Shrinking the Search Space for Clustered Planarity

Markus Chimani[1,⋆] and Karsten Klein[2,⋆⋆]

[1] Faculty of Math. and Comp. Sci., Friedrich-Schiller-University Jena, Germany
markus.chimani@uni-jena.de
[2] School of Information Technologies, The University of Sydney, Australia
kklein@it.usyd.edu.au

Abstract. A clustered graph is a graph augmented with a hierarchical inclusion structure over its vertices, and arises very naturally in multiple application areas. While it is long known that planarity—i.e., drawability without edge crossings—of graphs can be tested in polynomial (linear) time, the complexity for the clustered case is still unknown.

In this paper, we present a new graph theoretic reduction which allows us to considerably shrink the combinatorial search space, which is of benefit for all enumeration-type algorithms. Based thereon, we give new classes of polynomially testable graphs and a practically efficient exact planarity test for general clustered graphs based on an integer linear program.

1 Introduction

Clusters can be used to group similar objects or to reflect organizational structures, and allow to model the corresponding information in many application areas, as for example code packages in software engineering, departments in business processes, or cell compartments in biological processes.

In the following, we will always consider a clustered graph $C = (G = (V, E), T)$. Thereby, G is an undirected graph and the rooted tree T specifies the cluster hierarchy: the leaves of T are exactly the vertices of V. Any inner node of T is a *cluster*, the root of T is the *root-cluster*. Let σ be an arbitrary cluster. We denote the parent cluster of σ in T by $\mathrm{parent}(\sigma)$, and can label clusters by *levels*, i.e., their graph theoretic distance from the root cluster in T, denoted by $\mathrm{level}(\sigma)$. A *subcluster* of σ is a cluster in the subtree rooted at σ. The set $V(\sigma) \subseteq V$ specifies all leaves of the subtree of T rooted at σ. We write $G[\sigma] := G[V(\sigma)]$ to denote the subgraph of G induced by the vertices $V(\sigma)$ and $G - \sigma := G[V \setminus V(\sigma)]$ for the graph remaining after removing all vertices of σ.

A clustered embedding of a clustered graph $C = (G, T)$ is a crossing-free embedding of the underlying graph G into the plane. Furthermore, each cluster σ has to be drawable as a topological disc such that $G[\sigma]$ is completely inside and $G - \sigma$ completely

⋆ M. Chimani was funded by a Carl-Zeiss-Foundation juniorprofessorship.
⋆⋆ K. Klein was partly supported by ARC grant H2814 A4421, Tom Sawyer Software, and NewtonGreen Software.

W. Didimo and M. Patrignani (Eds.): GD 2012, LNCS 7704, pp. 90–101, 2013.

outside of this disc, and there is no point where the borders of any two cluster discs intersect. One may say the clusters' discs form a laminar family compatible with T. All edges connecting $G[\sigma]$ and $G - \sigma$ cross the border of σ's disc exactly once. A clustered graph that allows a clustered embedding is said to be clustered planar [6].

For brevity, we use the terms *c-graph*, *c-planarity*, and *c-embedding* to denote a clustered graph, clustered planarity, and a clustered embedding, respectively.

The quest for the complexity of deciding c-planarity is still ongoing. Currently only some special graph classes are known to be solvable in polynomial time, e.g., [4, 7, 8, 10–12]. Most importantly, there is a linear-time algorithm to check cluster-connected graphs (i.e., $G[\sigma]$ is connected for each σ) [5]. The only NP-hardness result [1] considers splitting clusters such that a c-graph becomes c-planar, but does not induce hardness of asking if 0 splits suffice.

Definition 1 (Completely connected). *A c-graph $C = (G, T)$ is* completely connected *if, for each cluster $\sigma \in T$, the graphs $G[\sigma]$ and $G - \sigma$ are connected.*

Within that context, we may speak of *cluster connectivity* (*c-connectivity* for short) and of *co-connectivity* when considering the connectivity of $G[\sigma]$ and $G - \sigma$, respectively. The following result is due to Cornelsen and Wagner [3].

Theorem 1 (Cornelsen and Wagner [3]). *A completely connected c-graph $C = (G = (V, E), T)$ is c-planar iff G is planar.*

Corollary 2. *A c-graph $C = (G, T)$ is c-planar iff it can be augmented with additional edges to obtain a c-graph $C' = (G', T)$ such that C' is completely connected and G' is planar. In this case we say C is* completely augmentable.

This corollary gives rise to a straight-forward scheme to test c-planarity: we planarly augment G such that C becomes completely connected, or show that this is infeasible. In [2], an integer linear program (ILP) based on this idea was proposed and solved using branch-and-cut techniques. In fact, the paper considered the problem of finding a maximum c-planar clustered subgraph, allowing pure c-planarity testing as a special case with pruning-based performance optimization. However, even for this restriction, the approach performed acceptably only for small graphs. This seemed to be mainly due to the fact that an enormous number of edges has to be considered in order to augment the c-graph.

Contribution. We investigate c-planarity from a graph-theoretic point of view (Section 3), allowing us to considerably reduce the number of edges that have to be considered for the augmentation. The consequences (Section 4) are two-fold: We can specify a new family of c-graphs for which c-planarity can be tested in polynomial time. It fact, we show that c-planarity testing is fixed parameter tractable (FPT) in the number of "complicated" edges. Furthermore, our result allows a practically stronger ILP than the one described in [2].

2 Definitions

A central ingredient of our approach are the following concise definitions, which allow us to deduce and exploit certain properties. Let $C = (G = (V, E), T)$ be a c-graph for

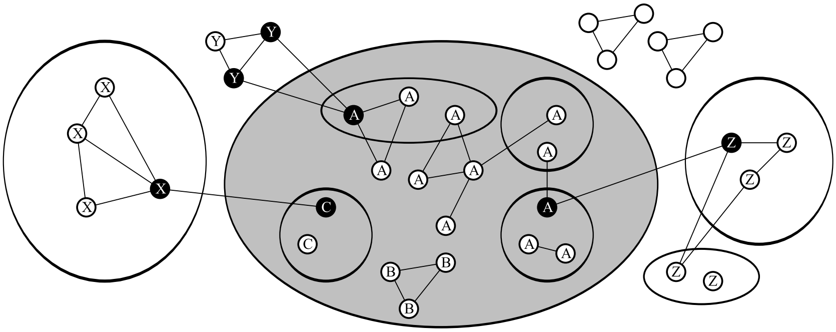

Fig. 1. Example of a (clustered planar) clustered graph: the large ellipses denote the clusters' discs. Let σ be the gray shaded cluster. It consists of 7 chunks, has 3 bags (A,B,C), and induces 3 satchels (X,Y,Z). Non-labeled vertices (outside of σ) belong to no satchel. Black vertices inside (outside) of σ denote outer-active (inner-active, resp.) vertices w.r.t. σ.

all definitions in this section. A well-established concept when considering c-planarity are *chunks*: A *chunk* of σ is a connected component in $G[\sigma]$. Some of our definitions will build upon this concept to observe some more helpful structures. Figure 1 visualizes the concepts.

Definition 2 (Outer-active, Inner-active). *Let σ be a cluster of C. A vertex $v \in V(\sigma)$ is* outer-active w.r.t. σ *iff there exists a vertex $u \in V \setminus V(\sigma)$ with $(v,u) \in E$. Let $V^{\text{out}}(\sigma) \subseteq V(\sigma)$ denote these vertices.*

Let σ be a non-root cluster of C. Any vertex $v \in V \setminus V(\sigma)$ is inner-active w.r.t. σ *iff there exists a vertex $u \in V(\sigma)$ with $(v,u) \in E$. Let $V^{\text{in}}(\sigma) \subseteq V \setminus V(\sigma)$ denote these vertices.*

Definition 3 (Bag). *Given a cluster σ of C, a* bag *is a maximum subgraph β of $G[\sigma]$ consisting of chunks in σ that are "connected via subclusters". Formally, for each pair of chunks ch_s, ch_t in β there exists a sequence $ch_s = ch_1, ch_2, \ldots, ch_k = ch_t$ of chunks in β such that for each pair ch_i, ch_{i+1}, $1 \leq i < k$, there exists a subcluster σ' of σ that contains vertices of both ch_i and ch_{i+1}.*

Definition 4 (Satchel). *Given a non-root cluster σ of C, a* satchel w.r.t. σ *is an inclusion-wise minimal bag in the c-graph induced by $G - \sigma$ that contains no outer-active vertex w.r.t. its smallest containing cluster, but at least one inner-active vertex w.r.t. σ.*

Let β be a bag or satchel, then $V(\beta)$ denotes the set of vertices in β.

Observation 3. *No two satchels w.r.t. the same cluster σ contain a common vertex. The union of all satchels w.r.t. σ does not necessarily cover $G - \sigma$.*

Proof. Assume two satchels β', β'' w.r.t. σ share a common vertex. Since satchels are bags, they must be bags of different clusters σ' and σ'', where one cluster (say σ'') is a subcluster of the other. Furthermore, $V(\beta'') \subseteq V(\beta')$, but since β' is a bag, β'' contains at least one vertex v adjacent to a vertex of $V(\beta') \setminus V(\beta'')$. But then v is outeractive w.r.t. σ'', and hence β'' cannot be a satchel.

An example for the non-covering of the satchel union can be seen in Fig. 1. $\square$

3 Potential Augmentation Edges

In order to achieve complete connectivity, we have to insert additional edges. We will show that we do not have to consider all potential edges $(V \times V) \setminus E$.

We first observe that we may decompose our given c-graph into several independent instances: Bags and satchels constitute clustered components in a cluster induced graph and its complement, respectively. They are composed of chunks of the respective graph that are connected via common clusters. If such a clustered component inside a parent cluster is not connected to the complement, i.e., does not contain outeractive vertices, we can process it independently of the remainder of the input graph. In the following we can hence assume that our graph is not further decomposable using this strategy.

Let $\mathcal{P}$ be a partition of a graph H into disjoint subgraphs (such a subgraph need not to be connected itself), and let $\mathcal{S} \subseteq \mathcal{P}$ be a subset of these subgraphs. We say two subgraphs $S, S' \in \mathcal{S}$ are *directly* connected if there is an edge in H connecting a vertex of S and a vertex of S'. They are *semi-directly* connected if there is an ordered subset $\{S = S_1, S_2 \ldots, S_k = S'\} \subseteq \mathcal{S}$ such that, for each $1 \leq i < k$, S_i is directly connected with S_{i+1} in H. We say S, S' are *indirectly* connected if there exists an arbitrary path between a vertex of S and one of S' in H. We say $\mathcal{S}$ is *semi-directly (indirectly) interconnected* if all pairs of subgraphs of $\mathcal{S}$ are semi-directly (indirectly, respectively) connected.

Recall that a chunk is a connected subgraph in G. Naïvly it is clear that in order to establish c-connectivity, it suffices to ensure that all chunks of a cluster σ are semi-directly interconnected. Similarly, to establish co-connectivity, it suffices to require semi-direct interconnectivity between the connected components in $G - \sigma$. Our definition of bags and satchels allows us to simplify this view.

Lemma 4. *A c-graph $C = (G, T)$ is completely augmentable iff there exists a planarly augmented graph $G' \supseteq G$ in which, for each cluster $\sigma \in T$, σ's bags are semi-directly interconnected (c-connectivity) and σ's satchels are indirectly interconnected in $G' - V_\sigma$ (co-connectivity).*

Proof. One direction of this lemma is straight-forward: If C is c-planar, let $G^* \supseteq G$ be a planarly augmented graph such that $C^* = (G^*, T)$ is completely connected. By definition, $G^*[V \setminus V(\sigma)]$ is connected for each cluster σ which hence induces indirected interconnectivity between its satchels. Similiary, $G^*[V(\sigma)]$ is connected and hence all bags are at least indirectly interconnected. Since the bags of σ cover all of $V(\sigma)$, the edges connecting the bags are the ones establishing semi-direct connections.

Hence we can concentrate on proving that an augmentation as described in the lemma (if it exists) suffices to show c-planarity of C.

We prove the c-connectivity part by induction bottom-up in the cluster tree: In clusters without subclusters, bags and chunks are the same concept, and hence connecting bags ensures c-connectivity of such a cluster. Consider some other cluster σ. By induction, we already ensured c-connectivity for all its subclusters, in particular also for the subclusters that establish the membership of chunks to some common bag in the definition. Hence, the chunks of a common bag are ensured to be connected, and at σ it suffices to connect bags with each other.

Consider co-connectivity. By the first part of the proof we know that each bag in each cluster is connected in G'. Since satchels are the remaining parts of bags for some cluster after the deletion of a subcluster, each satchel is connected in G' as well. Consider any non-root cluster σ, its satchels Σ, and let $G'_\sigma := G' - V_\sigma$. The satchels Σ are connected subgraphs in G'_σ, and a complete augmentation of C requires G'_σ to be connected. Our augmentation ensures that all satchels are indirectly interconnected within G'_σ. It hence remains to show that, given our augmentation G', we can always connect the connected subgraphs of G'_σ not covered by satchels without introducing crossings. Formally, let $\mathcal{G}^*_\sigma$ denote the connected subgraphs of G'_σ that do not contain at least one vertex of a satchel of Σ. We will identify new edges E' such that $G'' := G' + E'$ is planar and induces a completely connected supergraph of C.

By induction top-down in the cluster tree, we may assume that the c-graph $C'' := (G'', T)$ is co-connected for each ancestor cluster of σ. (The base case for the root-cluster is trivial, since co-connectivity simply means that the empty graph has at most one connected component.) For any two subgraphs $G^*_1, G^*_2 \in \mathcal{G}^*_\sigma$ at least one of them has to be completely contained in $\mu := \mathrm{parent}(\sigma)$ by our induction hypthesis. Therefore, all but one subgraph (say G^*_1) in $\mathcal{G}^*_\sigma$ has to be completely contained in μ. But then, any subgraph in $\mathcal{G}^*_\sigma \setminus G^*_1$ is also a connected component in G' since it has no inner-active vertices by definition, and no two such subgraphs share any common cluster beneath μ as G' assures c-connectivity. Hence, such a subgraph can be solved independently from the remainder of the graph: If it is not c-planar, then C is not c-planar as well. If it is c-planar, a c-embedding of it can be placed anywhere within the disc of μ. We can easily connect these subgraphs with additional edges (adding them to E') without introducing crossings, ensuring co-connectivity.

Finally, consider the unique component G^*_1 that contains vertices outside of μ (if it exists). If it also contains vertices from μ then it contains an inner-active vertex w.r.t. μ, was (part of a) satchel w.r.t. μ, and therefore is properly connected in G' by definition of the lemma. Hence it lies completely outside of μ, and was therefore already a non-satchel-covered subgraph w.r.t. μ. By our top-down induction hypothesis, μ is already connected via edges of E'. □

Using the above lemma as a starting point and considering the special roles of outer- and inner-active vertices, we can state the main theoretical result of this paper: a reduced set of edges potentially necessary for our graph augmentation.

Theorem 5. *A clustered graph $C = (G = (V, E), T)$ with planar G is clustered planar iff there is a planar augmented graph $G_1 = (V, E \cup A)$, $E \cap A = \emptyset$, of G satisfying the following properties for each cluster σ (assuming σ to be a non-root cluster only if necessary):*

(P1) Let $\mathcal{B}$ be the bags of σ in C that contain at least one vertex of $V^{\mathrm{out}}(\sigma)$. The bags of $\mathcal{B}$ are semi-directly interconnected among each other in G_1, using only edges incident (on both ends) to $V^{\mathrm{out}}(\sigma)$.

(P2) Let $\mathcal{B}$ be the satchels w.r.t. σ in C. The satchels of $\mathcal{B}$ are (possibly indirectly) connected among each other in $G_1 - \sigma$, using only additional edges connecting vertices that are inner-active or outer-active for at least one cluster σ' with $\mathrm{level}(\sigma') \leq \mathrm{level}(\sigma)$.

Let $C_1 = (G_1, T)$ be the c-graph resulting from augmenting C with the edges A. On the one hand, assume there is an augmentation as described in the theorem. This *partial* augmentation is not necessarily a *full* augmentation that would ensure complete connectivity! Therefore we will show:

Claim 1. *If there exists an augmentation as described in the theorem, then we can always extend C_1 via some further edges B and B' to achieve complete connectivity with planar $G_1 + B + B'$.*

On the other hand, we need to show:

Claim 2. *If there does not exist an augmentation as described in the theorem, then C is not c-planar.*

We will show this via contradiction: Whenever C is c-planar, we can construct a planar augmentation satisfying properties (P1) and (P2). Overall, by proving these two claims in the subsequent subsections, we establish our main theorem.

3.1 Proof of Claim 1

In the following, we will augment $C_1 = (G_1, T)$ stepwise to obtain a completely connected c-graph with planar underlying graph.

Lemma 6. *Let $\mathcal{C}$ be the set of clustered subgraphs of C_1 induced by the connected components of G_1. Each c-graph in $\mathcal{C}$ is completely cluster connected.*

Proof. Assume there would be a clustered subgraph $S = (G'_1, T') \in \mathcal{C}$ that is not completely connected. Hence, there exists some cluster σ such that, despite G'_1 being connected, either $G'_1[\sigma]$ or $G'_1 - \sigma$ is not connected. Assume $G'_1[\sigma]$ is not connected, and consider the chunks in this graph. Since G'_1 is connected, each chunk has to contain an outer-active vertex. Property (P1) ensures that they are connected via edges of A, a contradiction.

Assume $G'_1 - \sigma$ is not connected, and consider its connected components. Since G'_1 is connected, each arising component is incident to σ in G'_1, and hence has an inner-active vertex w.r.t. σ. Property (P2) guarantees that all satchels w.r.t. σ (having inner-active vertices) are interconnected between each other. For a contradiction we need to show that each component contains vertices of a satchel, and no satchel contains vertices of more than one component. The former is witnessed by the inner-active vertices. For the latter we only need to observe that a satchel is connected in G_1 due to property (P1) and can therefore not span multiple connected components. □

Hence, each clustered subgraph S in $\mathcal{C}$ is c-planar, and we can (arbitrarily) fix any c-embedding for each S, which we retain in all the following steps.

B-augmentation. We will first planarly augment $C_1 = (G_1 = (V, E \cup A), T)$ with additional edges B to obtain $C_2 = (G_2 = (V, E \cup A \cup B), T)$. This augmentation will proceed in a bottom-up fashion over T, establishing c-connectivity for all clusters. Thereby, the already fixed subembeddings are retained and extended w.r.t. the edges B.

Initially, $B = \emptyset$. Consider any cluster σ. By induction, all subclusters (if any) of σ are already connected via edges of $E \cup A \cup B$. We will only introduce edges (u, v) into the set B where σ is the least common ancestor of u and v in T. Let $H := G + A + B$ be the current graph, containing the up to now introduced edges B. Let $D = (H, T)$ be the corresponding clustered graph, and consider the chunks in $H[\sigma]$. Due to the induction, each chunk is itself also a bag and vice versa; furthermore, there is a one-to-one correspondence between the chunks in $H[\sigma]$ and the bags in $G[\sigma]$. Also by induction, we already have a c-planar embedding for each of these chunks.

The edges of A connect the chunks with outer-active vertices to a single embedded component K in $G_1[\sigma]$ due to property (P1). Assume that K exists. Hence it remains to connect the bags ignored by property (P1), i.e., the chunks in $H[\sigma]$ without any outer-active vertices, to K. Each such chunk ch does not have any connection to $G - \sigma$, and hence can be placed anywhere within the cluster σ (but not within subclusters of σ). In particular, we can choose an arbitrary outer-active vertex v on K, an arbitrary face f incident to v on the outside of K, and an arbitrary vertex u on the outer face of embedded ch. We add the edge (v, u) as an element of B and embed ch into f. No additional crossings arise.

If K does not exist, we can connect the chunks of $H[\sigma]$ in a very simple way: for each chunk, pick an arbitrary vertex on its outer face as its "docking vertex". Pick one of these vertices as the center and connect all other docking vertices to this center vertex (adding the new edges to B). It is trivial to choose a c-planar embedding for this connected structure. $\diamond$

B'-augmentation. We now further augment C_2 planarly, introducing additional edges B' such that the arising clustered graph $C_3 = (G_3 = (V, E \cup A \cup B \cup B'), T)$ is co-connected, and hence completely connected.

We proceed in a top-down fashion. Initially, $B' = \emptyset$. Let σ be any non-root cluster, and $H := G_2 + B'$ the current graph, containing the up to now introduced edges B'. If $H - \sigma$ is connected, there is nothing to do. Otherwise, let $\mathcal{K}$ be the corresponding connected components. Note that all satchels in C lie in a common such component and since C_2 is c-connected, all components in $\mathcal{K}$ are adjacent to vertices in σ in H. Let $K \in \mathcal{K}$ be any such component, and consider the embedded graph $H_K := H - K$. The component K lies in a face f of H_K, and since $K \cap H[\sigma] = \emptyset$ and C_2 was cluster embedded, we know that f cannot be in the inside of embedded $H[\sigma]$. Hence, there is at least one vertex v of some other component $K' \in \mathcal{K}$ incident to f. Clearly, there is a subface f' of f in embedded H that is incident to v and some vertex u of K. In H, we can add an edge (v, u) into face f' without introducing any crossings, simultaneously adding this edge to B'. We iterate the process until $H - \sigma$ is connected. $\diamond$

To summarize this first part (Claim 1) of the proof of Theorem 5: We established that a planar augmentation satisfying the properties (P1) and (P2) can always be extended to a planar augmentation yielding a completely connected supergraph of C. Hence, due to Theorem 1 and Corollary 2, an augmentation according to the theorem establishes that C is c-planar.

3.2 Proof of Claim 2

Assume that C is c-planar. We show that there always exists a planar augmentation A satisfying properties (P1) and (P2). Fix an arbitrary clustered embedding of C. We will first augment C with edges A_0 to establish c-connectivity, and afterwards with additional edges A_1 to establish co-connectivity. Then, $A_0 \cup A_1$ forms the feasible set A described by the theorem.

A_0-augmentation. We proceed in a bottom-up fashion over the cluster tree to establish c-connectivity via additional edges A_0 with planar $G + A_0$. We retain the previously fixed embedding of G. Initially, $A_0 = \emptyset$. Let σ be any cluster and $H := G + A_0$ the (embedded) current graph containing the already added edges A_0. By induction, all subclusters of σ are c-connected in the c-graph $D := (H, T)$. Hence every bag in σ w.r.t. C is already a chunk in σ w.r.t. D.

If σ contains only a single chunk in H, there is nothing to do. Assume there are multiple chunks, then consider the cyclic order ord of all the edges leaving σ, according to the fixed embedding. Let v and u be a pair of outer-active vertices on distinct chunks where an edge e_v incident to v directly preceeds an edge e_u incident to u in ord. We add the edge $e = (u, v)$ to A_0 (and therefore H), as it is feasible w.r.t. property (P1). It remains to argue why e does not introduce any crossings with other edges or other clusters. We route e from u close to e_u until the imaginary cluster border, and then along this border until we reach the first crossing edge, which by construction is e_v; we continue the routing along this e_v in order to reach v. As e_u and e_v are crossing free, so are the corresponding pieces of e. Since any part of the border is only traversed at most once by our constructions, there cannot be further crossings. Finally, our routing cannot illegaly cross any other cluster border, since cluster borders do not intersect.

Iterating this process leaves a graph H with a single chunk in σ, i.e., a c-connected cluster σ where the bags in C with outer-active vertices are semi-directly interconnected using only edges with outer-active end vertices. Therefore, performing these steps bottom-up over T, gives a planar, embedded c-connected c-graph $C_0 = (G_0 = (V, E \cup A_0), T)$ satisfying property (P1). $\diamond$

A_1-augmentation. We proceed level-wise in a top-down fashion over the cluster tree to establish co-connectivity via additional edges A_1 with planar $G_0 + A_1$. The previously fixed embedding of G_0 is retained. Initially, $A_1 = \emptyset$. Let σ be any non-root cluster and $H := G + A_1$ the (embedded) current graph containing the already added edges A_1. By the A_0-augmentation, $D := (H, T)$ is c-connected; in particular G_0 is connected. Hence every satchel in C w.r.t. σ is completely contained in some connected component in $G_0 - \sigma$. It may happen that two such satchels are within the same such component; even then, we have to ensure that they are connected using edges allowed by property (P2).

By induction, all superclusters of σ (if any) are co-connected in D using edges valid for property (P2). Let Σ be the satchels w.r.t. σ in C, and $\Sigma^{\text{out}} \subseteq \Sigma$ those that contain vertices outside of $\mu := \text{parent}(\sigma)$. Then Σ^{out} gives rise to satchels Σ^{μ} w.r.t. μ in C (with $|\Sigma^{\mu}| \geq |\Sigma^{\text{out}}|$). The satchels Σ^{μ} have already been feasibly connected by induction. It remains to feasibly connect the satchels $\Sigma^{\text{in}} := \Sigma \setminus \Sigma^{\text{out}}$ among each other and (if it exists) with the connected component K containing all Σ^{μ} in H.

If $|\Sigma| \leq 1$ or $|\Sigma^{\text{in}}| = 0$, there is nothing to do. Otherwise, let $\Sigma' := \Sigma^{\text{in}} \cup \{K\}$ (ignoring K if it does not exist). Consider the cyclic order ord of all the edges leaving σ in D, according to the fixed embedding. Let v and u be a pair of inner-active vertices on subgraphs in Σ' where an edge e_v incident to v directly preceeds an edge e_u incident to u in ord.

We want to (planarly) connect u to v. Naïvly this could be done by adding an edge $e = (u, v)$ routed as follows: we start at u and follow e_u until we reach the imaginary cluster border of σ. We follow this border on the outside until we reach e_v, which we follow towards v. If this routing does not introduce any edge or infeasible cluster crossings, we add e as described. Otherwise there are only two potential obstacles:

(a) We may cross through a cluster ψ (i.e. enter and leave ψ), an ancestor of σ, but neither u nor v belong to ψ. But then, u and v have also been inner-active w.r.t. ψ, and are hence, by induction, already connected using feasible edges.

(b) Assume we do not cross through a cluster but through an edge $\bar{e} \in A_0 \cup A_1$, which was therefore not considered when defining inner-active and outer-active vertices. Let w be the end vertex of $\bar{e}$ not in σ. However, when $\bar{e} \in A_0$ ($\bar{e} \in A_1$), then, by induction and our level-wise process, w was an outer-active (inner-or-outer-active, respectively) vertex when considering some cluster at a level of at most $\text{level}(\sigma)$. We can add an edge $e' = (u, w)$ routed along e_u until the border of the inclusion-wise maximal cluster containing σ but neither u and w, then along the outside of this border until it would cross the first edge (which, by construction, is $\bar{e}$), and then along this edge towards w. No crossings arise. We add this edge to A_1, consider w as the new u and iterate the full process. Observe that (u, w) satisfies property (P2). ◇

To summarize this second part (Claim 2) of the proof of Theorem 5: We constructively established that any c-planar graph allows a planar augmentation as required by our theorem.

4 Consequences of Theorem 5

Fixed Parameter Tractability. Theorem 5 directly induces a class of c-graphs for which testing c-planarity is *fixed parameter tractable* (FPT) w.r.t. some parameter (see, e.g., [13] for an introduction to FPT). The key idea is that, if some specific parameter p is considered bounded, then there is an algorithm running in polynomial time. Moreover, this runtime needs to be bounded by $\mathcal{O}(f(p) \cdot \text{poly}(n))$ for some computable (usually at least exponential) function f depending only on p, and some fixed polynomial $\text{poly}(\cdot)$ independent of p.

Let p be the number of edges crossing cluster borders, and bounded. Then, the number of outer- and inner-active vertices is bounded as well, and Theorem 5 ensures that we only have to consider augmentation edges connecting such vertices. Therefore, there is only a bounded number of augmentation edges, and consequently of possible augmentations. For any possible augmentation, we can test planarity (in linear time [9]) of the resulting graph. We have:

Theorem 7. *Consider any clustered graph C (with given bags and satchels) with bounded number of edges that connect different clusters. We can test clustered planarity of C in linear time.*

Table 1. (top) Properties of the benchmark instances. *c-con.* (*compl. con.*) give the number of c-connected (completely connected) instances. **(middle)** Influence of shrinking. Column ✓ gives the number of c-planar instances verified within 20 minutes. The column $\not S$ ($\not F$) gives the number of instances not solved by `SHRINK` (`FULL`) within this time bound. **(bottom)** Runtime comparison between `SHRINK` and `FULL`. *% Improv.* gives the percentage of instances where `SHRINK` lead to measurable speed-ups, the following column *Improv.* (*Increase*) gives the average difference in runtime when `SHRINK` (`FULL`) was the faster variant. The last two column give the overall average runtime of both approaches over the instances solved by both approaches.

	# inst.	*c-con.*	*compl. con.*	*# clusters min*	*avg*	*max*	*# vertices*	*# edges max*
PlanarSmall	1815	25	2	3	4	9	max 29	30
P-Small	84	24	2	6	10.16	16	25	37
P-Medium	168	48	0	11	25.25	46	50 and 75	112
P-Large	84	24	0	26	41.5	62	100	150

	# c-planar ✓	$\not S$	$\not F$	*Avg ratio #vars* `SHRINK` / `FULL`	*Avg # variables* for `SHRINK`	*#Added edges*
PlanarSmall	**1541**	16	19	0.63	85	6.6
P-Small	**62**	2	2	0.47	131	6.6
P-Medium	**67**	11	22	0.37	821	6.9
P-Large	**26**	11	20	0.32	1970	7.6

	% Improv. `SHRINK`	*average runtime change* *Improv.*	*Increase*	***average runtime*** **`SHRINK`**	**`FULL`**
PlanarSmall	35%	9.3	3.0	**3.5**	**4.8**
P-Small	41%	27.7	0.7	**1.5**	**12.6**
P-Medium	74%	67.1	1.4	**4.5**	**53.7**
P-Large	73%	28.1	8.1	**16.8**	**35.1**

ILP and Experiments. We briefly recapitulate the ILP model of [2], reduced to the special case of testing c-planarity. We have a binary variable x_f for each element $f \in F$, where F denotes the set of potential augmentation edges. Such an edge is used iff the corresponding variable is set to 1. In the original formulation, one has to consider $F := V \times V \setminus E$. Based on our theorem we can drastically shrink this set, and therefore reduce the number of variables. Our ILP will allow a feasible solution iff the c-graph can be planarly augmented to complete c-connectivity. Hence we would formally not require any non-constant objective function, but in practice it turns out to be beneficial to optimize $\min \sum_{f \in F} x_f$. We only require two types of constraints: *Cut* constraints assure c-connectivity and co-connectivity of the augmented graph, and *Kuratowski* constraints assure its planarity by forbidding Kuratowski subdivisions. We solve the ILP using branch-and-cut, with a separation step that checks c-/co-connectivity and planarity and adds corresponding constraints as needed, cf. [2] for details.

We (re)implemented our branch-and-cut approach as a module in OGDF (`www.ogdf.net`) using the ABACUS framework, resulting in a more efficient code than the one used in [2], fixing also the handling of some special cases. We experimentally compare the ILPs resulting from using the full and the shrinked search space, denoted by `FULL` and `SHRINK`, respectively, using our new consistent implementation.

The experiments were run on a 2.4GHz AMD Opteron with a time limit of 20 minutes. Our benchmark set consists of two classes of c-graphs: *PlanarSmall* denotes the c-graphs with underlying planar graph from [2]. The three sets *P-Small*, *P-Medium*, and *P-Large* are newly generated c-graphs with up to 100 vertices. They arose from random planar connected graphs, creating a random c-connected cluster structure, and some random reassignments of vertices to other clusters, cf. Table 1(top).

Table 1(middle) shows the success rate of our ILP: Even for the largest graphs we could decide c-planarity in all but 11 out of 84 instances. The table also shows that `SHRINK` reduces the search space to roughly a third of the original size for medium and large graphs. Yet, we can see that there is potential for further improvements, as the number of variables is still much larger than the number of finally required augmentation edges. Naïvely, the `SHRINK` ILP should always perform at least as well as `FULL`. Yet, for small graphs the overhead of first computing the shrinked search space may dominate the benefits of the smaller ILP, cf. Table 1(bottom). Overall we observe that using `SHRINK` decreases the average running time (over instances solved by both approaches) by a factor 2–10. Even for the smallest and simplest graphs, it is in general beneficial.

5 Conclusions

It was well-known that c-planarity of a c-graph can be shown by planarly augmenting it such that it becomes completely c-connected. We showed that we can drastically shrink the potential search space for such an augmentation, and that, in fact, we do not need a full augmentation in order to decide c-planarity.

The direct implications of this result are not only a new class of polynomially testable graphs but an FPT algorithm for any c-graph. Furthermore, it allows to speed-up the previously rather impractical ILP approach such that it may now be considered a viable option for medium sized graphs. From the practical point of view, certain special cases could be considered to reduce the search space even further. E.g., bags with ≤ 3 outgoing edges can be solved independently, as any c-planar embedding of such a bag (or its mirror) can always be reintroduced into the rest of the graph. As such options would result in bulky code and rather unelegant case distinctions, we refrain from doing so in detail.

We assume that further investigations to reduce the necessary search space could lead to even stronger c-planarity FPT results. Furthermore larger graph classes with polynomial c-planarity test seem to be achievable by non-trivial combinations of our results with other known efficiently testable graph classes.

Conjecture 8. *Let $C = (G, T)$ be a c-graph and F the edges connecting different clusters σ, σ' such that at least one cluster σ'' on the path $\sigma - \sigma'$ in T has non-connected $G[\sigma'']$. If $|F|$ is bounded, we can test c-planarity of C efficiently.*

We feel that such considerations are also a step towards the ultimate question of the complexity of general c-planarity testing, or could at least steer us to useful small examples where the problem becomes intuitively hard.

References

1. Angelini, P., Frati, F., Patrignani, M.: Splitting Clusters to Get C-Planarity. In: Eppstein, D., Gansner, E.R. (eds.) GD 2009. LNCS, vol. 5849, pp. 57–68. Springer, Heidelberg (2010)
2. Chimani, M., Gutwenger, C., Jansen, M., Klein, K., Mutzel, P.: Computing Maximum C-Planar Subgraphs. In: Tollis, I.G., Patrignani, M. (eds.) GD 2008. LNCS, vol. 5417, pp. 114–120. Springer, Heidelberg (2009)
3. Cornelsen, S., Wagner, D.: Completely connected clustered graphs. J. Discrete Algorithms 4(2), 313–323 (2006)
4. Cortese, P.F., Di Battista, G., Patrignani, M., Pizzonia, M.: Clustering cycles into cycles of clusters. J. Graph Alg. Appl. 9(3), 391–413 (2005)
5. Dahlhaus, E.: A Linear Time Algorithm to Recognize Clustered Planar Graphs and Its Parallelization. In: Lucchesi, C.L., Moura, A.V. (eds.) LATIN 1998. LNCS, vol. 1380, pp. 239–248. Springer, Heidelberg (1998)
6. Feng, Q.W., Cohen, R.F., Eades, P.: Planarity for Clustered Graphs. In: Spirakis, P.G. (ed.) ESA 1995. LNCS, vol. 979, pp. 213–226. Springer, Heidelberg (1995)
7. Goodrich, M.T., Lueker, G.S., Sun, J.Z.: C-Planarity of Extrovert Clustered Graphs. In: Healy, P., Nikolov, N.S. (eds.) GD 2005. LNCS, vol. 3843, pp. 211–222. Springer, Heidelberg (2006)
8. Gutwenger, C., Jünger, M., Leipert, S., Mutzel, P., Percan, M., Weiskircher, R.: Advances in C-Planarity Testing of Clustered Graphs. In: Goodrich, M.T., Kobourov, S.G. (eds.) GD 2002. LNCS, vol. 2528, pp. 220–235. Springer, Heidelberg (2002)
9. Hopcroft, J., Tarjan, R.: Efficient planarity testing. J. ACM 21(4), 549–568 (1974)
10. Jelínek, V., Jelínková, E., Kratochvíl, J., Lidický, B.: Clustered Planarity: Embedded Clustered Graphs with Two-Component Clusters. In: Tollis, I.G., Patrignani, M. (eds.) GD 2008. LNCS, vol. 5417, pp. 121–132. Springer, Heidelberg (2009)
11. Jelínek, V., Suchý, O., Tesař, M., Vyskočil, T.: Clustered Planarity: Clusters with Few Outgoing Edges. In: Tollis, I.G., Patrignani, M. (eds.) GD 2008. LNCS, vol. 5417, pp. 102–113. Springer, Heidelberg (2009)
12. Jelínková, E., Kára, J., Kratochvíl, J., Pergel, M., Suchý, O., Vyskocil, T.: Small clusters in cycles and eulerian graphs. J. Graph Alg. Appl. 13(3), 379–422 (2009)
13. Niedermeier, R.: Invitation to Fixed-Parameter Algorithms. Oxford Lecture Series in Mathematics and Its Applications. Oxford University Press (2006)