu(\mu_1) = V \setminus V_r(\nu)$ is now inside the boundary cycle of ν. Finally, for all nodes $\mu \neq \nu$ of T, there exists an $i \in \{1, \ldots, k\}$ such that $V_r(\mu) \subseteq V_r(\mu_i)$. Thus, $\mathcal{D}_\nu$ contains the cluster boundaries of all non-root nodes of a c-planar drawing of (G,T,ν). □

If the root r of a clustered graph (G,T,r) is not important – e.g., if we are only interested whether (G,T,r) is c-planar or completely connected – we will omit the root in the notation of the clustered graph. I.e. we refer to (G,T,r) by (G,T). The following theorem gives a surprisingly easy characterization of c-planar completely connected clustered graphs. Note that this result is independently described by Jünger et al. [15].

Theorem 3. *Let (G,T) be a completely connected clustered graph. Then*

$$(G,T) \text{ is c-planar} \Longleftrightarrow G \text{ is planar.}$$

Proof. Clearly, G has to be planar if (G,T) is c-planar. For the other direction, we show that for any planar embedding $\mathcal{E}$ and any outer face f_o of G the root r of T can be chosen such that $\mathcal{E}$ together with f_o is a c-planar embedding for (G,T,r). Hence, by Theorem 1, (G,T) is c-planar.

Let $v \in V$ be a vertex that is incident to the outer face f_o. Let r be a node of T that is adjacent to v in T. Let ν be any node of T. Suppose there exists a vertex $w \in V \setminus V_r(\nu)$ that is not drawn in the outer face of $G(V_r(\nu))$. Since (G,T,r) is completely connected, there exists a path from v to w in $G - V_r(\nu)$. But since v and w are contained in different faces of $G(V_r(\nu))$, this is not possible. □

The proofs of Lemma 1 and Theorem 3 even showed that every planar embedding of G is a c-planar embedding for the completely connected clustered graph (G,T). Only the outer face of G has to be chosen according to the root of T or vice versa.

4 The Root and the Outer Face

Let (G,T) be a completely connected clustered graph with planar underlying graph G and let the planar embedding $\mathcal{E}$ of G be fixed. We show in this section how the following two problems can be solved in linear time.

1. Given a fixed outer face f_o of G, which nodes r can be chosen to be the root of T such that $\mathcal{E}$ together with f_o is a c-planar embedding for the clustered graph (G,T,r)?
2. Given a fixed root r of T, which faces can be the outer face of G in a c-planar embedding for the clustered graph (G,T,r)?

For the first problem, let f_o be the outer face of $G = (V,E)$. We call a node r of T *rootable* (with respect to f_o), if the fixed embedding of G together with the outer face f_o is a c-planar embedding of (G,T,r). A c-planar drawing of (G,T,r) is called *compatible* with f_o if the drawing of the underlying graph G corresponds to the given embedding $\mathcal{E}$ of G with the fixed outer face f_o. In the proof of Theorem 3, we made the following observation.

Remark 2. A node of T is rootable, if it is adjacent to a vertex of G that is incident to the outer face f_o.

Hence, there exists at least one rootable node. The following remark follows immediately from the fact that in a c-planar embedding the complement of a cluster $V_r(\nu)$ is contained in the outer face of the subgraph $G(V_r(\nu))$.

Remark 3. Let r, ν be two nodes of T and let r be rootable. Then $V \setminus V_r(\nu)$ contains a vertex that is incident to f_o.

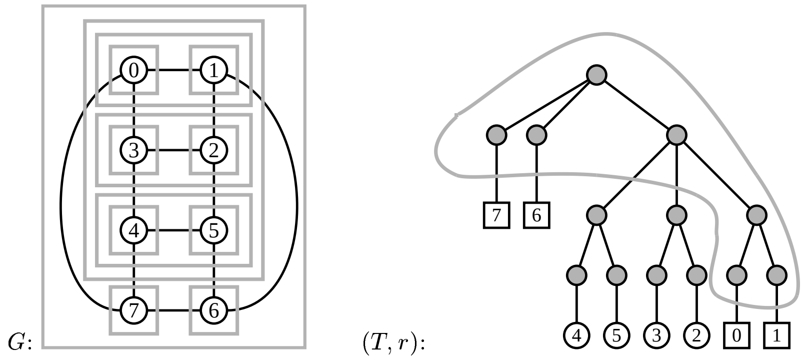

Fig. 2. A completely connected clustered graph. Rootable nodes are encircled. Vertices of G that are incident to the outer face are drawn as quadrangles in the inclusion tree.

We use this observation to characterize which nodes among those nodes that are adjacent to a rootable node are rootable themselves. Note that the following lemma is not true, if (G, T) is not completely connected. See Fig. 3 for an example.

Lemma 2. *Let r be a rootable node of T. A node ν of T that is adjacent to r is rootable if and only if $V_r(\nu)$ contains a vertex that is incident to f_o.*

Proof. **"⇒":** If ν is rootable, Remark 3 implies that $V_r(\nu) = V \setminus V_\nu(r)$ contains a vertex that is incident to f_o.

"⇐": Suppose now, that $V_r(\nu)$ contains a vertex that is incident to f_o. Let $\mathcal{D}_r$ be a c-planar drawing of (G, T, r) that is compatible with f_o. Let G_r be the auxiliary graph of $\mathcal{D}_r$.

Let C be the boundary cycle of ν. Since $V_r(\nu)$ contains a vertex that is incident to f_o there is an edge e in C that is contained in f_o. Since (G, T) is completely connected and ν is adjacent to the root r, it follows that e is incident to the outer face of $\mathcal{D}_r$:

Else let f be the face of $\mathcal{D}_r$ that is incident to e, but not contained in the simple region bounded by C. Let $C_f : v_1, \ldots, v_\ell$ be a cycle that is contained in the boundary of f such that f is contained in the simple region bounded by C_f. Suppose C_f contains boundary edges of a node μ. Let $\{v_j, v_{j+1}\}$ be the first and $\{v_{k-1}, v_k\}$ be the last such edge in C_f. Replace $v_j, \ldots, v_k$ in C_f by a path from v_j to v_k in $G(V_r(\mu))$. Since ν is adjacent to the root r, the interior of the simple region bounded by the original C_f does not intersect the simple region bounded by the boundary cycle of μ. Thus the modified C_f is still a cycle that bounds a simple region containing f. Proceeding like this, we can eliminate every boundary edge in C_f. Thus, f is contained in the simple region bounded by a cycle of G, which contradicts the fact that e is contained in f_o.

Now, let f be the inner face of $\mathcal{D}_r$ that is incident to e. A drawing of G_r with outer face f can be obtained by redrawing e through the outer face of $\mathcal{D}_r$. This yields a c-planar drawing of (G, T, ν) that is compatible with f_o. □

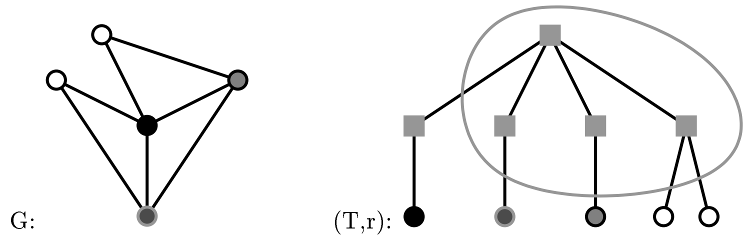

Fig. 3. A clustered graph for which Lemma 2 is not true. Rootable nodes are again encircled.

By Remark 3, the statement in Lemma 2 could also be formulated as follows. An adjacent node ν of a rootable node r is rootable if and only if both, $V_r(\nu)$ and $V \setminus V_r(\nu)$, contain a vertex that is incident to the outer face. This leads to the general characterization of rootable nodes.

Theorem 4. *Let (G, T) be a completely connected clustered graph. A node ν of T is rootable with respect to a fixed outer face f_o if and only if at least two connected components of $T - \nu$ contain vertices of G that are incident to f_o.*

Proof. **"⇒":** Suppose node ν is rootable but has a neighbor μ such that all vertices that are incident to the outer face are contained in $V_\nu(\mu)$. In this case $V_\nu(\mu)$ contains all vertices of G. Since ν is a node, i.e. an inner vertex of T, there would be leaves of T that are not vertices of G. This contradicts the definition of inclusion trees.

"⇐": Suppose now that at least two connected components of $T - \nu$ contain vertices of G that are incident to the outer face, but that ν is not rootable. Let r be a rootable node of T. Let ν_i be the first node on the path $r = \nu_1, \nu_2, \ldots, \nu_\ell = \nu$ that is not rootable. Note that $V \setminus V_r(\nu)$ is the set of leaves in one connected component of $T - \nu$. Thus, by the precondition, $V_r(\nu)$ has to contain at least one vertex that is incident to the outer face. Since $V_r(\nu) \subseteq V_r(\nu_i) = V_{\nu_{i-1}}(\nu_i)$, cluster $V_{\nu_{i-1}}(\nu_i)$ also contains a vertex that is incident to the outer face. Thus, by Lemma 2, ν_i is rootable, contradicting the choice of ν_i. □

The previous lemma and its proof lead to the following corollary.

Corollary 1. *The set of rootable nodes induces a subtree of T.*

Using Remark 2 and Theorem 4, the rootable nodes can be found in linear time by a bottom-up top-down approach starting from an arbitrary node r of T as described in Algorithm 1.

The idea of the algorithm is as follows. The node array PRED represents the predecessor of each node or leaf in the rooted tree (T, r). The outer face f_o of G is represented by the boolean node array OUTER-FACE, i.e. OUTER-FACE(v) is true if and only if v is a vertex of G – and hence a leaf of T – that is incident to the outer face.

In a first step, the algorithm proceeds from the leaves to the root r of T. It sets ONE(ν) to true if $V_r(\nu)$ contains at least one vertex that is incident to f_o. There are two cases in which ROOTABLE(ν) is set to true: if ν is adjacent to a vertex of G that is incident to f_o or if ν has at least two children μ_1, μ_2 in (T, r) such that $V_r(\mu_1)$ as well as $V_r(\mu_2)$ contain vertices that are incident to f_o.

In a second step, the algorithm proceeds from the root r to the leaves of T. It sets ROOTABLE(μ) to true, if both, $V_r(\mu)$ and $V \setminus V_r(\mu)$, contain vertices that are incident to f_o. In the end, the rootable nodes are exactly the nodes for which ROOTABLE is true.

Algorithm 1: Finding the rootable nodes.

```
Input  : tree T, boolean node array OUTER-FACE
Output : boolean node array ROOTABLE initialized to FALSE
Data   : node array PRED, boolean node array ONE initialized to false

BOTTOM-UP(node ν) begin
    foreach incident edge e = {ν, μ} of ν do
        if μ ≠ PRED(ν) then
            PRED(μ) ← ν;
            BOTTOM-UP(μ);
            if μ is a leaf and OUTER-FACE(μ) then
                ROOTABLE(ν) ← TRUE;
                ONE(ν) ← TRUE;
            else
                ROOTABLE(ν) ← ROOTABLE(ν) ∨ (ONE(ν) ∧ ONE(μ));
                ONE(ν) ← ONE(ν) ∨ ONE(μ);
end

TOP-DOWN(node ν) begin
    foreach incident edge e = {ν, μ} of ν do
        if μ ≠ PRED(ν) then
            ROOTABLE(μ) ← ROOTABLE(μ) ∨ (ONE(μ) ∧ ROOTABLE(ν));
            TOP-DOWN(μ);
end

begin
    choose a node r of T;
    PRED(r) ← r;
    BOTTOM_UP(r);
    TOP_DOWN(r);
end
```

We can also apply Theorem 4 to solve the second problem in linear time. Let r be a root of T and let $T_1, \ldots, T_k$ be connected components of $T - r$. Let v be a vertex of G and let $i \in \{1, \ldots, k\}$ be such that v is contained in T_i. Then v gets the label i. Now, by Theorem 4, each face that is incident to at least two vertices with different labels can be chosen as the outer face.

5 Subgraphs and Supergraphs

Di Battista et al. [5] showed that every connected clustered graph has a c-planar connected clustered subgraph: Proceeding from the leaves to the root of the inclusion tree, construct a subgraph in which every cluster induces a spanning tree. Unfortunately, not every completely connected clustered graph has a completely connected subgraph that is c-planar: See the clustered graph (G, T, r) in Fig. 4 for an example. G is a subdivision of a $K_{3,3}$ and hence is not planar. But the clustered graph (H, T, r) is not completely connected for any proper subgraph $H \subseteq G$. In the rest of this section, we show that at least Theorem 2 can be extended to completely connected clustered graphs. This result was independently obtained by Jünger et al. [15].

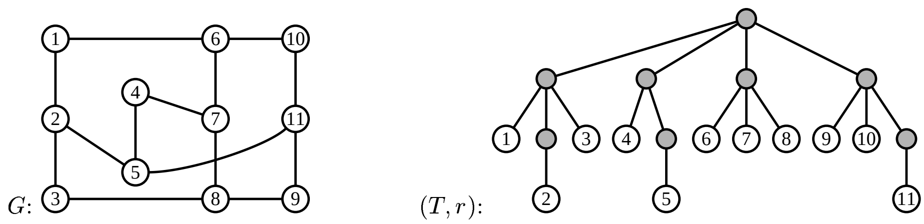

Fig. 4. A completely connected clustered graph that has no c-planar clustered subgraph which is still completely connected.

Theorem 5. *Every c-planar clustered graph is a subgraph of a c-planar completely connected clustered graph.*

Proof. Let (G, T, r) be a c-planar clustered graph with a fixed c-planar embedding. By Theorem 2, there exists a graph $G_0 \supseteq G$ such that (G_0, T, r) is c-planar and connected. Let $\nu_1, \ldots, \nu_i$ be the nodes of T in arbitrary order. We construct graphs $G_0 \subseteq \ldots \subseteq G_k$ such that

– (G_i, T, r) is c-planar and
– $G_i - V_r(\nu_i)$ is connected

We prove the existence of G_i by induction on the number k of components of $G_{i-1} - V_r(\nu_i)$. If $k = 0$ then $G_{i-1} - V_r(\nu_i)$ is connected and (G_{i-1}, T, r) is c-planar. Hence $G_i = G_{i-1}$ fulfills the required conditions.

Let $k > 1$. Let e_1, e_2 be two edges that are incident to $V_r(\nu_i)$ and to different connected components of $G_{i-1} - V_r(\nu_i)$. We may assume that e_1 and e_2 are consecutive in the cyclic order around $V_r(\nu_i)$. Let $v_1 = e_1 \cap (V \setminus V_r(\nu_i))$ and $v_2 = e_2 \cap (V \setminus V_r(\nu_i))$.

Then $G_{i-1} + \{v_1, v_2\}$ is c-planar: Let r' be the root of the smallest subtree of T containing v_1 and v_2. Since v_1 and v_2 are not contained in the same connected component of $G_i - V_r(\nu_i)$ but every cluster is already connected, the subtree rooted at r' also contains ν_i. Let $\mu \neq r'$ be the first node on the path from r' to ν_i. No edge crosses the cluster boundary of μ between e_1 and e_2. Else the area between e_1 and e_2 would contain vertices in $V_r(\mu) \setminus V_r(\nu_i)$ – contradicting that $V_r(\mu)$ induces a connected subgraph of G_{i-1}. Hence, we can route the edge $\{v_1, v_2\}$ along the edge e_1 the cluster boundary of μ and the edge e_2. By the choice of μ, this route crosses every cluster boundary at most once. Hence, we obtain a c-planar drawing of $G_{i-1} + \{v_1, v_2\}$.

The number of connected components of $(G_{i-1} + \{v_1, v_2\}) - V_r(\nu_i)$ is $k - 1$. Hence, we can apply the inductive hypothesis to finish the proof. □

The above proof describes a construction for obtaining a c-planar completely connected clustered graph by adding edges to a c-planar clustered graph. Note, however, that the number of additional edges depends on a c-planar drawing of the given graph and on the ordering of the nodes in the inclusion tree. The problem of minimizing the number of additional edges is a generalization of the $\mathcal{NP}$-complete problem planar biconnectivity augmentation [16].

6 Conclusion

We introduced completely connected clustered graphs, i.e. clustered graphs for which not only every cluster but also the complement of each cluster is connected. We made the surprising observation that every planar embedding of the underlying graph of a completely connected clustered graph is already a c-planar embedding. Only the outer face of the underlying graph has to be chosen according to the root of the inclusion tree or vice versa. Fixing the root (or the outer face), we gave a linear time algorithm that decides which faces can be chosen as the outer face (or which nodes can be chosen as the root). Finally, we showed that every c-planar clustered graph is a subgraph of a completely connected c-planar clustered graph.

Acknowledgments: We thank Ulrik Brandes for introducing to us the bottom-up top-down approach and Christian Fieß for implementing the algorithm that determines the rootable nodes.

References

1. F. Bertault and M. Miller. An algorithm for drawing compound graphs. In J. Kratochvíl, editor, *Proceedings of the 7th International Symposium on Graph Drawing*

(GD '99), volume 1731 of *Lecture Notes in Computer Science*, pages 197–204. Springer, 1999.

2. U. Brandes, S. Cornelsen, and D. Wagner. How to draw the minimum cuts of a planar graph. In J. Marks, editor, *Proceedings of the 8th International Symposium on Graph Drawing (GD 2000)*, volume 1984 of *Lecture Notes in Computer Science*, pages 103–114. Springer, 2001.
3. S. Cornelsen. *Drawing Families of Cuts in a Graph.* PhD thesis, Universität Konstanz, 2003. http://www.ub.uni-konstanz.de/kops/volltexte/2003/975/.
4. E. Dahlhaus. A linear time algorithm to recognize clustered planar graphs and its parallelization. In C. L. Lucchesi and A. V. Moura, editors, *Proceedings of the 3rd Latin American Symposium on Theoretical Informatics (LATIN '98)*, volume 1380 of *Lecture Notes in Computer Science*, pages 239–248. Springer, 1998.
5. G. Di Battista, W. Didimo, and A. Marcandalli. Planarization of clustered graphs. In M. Jünger and P. Mutzel, editors, *Proceedings of the 9th International Symposium on Graph Drawing (GD 2001)*, volume 2265 of *Lecture Notes in Computer Science*, pages 60–74. Springer, 2002.
6. P. Eades, R. F. Cohen, and Q. Feng. How to draw a planar clustered graph. In D.-Z. Du and M. Li, editors, *Proceedings of the 1st Annual International Conference on Coputing and Combinatorics (COCOON '95)*, volume 959 of *Lecture Notes in Computer Science*, pages 21–30. Springer, 1995.
7. P. Eades and Q. Feng. Multilevel visualization of clustered graphs. In North [18], pages 101–112.
8. P. Eades, Q. Feng, and X. Lin. Straight-line drawing algorithms for hierarchical graphs and clustered graphs. In North [18], pages 146–157.
9. P. Eades, Q. Feng, and H. Nagamochi. Drawing clustered graphs on an orthogonal grid. *Journal on Graph Algorithms and Applications*, 3(4):3–29, 1999.
10. P. Eades, H. Nagamochi, and Q. Feng. Straight-line drawing algorithms for hierarchical graphs and clustered graphs. Technical Report 98-03, Department of Computer Science and Software Engineering, University of Newcastle, Australia, 1998. Available at ftp://ftp.cs.newcastle.edu.au/pub/techreports/tr98-03.ps.Z.
11. Q. Feng, R. F. Cohen, and P. Eades. Planarity for clustered graphs. In P. Spirakis, editor, *Proceedings of the 3rd European Symposium on Algorithms (ESA '95)*, volume 979 of *Lecture Notes in Computer Science*, pages 213–226. Springer, 1995.
12. M. T. Goodrich and S. G. Kobourov, editors. *Proceedings of the 10th International Symposium on Graph Drawing (GD 2002)*, volume 2528 of *Lecture Notes in Computer Science*. Springer, 2002.
13. C. Gutwenger, M. Jünger, S. Leipert, P. Mutzel, M. Percan, and R. Weiskircher. Advances in c-planarity testing of clustered graphs. In Goodrich and Kobourov [12], pages 220–235.
14. M. L. Huang and P. Eades. A fully animated interactive system for clustering and navigating huge graphs. In S. H. Whitesides, editor, *Proceedings of the 6th International Symposium on Graph Drawing (GD '98)*, volume 1547 of *Lecture Notes in Computer Science*, pages 374–383. Springer, 1998.
15. M. Jünger, S. Leipert, and M. Percan. Triangulating clustered graphs. Technical Report zaik2002-444, Zentrum für Angewandte Informatik Köln, 2002. Available at http://www.zaik.uni-koeln.de/%7Epaper/preprints.html?show=zaik2002-444.
16. G. Kant and H. L. Bodlaender. Planar graph augmentation problems. In F. Dehne, J.-R. Sack, and N. Santoro, editors, *Algorithms and Data Structures, 2nd Workshop (WADS '91)*, volume 519 of *Lecture Notes in Computer Science*, pages 286–298. Springer, 1991.

17. D. Lütke-Hüttmann. Knickminimales Zeichnen 4-planarer Clustergraphen. Master's thesis, Universität des Saarlandes, 1999. (Diplomarbeit).
18. S. C. North, editor. *Proceedings of the 4th International Symposium on Graph Drawing (GD '96)*, volume 1190 of *Lecture Notes in Computer Science*. Springer, 1996.
19. M. Raitner. HGV: A library for hierarchies, graphs, and views. In Goodrich and Kobourov [12], pages 236–243.
20. K. Sugiyama and K. Misue. Visualization of structural information: Automativ drawing of compound digraphs. *IEEE Transactions on Systems, Man and Cybernetics*, 21:876–892, 1991.
21. X. Wang and I. Miyamoto. Generating customized layouts. In F. J. Brandenburg, editor, *Proceedings of the 3rd International Symposium on Graph Drawing (GD '95)*, volume 1027 of *Lecture Notes in Computer Science*, pages 504–515. Springer, 1996.