How to Draw the Minimum Cuts of a Planar Graph
(Extended Abstract)

Ulrik Brandes, Sabine Cornelsen, and Dorothea Wagner

University of Konstanz, Department of Computer and Information Science
{Ulrik.Brandes,Sabine.Cornelsen,Dorothea.Wagner}@uni-konstanz.de

Abstract. We show how to utilize the cactus representation of all minimum cuts of a graph to visualize the minimum cuts of a planar graph in a planar drawing. In a first approach the cactus is transformed into a hierarchical clustering of the graph that contains complete information on all the minimum cuts. We present an algorithm for c-planar orthogonal drawings of hierarchically clustered planar graphs with rectangularly shaped cluster boundaries and the minimum number of bends. This approach is then extended to drawings in which the two vertex subsets of every minimum cut are separated by a simple closed curve.

1 Introduction

The edge connectivity is a fundamental structural property of a graph. Dinitz et al. [1] discovered that the set of all minimum cuts of a connected graph G with positive edge weights has a tree-like structure. It can be represented by a cactus, i.e. by a connected graph in which every edge is contained in at most one cycle. Although the number of minimum cuts in a graph is in $\mathcal{O}(n^2)$, the size of the cactus is linear in the number n of vertices of G. From the cactus representation, the bipartitions of the vertex sets can easily be extracted, but it contains almost no information about the edges in G. We want to visualize a graph G together with the cactus representation of its minimum cuts in one drawing.

A simple closed curve divides the plane into two connected components. A minimum cut divides the set of vertices of a graph into two connected subsets. Thus, it is natural to visualize a minimum cut in a drawing of a graph by a simple closed curve separating the two subsets.

Feng et al. [4] introduced the model of hierarchically clustered graphs. In a drawing of a hierarchically clustered graph, a set of vertices of a graph is represented by a region that is bounded by a simple closed curve. The set of subsets of the vertex set that is represented simultaneously in this way has to have tree structure. In terms of cuts, this means that we can represent a set of pairwise non-crossing cuts as a hierarchically clustered graph. Graphs having no crossing minimum cuts are, for example, maximal planar graphs and chordal graphs.

If there are crossing cuts, the structure of the set of minimum cuts implies that they are represented at least implicitly in a drawing of the pairwise

non-crossing cuts. We show, however, that the model of hierarchically clustered graphs can be extended to cactus-clustered graphs such that the goal of visualizing every minimum cut by a simple closed curve is achieved. This extension is mainly based on the fact that for two crossing minimum cuts, the four corner cuts are also minimum.

The contribution of this paper is as follows. In Sect. 2 and 3, we provide some background on the cactus representation and on hierarchically clustered graphs, respectively. In Sect. 4, we show how to construct a hierarchically clustered graph from a cactus representation and prove an interesting property of crossing cuts in planar graphs that enables us to represent all minimum cuts by closed curves. Finally, our two methods for drawing planar graphs that are clustered according to their minimum cuts are presented in Sect. 5.

2 The Cactus of the Set of Minimum Cuts

Let G be an undirected connected graph. With $E(G)$ we denote the set of edges of G and with $V(G)$ the set of vertices of G. A graph G together with a positive weight function $\omega : E(G) \to \mathbb{R}^+$ is a *weighted graph*. For two subsets S and T of $V(G)$, let $E(S,T) := \{\{v,w\}; v \in S \text{ and } w \in T\}$ be the set of edges between S and T, and let $\omega(S,T) := \sum_{e \in E(S,T)} \omega(e)$ be the sum of weights of the edges between the two subsets.

A *cut* is an unordered pair $\{S, \overline{S}\}$ where $\emptyset \subsetneq S \subsetneq V(G)$ and $\overline{S} := V(G) \setminus S$. A set S *induces* the cut $\{S, \overline{S}\}$. The *weight* of this cut is $\omega(S, \overline{S})$. With $\lambda := \min_{\emptyset \subsetneq S \subsetneq V(G)} \omega(S, \overline{S})$ we denote the minimum of all these weights and a cut $\{S, \overline{S}\}$ of G satisfying $\omega(S, \overline{S}) = \lambda$ is called a *minimum cut*. With $\mathcal{C}(G)$ we denote the set of minimum cuts of G. By $G(S)$ we denote the subgraph of G induced by a set S.

A cycle $c : v_1, \ldots, v_k$ is a sequence of $k \geq 3$ distinct vertices, such that $E(c) := \{\{v_1, v_2\}, \ldots, \{v_{k-1}, v_k\}, \{v_k, v_1\}\} \subset E(G)$. For a subset $E' \subset E(G)$, we denote by $G - E'$ the graph $(V(G), E(G) \setminus E')$.

Definition 1. *A* representation *for a set $\mathcal{C}$ of cuts of a graph G is a pair $(\mathcal{G}, \varphi)$ such that $\mathcal{G}$ is a weighted graph and $\varphi : V(G) \longrightarrow V(\mathcal{G})$ is a mapping such that $\mathcal{C} = \varphi^{-1}(\mathcal{C}(\mathcal{G})) := \{\{\varphi^{-1}(S), \varphi^{-1}(\overline{S})\}; \{S, \overline{S}\} \in \mathcal{C}(\mathcal{G})\}$. A node $\nu \in V(\mathcal{G})$ is called* empty *if $\varphi^{-1}(\nu) = \emptyset$.*

Definition 2. *Two cuts $\{S, \overline{S}\}$ and $\{T, \overline{T}\}$ are* crossing*, if none of the* corner sets *$S \cap T$, $S \cap \overline{T}$, $\overline{S} \cap T$, and $\overline{S} \cap \overline{T}$ is empty. A cut induced by a corner set is a* corner cut *and the cut induced by $S \Delta T := S \setminus T \cup T \setminus S$ is the* diagonal cut*.*

A cut is a *crossing cut* of a family $\mathcal{C}$ of cuts, if it crosses any cut in $\mathcal{C}$. If $\mathcal{C}(G)$ contains no crossing cuts, the set of minimum cuts of G can be represented by a tree. Dinitz et al. [1] showed that the set of minimum cuts of an arbitrary weighted connected graph can be represented by a cactus where cycles correspond to sets of crossing cuts. More precisely:

Definition 3 (Cactus). *A* cactus *is a connected graph in which every edge belongs to at most one cycle. An edge that belongs to no cycle is called a* tree edge. *An edge that belongs to one cycle is called a* cycle edge.

In what follows, we assume that a weighted cactus is *uniform*, i.e. that all cycle edges have the same weight and that every tree edge has twice the weight of a cycle edge.

Theorem 1 ([1]). *The set $\mathcal{C}(G)$ of all minimum cuts of a weighted connected graph G has a representation $(\mathcal{G}, \varphi)$ such that $\mathcal{G}$ is a uniform cactus with $\mathcal{O}(n)$ nodes.*

Figure 1 shows an example of a weighted graph and its cactus. Dinitz and Nutov

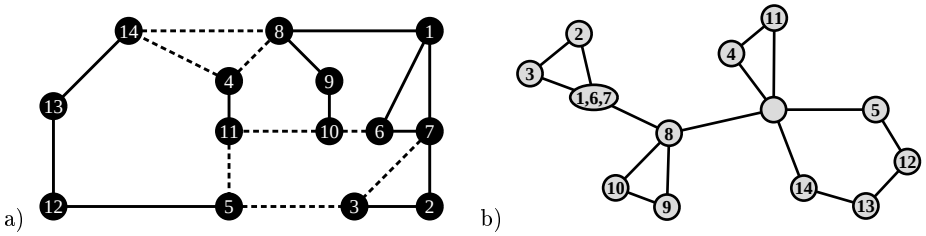


Fig. 1. a) A weighted connected graph and b) the cactus representation of its minimum cuts. In a), solid edges have weight 2 and dashed edges have weight 1. In b), φ is represented by the labels of the nodes.

[2] characterized all sets of cuts that can be represented by a cactus.

Theorem 2. *A set $\mathcal{C}$ of cuts can be represented by a cactus if and only if for any two crossing cuts in $\mathcal{C}$*

- *the four corner cuts are in $\mathcal{C}$ and*
- *the diagonal cut is not in $\mathcal{C}$.*

In this case, there is always a cactus representation with $O(n)$ nodes.

In what follows, we denote by $\mathcal{G}$ the cactus representation of all minimum cuts of G. Note that there is a bijection between the set of minimum cuts of $\mathcal{G}$ and the set of tree edges and pairs of cycle edges belonging to the same cycle. Thus, we can also say that a cut in G is represented by a tree edge or a pair of cycle edges of $\mathcal{G}$. An important property of a cactus representation that we will use later is, that "edges in G do not cross a cycle of $\mathcal{G}$". More precisely:

Lemma 1. *For a cycle $c : \nu_1, \ldots, \nu_k$ in $\mathcal{G}$ let $\mathcal{V}_i, i = 1, \ldots, k$ be the set of vertices in the connected component of $\mathcal{G} - E(c)$ that contains ν_i and $V_i := \varphi^{-1}(\mathcal{V}_i)$. Then*

$$\omega(V_i, V_j) = 0 \text{ if } i - j \not\equiv \pm 1 \bmod k.$$

Fleischer [5] showed that the cactus of all minimum cuts of a weighted graph can be constructed in $\mathcal{O}(mn \log \frac{n^2}{m})$ time. For an unweighted graph, it can be computed in $\mathcal{O}(\lambda n^2)$ time [12]. Using the linear-time shortest-path algorithm of Henzinger et al. [7] for max-flow computations, the cactus of a weighted planar graph can be obtained in $\mathcal{O}(n^2)$ time with the construction described in [5].

3 Hierarchically Clustered Graphs

Feng et al. [4] introduced the hierarchically clustered graph model and characterized graphs that have a planar drawing with respect to the clustering. In this section, we summarize definitions and results of [4] and [3] that we will use later.

A *hierarchically clustered graph* (G, T) consists of a graph $G = (V, E)$ and a rooted tree T such that the set of leaves of T is exactly V. Vertices of T are called *nodes*. Each node ν of T represents the *cluster* $V(\nu)$ of leaves in the subtree of T rooted at ν. T is called the *inclusion tree* of (G, T). An edge e of G is said to be *incident* to a cluster $V(\nu)$, if $|e \cap V(\nu)| = 1$.

A hierarchically clustered graph (G, T) is *connected*, if each cluster induces a connected subgraph of G.

A *drawing* $\mathcal{D}$ of a hierarchically clustered graph (G, T) includes the drawing of the underlying graph G and of the inclusion tree T in the plane. Each vertex v of G is represented as a point $\mathcal{D}(v)$ and each edge $e = \{v, w\}$ as a simple curve $\mathcal{D}(e)$ between $\mathcal{D}(v)$ and $\mathcal{D}(w)$. Each non-leaf node ν of T is drawn as a simple closed region $\mathcal{D}(\nu)$ bounded by a simple closed curve $\partial\mathcal{D}(\nu)$ such that

- $\mathcal{D}(\mu) \subset \mathcal{D}(\nu)$ for all descendents μ of ν.
- $\mathcal{D}(\mu) \cap \mathcal{D}(\nu) = \phi$ if μ is neither a descendent nor an ancestor of ν.
- $\mathcal{D}(e) \subset \mathcal{D}(\nu)$ for all edges e of G with $e \subset V(\nu)$.
- $\partial\mathcal{D}(e) \cap \partial\mathcal{D}(\nu)$ is a single point if $|e \cap V(\nu)| = 1$.

Roughly speaking, T is drawn in the inclusion representation and edges of G may only cross cluster boundaries if necessary.

The drawings of an edge e and a cluster ν have an *edge-cluster-crossing*, if $e \cap V(\nu) = \phi$ but $\mathcal{D}(e) \cap \mathcal{D}(\nu) \neq \phi$. A drawing of a hierarchically clustered graph is *c-planar*, if there are no crossing edges and no edge-cluster-crossings. A graph is *c-planar* if it has a c-planar drawing.

Theorem 3 (Characterization of c-Planar Graphs [4]). *A connected hierarchically clustered graph $C = (G, T)$ is c-planar if and only if there exists a planar drawing of G, such that for each node ν of T all vertices of $V - V(\nu)$ are in the outer face of the drawing of $G(\nu)$.*

In an *OGRC (orthogonal grid rectangular cluster) drawing* of a hierarchically clustered graph (G, T), curve $\mathcal{D}(e)$ is a sequence of horizontal and vertical segments for every edge e of G and $\mathcal{D}(\nu)$ is an axis-parallel rectangle for every non-leaf node ν of T.

Theorem 4 ([3]). *For a c-planar connected clustered graph with n vertices of degree at most 4, a c-planar OGRC drawing with $\mathcal{O}(n^2)$ area and with at most 3 bends per edge can be constructed in $\mathcal{O}(n)$ time.*

4 From the Cactus Representation to a Hierarchically Clustered Graph

Both, the cactus representation of the minimum cuts of a graph and the inclusion tree of a hierarchically clustered graph, represent structural information of a graph. We show how to transform the cactus representation into an inclusion tree such that all minimum cuts can be recognized in a drawing of the corresponding hierarchically clustered graph.

Let $(\mathcal{G}, \varphi)$ be the cactus structure of the minimum cuts of a weighted connected planar graph G with n vertices.

1. For every cycle $c : \nu_1, \ldots, \nu_k$ in $\mathcal{G}$, delete all edges in c and add a new (empty) node ν_c and edges $\{\nu_i, \nu_c\}$, $i = 1, \ldots, k$.
2. Replace every empty node of degree 2 and its incident edges by a single edge.
3. For every vertex v of G, add a new node ν_v and an edge $\{\varphi(v), \nu_v\}$.
4. Find a suitable root.

We call the thus constructed rooted tree $\mathcal{T}(G)$. Note that $(G, \mathcal{T}(G))$ is now a hierarchically clustered graph. The number of nodes in $\mathcal{T}(G)$ is in $\mathcal{O}(n)$: By Theorem 1, we have $|V(\mathcal{G})| \in \mathcal{O}(n)$. In Step 1, we add a new node for every cycle in $\mathcal{G}$ and in Step 3, we add n new nodes. Thus $|V(\mathcal{T}(G))|$ remains in $\mathcal{O}(n)$.

Figure 2 shows the inclusion tree $\mathcal{T}(G)$ of the graph G from Fig. 1. There are

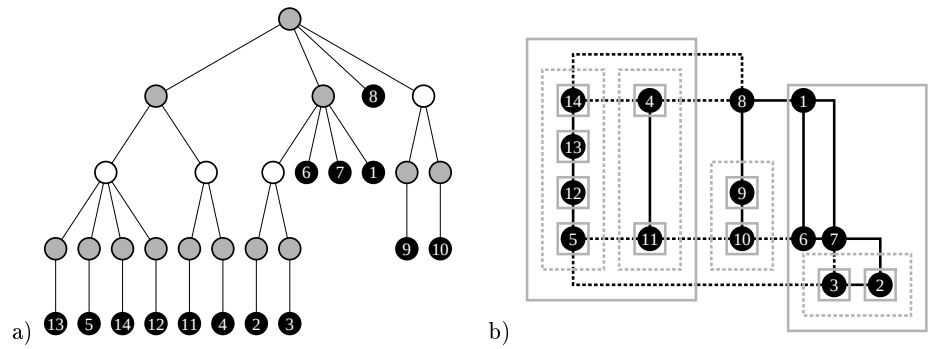


Fig. 2. a) White nodes in the inclusion tree $\mathcal{T}(G)$ of the graph G in Fig. 1 represent the nodes that were added for a cycle in $\mathcal{G}$. b) The corresponding cluster boundaries are drawn as dashed grey rectangles in the c-planar OGRC drawing of $(G, \mathcal{T}(G))$.

several options for choosing a root. We have chosen the root such that $|V(\nu)| \leq |\overline{V(\nu)}|$ for every inner node ν of $\mathcal{T}(G)$. This has the advantage that the most balanced minimum cut $\{S, \overline{S}\}$, i.e. the cut such that $||S| - |\overline{S}||$ is minimal, is seen on the top level. Another possibility is to take the center of the tree, i.e. to minimize the height. In both cases, the root can be computed in linear time.

To visualize the minimum cuts in G we want to construct a c-planar drawing of the hierarchically clustered graph $(G, \mathcal{T}(G))$. The next lemma guarantees that we can fix an arbitrary embedding of G and add cluster boundaries.

Lemma 2. *Every planar embedding of a weighted connected planar graph G can be extended to a c-planar drawing of the hierarchically clustered graph $(G, \mathcal{T}(G))$.*

Proof. To see this, recall the following facts about minimum cuts.

Remark 1. Let $\{S, \overline{S}\}$ be a minimum cut in a weighted connected graph. Then both $G(S)$ and $G(\overline{S})$ are connected.

Remark 2. Let $\{S, \overline{S}\}$ be a minimum cut in a weighted connected planar graph G. In any embedding of G, the dual of the graph induced by $E(S, \overline{S})$ is a cycle.

These two facts guarantee that for every embedding of the weighted connected planar graph G the hierarchically clustered graph $(G, \mathcal{T}(G))$ fulfills the preconditions of Theorem 3 and thus has a c-planar drawing. □

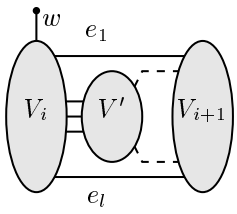
In Step 1, we replace each cycle of the cactus by a star. Thus, the information about the cyclic order of the edges in a cycle of $\mathcal{G}$ is not preserved in $\mathcal{T}(G)$. However, this order can be reconstructed from a c-planar drawing of $(G, \mathcal{T}(G))$ as follows.

Let $c : \nu_1, \ldots, \nu_k$ be a cycle in $\mathcal{G}$ and let $\mathcal{V}_i$ and V_i be defined as in Lemma 1. Consider the dual of the graph induced by $E(V_i, \overline{V_i})$. Since $\{V_i, \overline{V_i}\}$ is a minimum cut, its dual is a cycle, denoted by d_i^*. Note that the cycles d_i^* correspond to cluster boundaries in a c-planar drawing of $(G, \mathcal{T}(G))$. For each edge e in G let $\omega(e^*) = \omega(e)$. An immediate consequence of Lemma 1 is

Lemma 3. $\sum_{e^* \in E(d_i^*) \cap E(d_j^*)} \omega(e^*) = 0 \text{ if } i - j \not\equiv \pm 1 \bmod k.$

Lemma 4. *The set of edges $E(d_i^*) \cap E(d_{i+1}^*)$ is consecutive in d_i^* and d_{i+1}^*.*

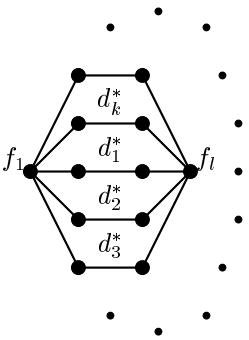
Proof. Suppose not. Let $e_1^*, \ldots, e_l^*$ be the sequence of edges in a path in d_i^* such that $e_1^*, e_l^* \in E(d_{i+1}^*)$, $e_2^*, \ldots, e_{l-1}^* \notin E(d_{i+1}^*)$. Let $e^* \in E(d_i^*) \setminus (E(d_{i+1}^*) \cup \{e_2^*, \ldots, e_{l-1}^*\})$ be another edge in d_i^* and let $e = \{v, w\}$ with $w \notin V_i$. For $j = i, i+1$ let p_j be a path from $e_1 \cap V_j$ to $e_l \cap V_j$ in the graph induced by V_j. Let c' be the cycle in G that is induced by edge e_1, path p_i, edge e_l and path p_{i+1}. Without loss of generality we can assume that edges $e_2^*, \ldots, e_{l-1}^*$ are inside cycle c'. Let $V' \subset V(G)$ be the set of vertices that are incident to $e_2, \ldots, e_{l-1}$ and that are not in V_i. Then, by Lemma 1, $w \in V_{i-1}$ and $V' \subset V_{i-1}$. Thus, V_{i-1} is not connected, contradicting Remark 1. □



Lemma 5. $\left| \bigcap_{i=1}^{k} V(d_i^*) \right| = 2$

Proof. Let $d_1^* : f_1, \ldots, f_l, f_{l+1}, \ldots, f_r$ and $d_2^* : f_1, \ldots, f_l, f'_{l+1}, \ldots, f'_s$ be such that $f_{l+1} \neq f'_{l+1}$ and $f_r \neq f'_s$. From Lemma 3 we conclude that $E(d_3^*) \cap E(d_2^*) = E(d_2^*) \setminus E(d_1^*)$ and with Lemma 4 we get $\cap_{i=1}^{3} V(d_i^*) = \{f_1, f_l\}$. It follows inductively that $\cap_{i=1}^{k} V(d_i^*) = \{f_1, f_l\}$. □

Thus, we can choose one face f where all the cycles in G^* that correspond to the cuts of a cycle c in $\mathcal{G}$ intersect. The cyclic order in f displays the cyclic order of c. Moreover, every cut that is represented by two edges of c can be reconstructed from a c-planar drawing of $(G, \mathcal{T}(G))$ in the following way. Divide every cluster boundary corresponding to a cycle d_i^* into two paths between f_1 and f_l. This results in k disjoint paths. The cycle consisting of any two of these paths is the dual of a cut represented by two edges of c and every cut that is represented by a two-cut of c is of this form.



5 Two Models for Drawing Cuts

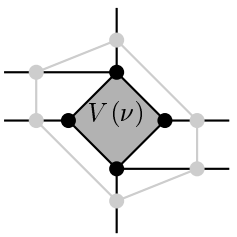
5.1 C-Planar Drawings of Hierarchically Clustered Graphs

As mentioned in Sect. 3, Eades et al. [3] introduced a method for drawing hierarchically clustered planar graphs orthogonally with rectangularly shaped cluster boundaries. To do this, they made the undirected graph directed and allowed edges to cross the cluster boundary only at the top or bottom of the rectangle. This might introduce unnecessary bends into the drawing. We introduce a different way of drawing a connected c-planar graph (G, T). We add edges and vertices to G such that the newly constructed graph G' remains planar and each cluster boundary corresponds to a cycle in G'. Now any embedding preserving algorithm can be applied to draw graph G' and thus to obtain a c-planar drawing of (G, T). In case G has maximum degree 4, using the model of Tamassia [13] with some additional constraints on the flow, this leads to a c-planar OGRC drawing with the minimum number of bends.[1]

Let (G, T) be a hierarchically clustered c-planar graph with an embedding in the plane that fulfills the conditions of Theorem 3. Recall, that for $T = \mathcal{T}(G)$, by Lemma 2, every embedding of a planar graph is suitable.

For every cluster, we add a cycle of new edges and new vertices to G in the following way:

Proceeding from the leaves to the root of T, for every non-leaf node ν of T let $e_1, \ldots, e_k$ be the edges incident to cluster $V(\nu)$ in their cyclic order around $V(\nu)$. Let $e_{k+1} = e_1$ and $e_i = \{v_i, w_i\}, i = 1, \ldots, k+1$. For $i = 1, \ldots, k$, we *split* edge e_i, i.e. we add a vertex v_{e_i} to $V(G)$ and replace edge e_i by edges $\{v_i, v_{e_i}\}$ and $\{w_i, v_{e_i}\}$. Finally, we add edges $\{v_{e_i}, v_{e_{i+1}}\}$. These k edges are called *boundary edges* of $V(\nu)$. They form a cycle, called the *boundary cycle* of $V(\nu)$, that model the cluster boundary of $V(\nu)$.



[1] While implementing the extension of Tamassia's model to hierarchically clustered graphs we learned that it was independently described in [9] and is now part of the AGD library [11].

A special case occurs, if there are only one or two edges incident to a cluster. In that case, two or one additional vertices are inserted in this cycle to avoid loops and multiple edges. The resulting edges are also called boundary edges. Let the resulting graph be G'. Let $n = |V(G)|$ and $\mathrm{h}(T)$ be the height of the inclusion tree T.

Lemma 6. $|V(G')| \in \mathcal{O}(n \cdot \mathrm{h}(T))$

Proof. Let $e = \{u, v\} \in E(G)$ and let k be the number of vertices on the path in T between u and v. Then $k - 3 \leq 2 \cdot \mathrm{h}(T)$ vertices are inserted into e. Thus, $|V(G')| \leq n + 2|E(G)| \cdot \mathrm{h}(T) \in \mathcal{O}(n \cdot \mathrm{h}(T))$. □

Note that in case $T = \mathcal{T}(G)$ and $\omega(e) \geq 1$ for every edge $e \in E(G)$ it is also true, that $|V(G')| \in \mathcal{O}(\lambda \cdot n)$: Every cluster is incident to at most λ edges and the number of clusters is in $\mathcal{O}(n)$. For unweighted planar graphs we have $\lambda \leq 5$ and this implies $|V(G')| \in \mathcal{O}(n)$.

Lemma 7. *G' can be constructed in $\mathcal{O}(|V(G')|)$ time.*

Proof. Proceeding for each edge $\{u, v\} \in E(G)$ along the path in T between u and v, splitting the edges can be done in $\mathcal{O}(|E(G)| + |\text{added vertices}|) = \mathcal{O}(|V(G')|)$.

From the leave to the root of T, add the boundary edges along the outer face of each cluster. Doing this, every edge can be touched at most twice.Thus, inserting the boundary edges is in $\mathcal{O}(|E(G')|) = \mathcal{O}(|V(G')|)$. □

In the flow network for an orthogonal [13] or quasi-orthogonal drawing [14, 8] of G', we restrict the flow over a boundary edge to be zero, if it goes from outside the corresponding boundary cycle into it. This guarantees that the boundary cycles are rectangularly shaped in any resulting orthogonal drawing. This restriction is necessary. Even in the case of unweighted graphs with the root chosen in such a way, that $|V(\nu)| \leq |\overline{V(\nu)}|$, there are examples of planar graphs G such that the bend minimum solution without restriction of the clustered graph $(G, \mathcal{T}(G))$ have non-rectangularly shaped cluster boundaries. See for example Fig. 3. Theorem 4 guarantees that there is a feasible flow for the restricted flow network. The resulting drawing is a bend minimum c-planar OGRC drawing. Moreover, all inserted vertices have degree 4 and split edges alternate with boundary edges. Thus, the corresponding original edges in G have no bends at cluster boundaries.

Lemma 8. *The area requirement of the constructed bend–minimum c–planar OGRC drawing of (G, T) is in $\mathcal{O}(n^2)$.*

Proof. There are $\mathcal{O}(n)$ clusters and each cluster boundary requires two horizontal and two vertical lines.

Those edges in G' that are not boundary edges correspond to $\mathcal{O}(n)$ original edges in G. As the constructed drawing is bend–minimum, by Theorem 4, there are at most $3 \cdot |E(G)|$ bends on those edges. Thus, the non–boundary edges require at most $4 \cdot |E(G)| \in \mathcal{O}(n)$ horizontal and vertical lines. □

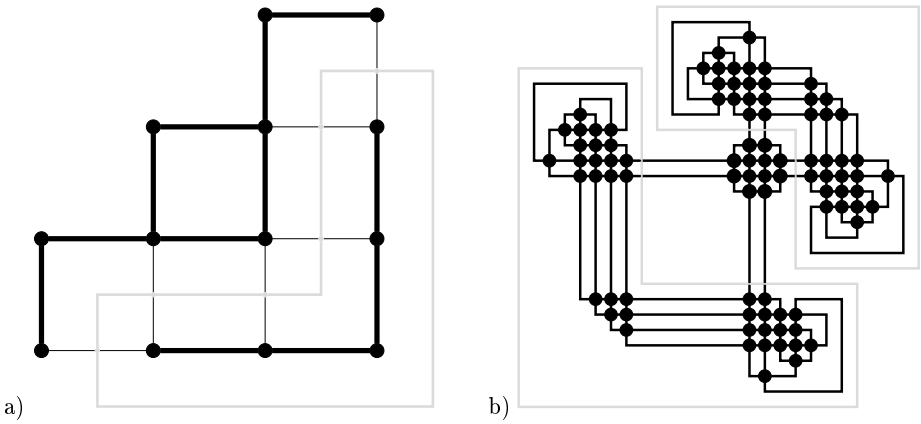


Fig. 3. Drawings of the non-trivial minimum cuts of a) a weighted and b) an unweighted graph without rectangularity restriction on the cluster shape. Grey edges are boundary edges, thin edges have weight 1 and thick edges have weight 6.

The running time of our algorithm is as follows:

- Constructing the cactus $\mathcal{G}$ of a planar connected weighted graph G is in $\mathcal{O}(n^2)$.
- Constructing the inclusion tree $\mathcal{T}(G)$ from the cactus is in $\mathcal{O}(n)$.
- Constructing G' from (G, T) is in $\mathcal{O}(n \cdot \mathrm{h}(T))$.
- Constructing the orthogonal drawing of G' with $N := |V(G')|$ vertices is in $\mathcal{O}(N^{7/4}\sqrt{\log N})$ [6].

We can finally summarize that the running time is dominated by the construction of the cactus and the orthogonal drawing and is in $\mathcal{O}(n^2 + N^{7/4}\sqrt{\log N})$ time. Figure 2 shows an OGRC drawing of the hierarchically clustered graph $(G, \mathcal{T}(G))$ where G is the graph in Fig. 1.

5.2 Planar Cactus-Clustered Drawings

The information about the cyclic order of the edges in a cycle of $\mathcal{G}$ is preserved in a c-planar drawing of $(G, \mathcal{T}(G))$. However, crossing cuts are given only implicitly and have to be reconstructed from the others as mentioned at the end of Sect. 4. In this subsection we show how we can modify the c-planar drawing in such a way that on one hand, for every minimum cut $\{S, \overline{S}\}$ there is a cycle c added to G separating S from $\overline{S}$ and, on the other hand, for every cycle c added to G the cut defined by the inside and the outside of c is a minimum cut. We achieve this, roughly speaking, by merging the cluster boundaries corresponding to pairs of incident nodes on a cycle in the cactus.

Let $\mathcal{T}(G)$ be defined as in Sect. 4, except that we do not replace nodes of degree 2. Let again $c : \nu_1, \ldots, \nu_k$ be a cycle in $\mathcal{G}$ and let $\mathcal{V}_i$ and V_i be defined as in Lemma 1. Let ν_c be the node added for c in $\mathcal{T}(G)$. Note that $\overline{V_j} = V(\nu_c)$ and

$V(\nu_j) \notin \{V_j, \overline{V_j}\}$ if ν_j is the predecessor of ν_c. In this case, node ν_c represents cut $\{V_j, \overline{V_j}\}$ and in the following, we substitute ν_j by ν_c in c. Now, every node in $\mathcal{T}(G)$ is associated with at most one cycle in $\mathcal{G}$.

By Lemma 4, we already know that duals of the edges between V_i and V_{i+1} are consecutive on the cycles corresponding to their cluster boundaries. We now show that those vertices that were inserted into an edge of G for a drawing of these duals are adjacent.

Lemma 9. *Let $e \in E(V_i, V_{i+1})$ and let v_i and v_{i+1} be the vertices that were inserted into e for the cluster corresponding to ν_i and ν_{i+1}, respectively. Then $\{v_i, v_{i+1}\} \in E(G')$.*

Proof. Let μ be the root of the smallest subtree of $\mathcal{T}(G)$ containing ν_i and ν_{i+1}. If $\nu_c = \mu$, the vertices corresponding to the two descendents ν_i and ν_{i+1} of μ are inserted consecutively in e. If not, one of the two nodes, say ν_{i+1}, was the node that was substituted by ν_c. The vertices corresponding to ν_i and ν_c are inserted consecutively in e. □

Thus, we can merge vertices that were inserted into the same edge for clusters corresponding to nodes of the same cycle. Finally, for each cycle c, we add a vertex in both of the faces f_1 and f_l identified in Lemma 5 and we split each edge that is incident to one of the faces and is a boundary edge of a cluster corresponding to c at the added vertex. This results in k paths between f_1 and f_l.

Let the constructed graph be G''. As the number of cycles in $\mathcal{G}$ is in $\mathcal{O}(n)$, we add $\mathcal{O}(n)$ vertices to G'. Thus, $|V(G'')| \in \mathcal{O}(|V(G')|)$ and G'' can be constructed in $\mathcal{O}(|V(G')|$ time.

As in the previous subsection, we can now apply any embedding preserving algorithm to draw graph G'' and thus to visualize all minimum cuts in G. We call the resulting drawing a planar *cactus-clustered* drawing.[2] An example using the quasi-orthogonal drawing method is shown in Fig. 4. The advantage of this drawing is obviously that every minimum cut is represented by a simple closed curve. Moreover, parallel boundary edges are avoided.

[2] In analogy to c-planar drawings of hierarchically clustered graphs, we define a *planar cactus-clustered drawing* $\mathcal{D}$ of a planar graph G with a cactus representation of a set $\mathcal{C}$ of cuts as follows. Again, each vertex of G is represented as a distinct point and each edge $\{v, w\}$ as a simple curve between $\mathcal{D}(v)$ and $\mathcal{D}(w)$ such that no two edges cross. Each cut $C = \{S, \overline{S}\} \in \mathcal{C}$ is represented by a simple closed curve such that (1) $\mathcal{D}(S)$ and $\mathcal{D}(\overline{S})$ are in different connected components of $\mathbb{R}^2 \setminus \mathcal{D}(C)$ and such that (2) for every simple closed curve c in $\bigcup_{C \in \mathcal{C}} \mathcal{D}(C)$ there is a cut $\{T, \overline{T}\} \in \mathcal{C}$ such that (a) $\mathcal{D}(T)$ and $\mathcal{D}(\overline{T})$ are contained in different connected components of $\mathbb{R}^2 \setminus c$ and (b) for an edge $e \in E$ we have $|\mathcal{D}(e) \cap c| = 1$, if e is incident to both, a vertex in T and a vertex in $\overline{T}$ and $\mathcal{D}(e) \cap \mathcal{D}(C) = \emptyset$, if not.

Moreover, for two cuts $C, C' \in \mathcal{C}$ it shall hold that $|\mathcal{D}(C) \cap \mathcal{D}(C')| \neq 1$.

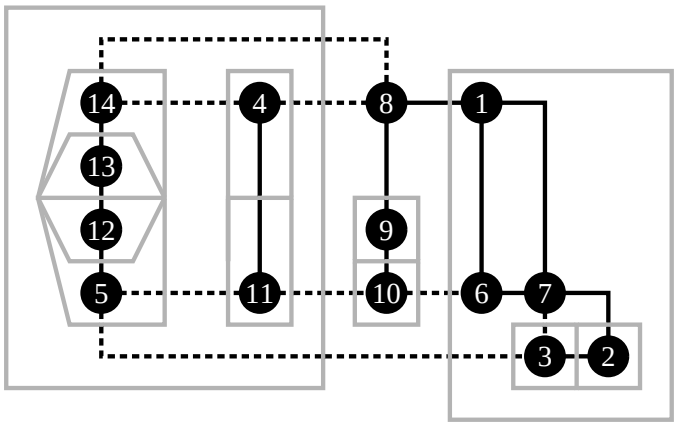


Fig. 4. A cactus-clustered drawing of the graph in Fig. 1.

6 Conclusion

We outlined two methods for representing the minimum cuts of a weighted planar graph in a planar drawing of the graph. Utilizing the cactus representation, all minimum cuts can be shown in a c-planar drawing of a hierarchical clustering of the graph. This approach was then extended to cactus-clustered drawings that visualize all minimum cuts by simple closed curves. Both approaches have been demonstrated to work for bend-minimum orthogonal drawings, but can be used with any drawing algorithm that preserves the embedding of cluster boundaries.

Moreover, our methods can be extended to any set $\mathcal{C}$ of, not necessarily minimum, cuts of a planar graph G that has a cactus representation $(\mathcal{G}, \varphi)$ and the following additional properties:

1. For each cut $\{S, \overline{S}\} \in \mathcal{C}$, the graphs $G(S)$ and $G(\overline{S})$ are connected.
2. No edge of G crosses a cycle of $\mathcal{G}$.

If $\mathcal{T}(\mathcal{G})$ is the inclusion tree constructed from $\mathcal{G}$ as described in Sect. 4, it holds that $(G, \mathcal{G}, \varphi)$ has a planar cactus-clustered drawing if and only if $(G, \mathcal{T}(\mathcal{G}))$ is c-planar.

Acknowledgments. We thank Christian Fieß for providing part of the implementation used to compute the cactus, determine a clustering, and draw the clustered graphs. Our implementation uses LEDA [10] and an older version of the AGD library [11].

References

1. Y. Dinitz, A. V. Karzanov, and M. Lomonosov. On the structure of a family of minimal weighted cuts in a graph. In A. Fridman, editor, *Studies in Discrete Optimization*, pages 290–306. Nauka, 1976. (in Russian).

2. Y. Dinitz and Z. Nutov. A 2-level cactus model for the system of minimum and minimum+1 edge–cuts in a graph and its incremental maintenance. In *Proceedings of the 27th Annual ACM Symposium on the Theory of Computing (STOC '95)*, pages 509–518. ACM, The Association for Computing Machinery, 1995.
3. 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.
4. 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.
5. L. Fleischer. Building chain and cactus representations of all minimum cuts from Hao–Orlin in the same asymptotic run time. *Journal of Algorithms*, 33(1):51–72, 1999.
6. A. Garg and R. Tamassia. A new minimum cost flow algorithm with applications to graph drawing. In S. C. North, editor, *Proceedings of the 4th International Symposium on Graph Drawing (GD '96)*, volume 1190 of *Lecture Notes in Computer Science*, pages 201–213. Springer, 1996.
7. M. R. Henzinger, P. Klein, S. Rao, and S. Subramanian. Faster shortest-path algorithms for planar graphs. *Journal of Computer and System Sciences*, 55:3–23, 1997. Special Issue on Selected Papers from STOC 1994.
8. G. W. Klau and P. Mutzel. Quasi orthogonal drawing of planar graphs. Technical Report MPI-I-98-1-031, Max-Planck-Institut für Informatik, Saarbrücken, Germany, 1998.
9. D. Lütke-Hüttmann. Knickminimales Zeichnen 4-planarer Clustergraphen. Master's thesis, Universität des Saarlandes, 1999. (Diplomarbeit).
10. K. Mehlhorn and S. Näher. *The* LEDA *Platform of Combinatorial and Geometric Computing*. Cambridge University Press, 1999. Project home page at `http://www.mpi-sb.mpg.de/LEDA/`.
11. P. Mutzel, C. Gutwenger, R. Brockenauer, S. Fialko, G. W. Klau, M. Krüger, T. Ziegler, S. Näher, D. Alberts, D. Ambras, G. Koch, M. Jünger, C. Buchheim, and S. Leipert. A library of algorithms for graph drawing. 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 456–457. Springer, 1998. Project home page at `http://www.mpi-sb.mpg.de/AGD/`.
12. H. Nagamochi and T. Kameda. Constructing cactus representation for all minimum cuts in an undirected network. *Journal of the Operations Research*, 39(2):135–158, 1996.
13. R. Tamassia. On embedding a graph in the grid with the minimum number of bends. *SIAM Journal on Computing*, 16:421–444, 1987.
14. R. Tamassia, G. Di Battista, and C. Batini. Automatic graph drawing and readability of diagrams. *IEEE Transactions on Systems, Man and Cybernetics*, 18(1):61–79, 1988.