Constrained Level Planarity Is FPT with Respect to the Vertex Cover Number

Boris Klemz
Universität Würzburg, Germany

Marie Diana Sieper
Universität Würzburg, Germany

Abstract

The problem Level Planarity asks for a crossing-free drawing of a graph in the plane such that vertices are placed at prescribed y-coordinates (called *levels*) and such that every edge is realized as a y-monotone curve. In the variant Constrained Level Planarity, each level y is equipped with a partial order $\prec_y$ on its vertices and in the desired drawing the left-to-right order of vertices on level y has to be a linear extension of $\prec_y$. Constrained Level Planarity is known to be a remarkably difficult problem: previous results by Klemz and Rote [ACM Trans. Alg.'19] and by Brückner and Rutter [SODA'17] imply that it remains NP-hard even when restricted to graphs whose tree-depth and feedback vertex set number are bounded by a constant and even when the instances are additionally required to be either *proper*, meaning that each edge spans two consecutive levels, or *ordered*, meaning that all given partial orders are total orders. In particular, these results rule out the existence of FPT-time (even XP-time) algorithms with respect to these and related graph parameters (unless P=NP). However, the parameterized complexity of Constrained Level Planarity with respect to the vertex cover number of the input graph remained open.

In this paper, we show that Constrained Level Planarity can be solved in FPT-time when parameterized by the vertex cover number. In view of the previous intractability statements, our result is best-possible in several regards: a speed-up to polynomial time or a generalization to the aforementioned smaller graph parameters is not possible, even if restricting to proper or ordered instances.

2012 ACM Subject Classification Theory of computation → Fixed parameter tractability; Human-centered computing → Graph drawings; Mathematics of computing → Graphs and surfaces; Theory of computation → Computational geometry; Mathematics of computing → Paths and connectivity problems

Keywords and phrases Parameterized Complexity, Graph Drawing, Planar Poset Diagram, Level Planarity, Constrained Level Planarity, Vertex Cover, FPT, Computational Geometry

Digital Object Identifier 10.4230/LIPIcs.ICALP.2024.99

Category Track A: Algorithms, Complexity and Games

Related Version *Full Version*: `https://arxiv.org/abs/2404.16723` [28]

1 Introduction

A large body of literature related to graph drawing is dedicated to so-called upward planar drawings, which provide a natural way of visualizing a partial order on a set of items. An *upward planar drawing* of a directed graph is a crossing-free drawing in the plane where every edge $e = (u, v)$ is realized as a y-monotone curve that goes upwards from u to v, i.e., the y-coordinate strictly increases when traversing e from u towards v. The most classical computational problem in this context is Upward Planarity: given a directed graph, decide whether it admits an upward planar drawing. The standard version of this problem is NP-hard [18], but, if the y-coordinate of each vertex is prescribed, it can be solved in polynomial time [13, 21, 25], which suggests that a large part of the challenge of Upward

Planarity comes from choosing an appropriate y-coordinate for each vertex. However, when both the y-coordinate and the x-coordinate of each vertex are prescribed, the problem is yet again NP-hard [27], indicating that another part of the challenge comes from drawing the edges in a y-monotone non-crossing fashion while respecting the given or chosen coordinates of their endpoints. The paper at hand is concerned with the parameterized complexity of a generalization of the latter of these two variants of Upward Planarity, which is known as Constrained Level Planarity. It is expressed in terms of so-called level graphs, which are defined next; we adopt the notation and terminology used in [27].

Level Planarity. A *level graph* $\mathcal{G} = (G, \gamma)$ is a directed graph $G = (V, E)$ together with a *level assignment*, which is a function[1] $\gamma\colon V \to \mathbb{R}$ where $\gamma(u) < \gamma(v)$ for every edge $(u, v) \in E$. For every $i \in \mathbb{R}$ where $V_i = \{v \in V \mid \gamma(v) = i\}$ is non-empty, the set V_i is called a (the i-th) *level* of $\mathcal{G}$. The *width* of level V_i is $|V_i|$. The *level-width* of $\mathcal{G}$ is the maximum width of any level in $\mathcal{G}$ and the *height* of $\mathcal{G}$ is the number of (non-empty) levels. A *level planar drawing* of $\mathcal{G}$ is an upward planar drawing of G where the y-coordinate of each vertex v is $\gamma(v)$. We use L_i to denote the horizontal line with y-coordinate i. The level graph $\mathcal{G}$ is called *proper* if every edge spans two consecutive levels, that is, for every edge $(u, v) \in E$ there is no level V_j with $\gamma(u) < j < \gamma(v)$. The problem Level Planarity asks whether a given level graph admits a level planar drawing. In a series of papers [13, 21, 24, 25], it was shown that Level Planarity can be solved in linear[1] time; we refer to [16] for a more detailed discussion of the history of the corresponding algorithm and of alternative approaches to solve Level Planarity.

Constrained Level Planarity. In 2017, Brückner and Rutter [8] and Klemz and Rote [27] independently introduced and studied two closely related variants of Level Planarity, which are defined in the following. A *constrained level graph* $\mathcal{G} = (G, \gamma, (\prec_i)_i)$ is a triple corresponding to a level graph (G, γ) equipped with a family $(\prec_i)_i$ containing, for each level V_i, a partial order $\prec_i$ on the vertices V_i. A *constrained* level planar drawing of $\mathcal{G}$ is a level planar drawing of (G, γ) where, for each level V_i, the left-to-right order of the vertices V_i corresponds to a linear extension of $\prec_i$. For a pair of vertices $u, v \in V_i$ with $u \prec_i v$, we refer to $u \prec_i v$ as a *constraint* on u and v. The problem Constrained Level Planarity (CLP) asks whether a given constrained level graph admits a constrained level planar drawing. Ordered Level Planarity (OLP) corresponds to the special case of CLP where the given partial orders are total orders, which is polynomial time equivalent to prescribing the x-coordinate (in addition to the y-coordinate) of each vertex.

Klemz and Rote [27] established a complexity dichotomy for OLP with respect to both the maximum degree and the level-width. In particular, they showed that OLP is NP-hard even when restricted to the case where $\mathcal{G}$ has a level-width of 2 and the underlying undirected graph of G is a disjoint union of paths, i.e., a graph of maximum degree 2, path-width (and tree-width) 1, and feedback vertex/edge set number 0. In fact, with a simple modification[2]

[1] Traditionally, in the literature, the level assignment γ is defined as a surjective function that maps to an integer interval $\{1, 2, \dots, h\}$; it merely acts as a convenient way to encode a total preorder on V. It is well known that the traditional and our (more general) definition are polynomial-time equivalent: algorithms designed assuming the classical definition can also be applied in the more general context: one simple has to first sort the vertices by y-coordinates and then apply the traditional algorithm using the sorting-ranks as y-coordinates. We are using the given general definition as it eases the description of our algorithms; though, specific polynomial runtimes obtained in previous work that are stated in our introduction assume the classical definition.

[2] In the variable gadget of every variable u_j, one can remove the subdivision vertices of the tunnels with index larger than j. This modification does not influence the realizability of the instance since the left-to-right order of all tunnels is already fixed due to the subdivision vertices on level ℓ_0.

to their construction, the underlying undirected graph produced by the reduction becomes a disjoint union of paths with constant length, implying that even the tree-depth is bounded. (The definitions of all these classical graph parameters can be found, e.g., in [10].) It follows that CLP is NP-hard in the same scenario and when, additionally, each of the prescribed partial orders $\prec_i$ is a total order. OLP is (trivially) solvable in linear[1] time when restricted to proper instances [27]. In contrast, an instance of CLP can always be turned into an equivalent proper instance by subdividing each edge on each level it passes through without introducing any constraints on the resulting subdivision vertices [8]. Hence, CLP is NP-hard even in the proper case. Independently, Brückner and Rutter [8] also presented a proof for the NP-hardness of CLP, which relies on a very different strategy. It is not obvious whether the graphs produced by their construction have bounded tree-width, however, it is not difficult to see[3] that the socket/plug gadget used in their reduction can be utilized in the context of a reduction from 3-Partition to show that CLP remains NP-hard for proper instances whose underlying undirected graph is a single (rooted) tree of constant depth. In fact, the unpublished full version of [8] features such a construction [29].

On the positive side, Brückner and Rutter [8] presented a polynomial time algorithm for the special case of CLP where the input graph G has a single source. They further improved the runtime of this algorithm in [9]. Very recently, Blažej, Klemz, Klesen, Sieper, Wolff, and Zink studied the parameterized complexity of CLP and OLP with respect to the height of the input graph [7]. They showed that OLP parameterized by height is XNLP-complete (implying that it is in XP, but $W[t]$-hard for every $t \geq 1$). In contrast, CLP is NP-hard even if restricted to instances of height 4, but it can be solved in polynomial time if restricted to instances of height at most 3.

Other related work. Several other restricted variants of Level Planarity have been studied, e.g., Clustered Level Planarity [15, 2, 27], T-Level Planarity [30, 2, 27], and Partial Level Planarity [8]. In particular, in Partial Level Planarity, a given level planar drawing of a subgraph of the input graph $\mathcal{G}$ has to be extended to a full drawing of $\mathcal{G}$, which can be seen as a generalization of OLP and, in the proper case, a specialization of CLP. Level Planarity has been extended to surfaces different from the plane [4, 1, 5]. There are also related problems with a more geometric flavor, e.g., finding a level planar straight-line drawing where each face is bounded by a convex polygon [23, 26], and problems where the input is an undirected graph without a level assignment and the task is to find a crossing-free drawing with y-monotone edges that, if interpreted as a level planar drawing, satisfies or optimizes certain criteria, e.g., being proper or having minimum height [6, 14, 22].

Contribution. As discussed above, the previous results of Brückner and Rutter [8] and Klemz and Rote [27] rule out the existence of FPT-time (even XP-time) algorithms for CLP when considering the tree-width, path-width, tree-depth, or feedback vertex set number as a parameter, even when restricted to OLP or proper CLP instances (unless P=NP). As all of these parameters are bounded[4] by the vertex cover number, it is natural to study the parameterized complexity of CLP with respect to this parameter. We prove the following main result:

---

[3] In the strongly NP-hard 3-Partition problem [17], one has to partition $3n$ positive integers $B/4 < a_1, a_2, \ldots, a_{3n} < B/2$ of total sum nB into n triples (or *buckets*) of sum (or *size*) B. To reduce to CLP, one can simulate a bucket of size B as a sequence of B consecutive sockets and a number a_i as a_i plugs that are connected in a star-like fashion to a common ancestor v_i located above all these plugs. Finally, all ancestors v_i and all sockets are connected in a star-like fashion to a common root vertex.

[4] More precisely, $\mathrm{tw}(G) \leq \mathrm{pw}(G) \leq \mathrm{td}(G) - 1 \leq \mathrm{vc}(G)$ and $\mathrm{fvs}(G) \leq \mathrm{vc}(G)$.

▶ **Theorem 1.** *CLP parameterized by the vertex cover number is FPT.*

In view of the previous intractability statements, Theorem 1 is best-possible in several regards: a speed-up to polynomial time or a generalization to the aforementioned smaller graph parameters is not possible, even if restricting to OLP or proper CLP instances.

Organization. The proof of Theorem 1 and the remainder of this paper are organized as follows. We begin by introducing some basic notation, terminology, and other preliminaries in Section 2. In particular, we describe a partition of the vertex set of a given constrained level graph $\mathcal{G}$ into different categories with respect to a given vertex cover C and we show that the vertices of two of these categories are, in some sense, easy to handle. In Section 3, we introduce cores and (refined) visibility extensions of level planar drawings with respect to a fixed vertex cover C. Intuitively, the core-induced subdrawing of a (refined) visibility extension of a constrained level planar drawing Γ^* of $\mathcal{G}$ with respect to C is a drawing Λ_{core} that captures crucial structural properties of Γ^* and whose total complexity is bounded in $|C|$. The latter allows us to efficiently obtain such a core-induced subdrawing via the process of exhaustive enumeration. This is the first main step of the algorithm corresponding to the proof of Theorem 1, which is described in Section 4. Due to the properties of the core-induced subdrawing, it is then possible to place the remaining vertices in the subsequent main steps of the algorithm, each of which is concerned with the placement of the vertices of a particular vertex category. We conclude with a discussion of an open problem in Section 5. Proofs of statements marked with a $\star$ can be found in the appendix of the full preprint version [28].

2 Preliminaries

Conventions. Recall that in a level graph $\mathcal{G} = (G = (V, E), \gamma)$, the graph G is directed by definition. However, when it comes to vertex-adjacencies, we always refer to the underlying undirected graph of G, that is, the neighborhood of $v \in V$ is $\mathrm{N}_G(v) = \{u \in V \mid (u, v) \in E \vee (v, u) \in E\}$, the degree of v is $|\mathrm{N}_G(v)|$, and "a vertex cover of G" refers to a vertex cover[5] of the underlying undirected graph of G. The *level planar embedding* of a level planar drawing of $\mathcal{G}$ lists, for each level V_i, the left-to-right sequence of vertices and edges intersected by the line L_i in the drawing. Note that this corresponds to an equivalence class of drawings from which an actual drawing is easily derived, which is why algorithms for constructing level planar drawings (including our algorithms) usually just determine a level planar embedding. For brevity, we often use the term "drawing" as a synonym for "embedding of a drawing".

Vertex categories & notation. For $m \in \mathbb{N}$, we use $[m]$ to denote the set $\{1, 2, \ldots, m\}$. Let $\mathcal{G} = (G = (V, E), \gamma)$ be a (constrained) level graph and let C be a vertex cover of G. An *ear* of $\mathcal{G}$ with respect to C is a degree-2 vertex of $V \setminus C$ that is a source or sink. For a subset $X \subseteq C$, we define $V_X(C) = \{v \in V \setminus C \mid \mathrm{N}_G(v) = X\}$. We partition the vertices $V \setminus C$ of the graph G into four sets $V_{=0}(C), V_{=1}(C), V_{=2}(C), V_{\geq 3}(C)$ where $V_{=0}(C) = \{v \in V \setminus C \mid \deg(v) = 0\}$, $V_{=1}(C) = \{v \in V \setminus C \mid \deg(v) = 1\}$ (the *leaves*), $V_{=2}(C) = \{v \in V \setminus C \mid \deg(v) = 2\}$, and $V_{\geq 3}(C) = \{v \in V \setminus C \mid \deg(v) \geq 3\}$. The set $V_{=2}(C)$ is further partitioned into two sets $V^{\mathrm{e}}_{=2}(C), V^{\mathrm{t}}_{=2}(C)$ where $V^{\mathrm{e}}_{=2}(C)$ contains the ears and $V^{\mathrm{t}}_{=2}(C)$ the non-ears, called *transition*

[5] A *vertex cover* of an undirected graph $G = (V, E)$ is a vertex set $C \subseteq V$ such that every edge in E is incident to at least one vertex in C. The *vertex cover number* of G is the size of a smallest vertex cover of G.

vertices. Let $v \in V_{=1}(C)$ and let $c \in C$ denote its (unique) neighbor. We say that v is a *leaf* of c. Similarly, let $u \in V_{=2}(C)$ and let $c_a, c_b \in C$ denote its (unique) two neighbors. We say that u is a *transition vertex* (*ear*) of c_a and c_b if $u \in V^{\mathrm{t}}_{=2}(C)$ (if $u \in V^{\mathrm{e}}_{=2}(C)$). We often omit C if it is clear from the context.

Let $\mathcal{G} = (G = (V, E), \gamma, (\prec_i)_i)$ be a constrained level graph and let C be a vertex cover of G. The main challenge when constructing a constrained level planar drawing of $\mathcal{G}$ is the placement of the leaves, ears, and transition vertices (along with their incident edges). Indeed, it is not difficult to insert the isolated vertices (which include $V_{=0}$) in a post-processing step (performing a topological sort on each level), see Lemma 4. Moreover, since we may assume G to be planar, the size of $V_{\geq 3}(C)$ is linear in $|C|$. This well known bound can be derived, e.g., by combining the fact that the complement of a vertex cover is an independent set with the following statement (setting $A = C$).

▶ **Lemma 2** ([10, Corollary 9.25]). *Let $G = (V, E)$ be a planar graph and $A \subseteq V$. Then there are at most $2|A|$ connected components in the subgraph of G induced by $V \setminus A$ that are adjacent to more than two vertices of A.*

▶ **Corollary 3** (Folklore). *Let G be a planar graph and let C be a vertex cover of G. Then $|V_{\geq 3}(C)| \leq 2k$, where $k = |C|$.*

▶ **Lemma 4** ($\star$). *Let $\mathcal{G} = (G, \gamma, (\prec_i)_i)$ be a constrained level graph, let G' be the subgraph of G induced by the non-isolated vertices V', and let γ' and $(\prec'_i)_i$ be the restrictions of γ and $(\prec_i)_i$ to V', respectively. There is an algorithm that, given $\mathcal{G}$ and a constrained level planar drawing Γ' of $\mathcal{G}' = (G', \gamma', (\prec'_i)_i)$, constructs a constrained level-planar drawing of $\mathcal{G}$ in polynomial time.*

Our main algorithm will exploit the fact that only few ears may share a common level:

▶ **Lemma 5** ($\star$). *Let $\mathcal{G} = (G, \gamma)$ be a level graph, let Γ be a level planar drawing of $\mathcal{G}$, let C be a vertex cover of G. Then there are at most $2|C|$ ears with respect to C per level.*

Compatible edge orderings. Let Γ be a level planar drawing of a (possibly constrained) level graph $\mathcal{G} = (G = (V, E), \gamma)$ without isolated vertices. We will now define a useful (not necessarily unique) linear order $\prec^{\mathrm{e}}$ on the edges E with respect to Γ. We refer to $\prec^{\mathrm{e}}$ as an edge ordering of E that is *compatible* with Γ. Compatible edge orderings can be seen as a generalization of a linear order described in [27, Proof of Lemma 4.4] for a set of pairwise disjoint y-monotone paths, which in turn follows considerations about horizontal separability of y-monotone sets by translations [12, 3, 19, 20]. Intuitively, $\prec^{\mathrm{e}}$ is a linear extension of a partial order in which $e \in E$ precedes $f \in E$ if it is possible to shoot a horizontal rightwards ray from e to f in Γ without crossing other edges before reaching f. Formally, we say that a vertex v is *visible from the left* in Γ if the horizontal ray r_v emanating from v to the left intersects Γ only in v. We say that an edge $e = (u, v)$ is *visible from the left* in Γ if the closed (unbounded) region that is to the left of e and whose boundary is described by e, r_u, r_v intersects Γ only in e. The order $\prec^{\mathrm{e}}$ is now constructed as follows: the minimum of $\prec^{\mathrm{e}}$ is an edge e_1 of E that is visible from the left in Γ. Such an edge always exists [27, 19, 20]: among the edges whose lower endpoint is visible from the left, the edge with the topmost lower endpoint is visible from the left. Let Γ' denote the drawing derived from Γ by removing e_1 and any isolated vertices created by the removal of e_1. The restriction of $\prec^{\mathrm{e}}$ to the remaining edges $E \setminus e_1$ corresponds to an edge ordering compatible with Γ', which is constructed recursively. Note that $\mathcal{G}$ and $\prec^{\mathrm{e}}$ uniquely describe the drawing Γ and, given $\mathcal{G}$ and $\prec^{\mathrm{e}}$, it is possible to construct Γ in polynomial time (by traversing $\prec^{\mathrm{e}}$ in reverse).

3 Visibility extensions and cores

In this section, we introduce and study (refined) visibility extensions and cores of level planar drawings. We will see that the core-induced subdrawing of a (refined) visibility extension of a level planar drawing Γ with respect to some vertex cover C captures crucial structural properties of Γ while having a size that is bounded in $|C|$.

Visibility extensions. Let Γ be a level planar drawing of a level graph $\mathcal{G} = (G = (V, E), \gamma)$. A *visibility edge* e for Γ is (1) a y-monotone curve that joins two vertices of Γ and can be inserted into Γ without creating any crossings (but possibly a pair of parallel edges); or (2) a horizontal segment that joins two consecutive vertices on a common level of Γ and can be inserted into Γ without creating any crossings. A *visibility extension* of Γ with respect to a vertex set $V' \subseteq V$ is a drawing Λ derived from Γ by inserting a maximal set of pairwise non-crossing visibility edges incident only to vertices of V' such that for each pair e, e' of parallel edges in Λ there is at least one vertex of V' in the interior of the simple closed curve formed by e and e'; for an illustration see Figure 1(a). We remark that if $V' = V$, then Λ is essentially an interior triangulation containing Γ. However, we will always choose V' to be a (small) vertex cover, resulting in a much sparser yet still connected augmentation:

▶ **Lemma 6** (⋆). *Let $\mathcal{G} = (G = (V, E), \gamma)$ be a level graph without isolated vertices, let C be a vertex cover of G, let Γ be a level planar drawing of $\mathcal{G}$, let Λ be a visibility extension of Γ with respect to C, and let Λ_C be the subdrawing of Λ that is induced by C. Then Λ_C is connected and has $\mathcal{O}(k)$ edges, where $k = |C|$.*

Cores and refined visibility extensions. Intuitively, the core of a level planar drawing is a subset of the vertex set with certain crucial properties. To define it formally, we will first classify the ears of the drawing according to several categories. The concepts introduced in this paragraph are illustrated in Figure 1(b). Let $\mathcal{G} = (G = (V, E), \gamma)$ be a level graph, let C be a vertex cover of G, and let Γ be a level planar drawing of $\mathcal{G}$. Consider an ear $v \in V^{\mathrm{e}}_{=2}(C)$ with neighbors c_a, c_b where $\gamma(c_a) \geq \gamma(c_b)$. If $\gamma(v) > \gamma(c_a)$, we say that v is a *top* ear. Otherwise (if $\gamma(v) < \gamma(c_b)$), we say that v is a *bottom* ear. Assume that $\gamma(c_a) > \gamma(c_b)$ and that v is a top ear. If in Γ the edge $c_b v$ is drawn to the left (right) of c_a, we say that v is a *left* (*right*) ear in Γ. The terms "left" and "right" are defined analogously for bottom ears. If $\gamma(c_a) = \gamma(c_b)$, we consider v to be a left ear if it is a top ear; otherwise it is a right ear. Consider a pair $c_a, c_b \in C$ with at least one left ear in Γ and let Γ' denote the subdrawing of Γ induced by the set of edges that are incident to at least one left ear of c_a, c_b in Γ. Note that either all these ears are top ears or all these ears are bottom ears in Γ, and they are arranged in a nested fashion. In case of top (bottom) ears, we refer to the unique one with the largest (smallest) y-coordinate as the *outermost* left ear of c_a, c_b. The *innermost* left ear of c_a, c_b is defined symmetrically. If Γ' has an interior (i.e., bounded) face f such that the open region enclosed by the boundary of f contains a vertex of C in Γ, then we say that the two ears v_1, v_2 on the boundary of f are *bounding ears* of c_a, c_b in Γ with respect to C. Moreover, we say that v_1, v_2 are a pair of *matching* bounding ears whose *region* corresponds to f. The terms "outermost", "innermost" and "(region of matching pair of) bounding ears" are analogously defined for the right ears of c_a, c_b. Every vertex of Γ that is an outermost, innermost, or bounding ear (with respect to some pair $c_a, c_b \in C$) is called *crucial* with respect to C.

The *core* of Γ with respect to C is the (unique) subset of V that contains C, $V_{\geq 3}(C)$, as well as all crucial ears of Γ with respect to C. The subdrawing Λ_{core} of a visibility extension Λ of Γ with respect C that is induced by the core of Λ with respect to C captures crucial structural properties of Γ, which we will exploit in our main algorithm for constructing constrained level planar drawings in FPT-time. Due to the fact that Λ_{core} has only $\mathcal{O}(|C|)$ vertices and edges, it is not difficult to "guess" Λ_{core} in *XP*-time (via the process of exhaustive enumeration) when given $\mathcal{G}$ and C. The main bottleneck is the enumeration of all possible sets of crucial ears. To improve the runtime of this step to *FPT*-time, we will now describe a variant of visibility extensions that contains some additional ears, which take over the role of the original crucial vertices. Loosely speaking, one can create such a drawing by placing one or two new ears near each crucial ear in a visibility extension. The resulting augmentation retains the helpful structural properties of its underlying visibility extension and we will see that the (positions of the) crucial vertices of some such augmentation can be guessed more efficiently since we can restrict the possible levels of these new vertices to a small set. Formally, a *refined visibilty extension* Λ' of Γ is a crossing-free drawing of a level graph $\mathcal{G}' = (G' = (V', E'), \gamma')$ such that G is a subgraph of G', C is a vertex cover of G', every vertex in $V' \setminus V$ is an ear with respect to C and its incident edges are drawn as y-monotone curves, the subdrawing of Λ' induced by V is a visibility extension Λ of Γ, the crucial ears of Λ' are precisely the vertices in $V' \setminus V$, and $|V' \setminus V| \in \mathcal{O}(|C|)$.

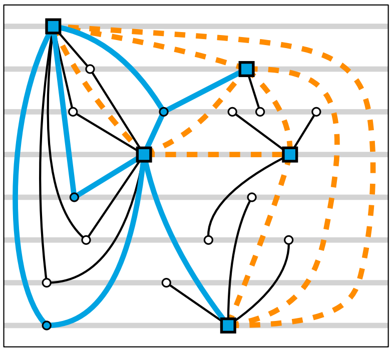
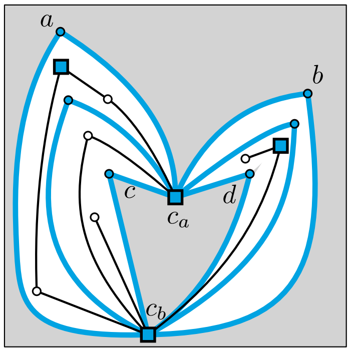
(a)  (b) 

Figure 1 In this (and all other) figure(s), filled square vertices belong to a vertex cover C of the depicted graph and filled (round or square) vertices belong to the core of the shown drawing with respect to C. (a) A drawing Γ (non-dashed edges) is augmented with visibility edges (dashed) to obtain a visbility extension Λ with respect to C (note that this augmentation is not unique). The thick (non-dashed or dashed) edges and filled vertices represent Λ_{core}. (b) All filled round vertices are top crucial ears of c_a and c_b. All of them are bounding ears except for b and c. The vertex a / b / c / d is an outermost left / outermost right / innermost left / innermost right ear.

▶ **Lemma 7** ($\star$). *Let $\mathcal{G} = (G = (V, E), \gamma)$ be a level graph without isolated vertices, let C be a vertex cover of G, let Γ be a level planar drawing of $\mathcal{G}$, let Λ be a refined or non-refined visibility extension of Γ with respect to C, and let Λ_{core} the subdrawing of Λ induced by the core of Λ with respect to C. Then Λ_{core} is connected and has $\mathcal{O}(k)$ vertices and $\mathcal{O}(k)$ edges, where $k = |C|$.*

4 Algorithm

In this section, we describe the algorithm corresponding to the proof of Theorem 1. Let $\mathcal{G} = (G = (V, E), \gamma, (\prec_i)_i)$ be a constrained level graph and let C be a vertex cover of G. Our goal is to construct a constrained level planar drawing of $\mathcal{G}$ or correctly report that such a drawing does not exist. In view of Lemma 4, we may assume that $\mathcal{G}$ has no isolated vertices. To construct the desired drawing, we proceed in three main steps. In Step 1, we "guess" a core-induced subdrawing of a refined visibility extension of a constrained level planar drawing of $\mathcal{G}$ with respect to C (via the process of exhaustive enumeration). In Step 2, we augment our drawing by inserting the transition vertices of $\mathcal{G}$ with respect to C. In Step 3, we finalize our drawing by inserting the leaves and ears of $\mathcal{G}$ with respect to C.

Step 1: Guessing a core-induced subdrawing. Assume that there is a constrained level planar drawing Γ^* of $\mathcal{G}$, let Λ^* be a refined visibility extension of Γ^* with respect to C, and let Λ_{core} be the subdrawing of Λ^* induced by the core of Λ^* with respect to C. The procedures corresponding to Steps 2 and 3 of our algorithm are guaranteed to produce a constrained level planar drawing (not necessarily Γ^*) of $\mathcal{G}$ when given Λ_{core}. Hence, the goal of Step 1 is to determine (or, rather, guess) Λ_{core}, given $\mathcal{G}$ and C. More precisely, we will construct a set $\mathcal{F}$ of drawings such that $\Lambda_{\text{core}} \in \mathcal{F}$ and the number $|\mathcal{F}|$ of drawings in $\mathcal{F}$ is sufficiently small. For each drawing in $\mathcal{F}$, we then apply Steps 2 and 3 of the algorithm (incurring a factor of $|\mathcal{F}|$ in the total running time). Given that $\Lambda_{\text{core}} \in \mathcal{F}$, one of the iterations is guaranteed to terminate with a constrained level planar drawing of $\mathcal{G}$.

▶ **Lemma 8** ($\star$). *Let $\mathcal{G} = (G = (V, E), \gamma, (\prec_i)_i)$ be a constrained level graph without isolated vertices, let C be a vertex cover of G, and let Γ^* be a constrained level planar drawing of $\mathcal{G}$. There is an algorithm that, given $\mathcal{G}$ and C, constructs a set $\mathcal{F}$ of $2^{\mathcal{O}(k \log k)}$ drawings in $2^{\mathcal{O}(k \log k)} \cdot n^{\mathcal{O}(1)}$ time, where $n = |V|$ and $k = |C|$, such that all drawings in $\mathcal{F}$ have size $\mathcal{O}(k)$ and are level planar drawings of subgraphs of G induced by C and $V_{\geq 3}(C)$ that respect γ and the orderings $\prec_i$ and are augmented by some visibility edges and additional ears (with respect to C). Further, there exists a refined visibility extension Λ^* of Γ^* such that the subdrawing Λ_{core} of Λ^* induced by the core of Λ^* with respect to C is contained in $\mathcal{F}$.*

Proof sketch. We introduce the following terminology: let x be a vertex in a level planar drawing (possibly augmented by some horizontal edges). Let ℓ be the y-coordinate of x and let ℓ' be the largest y-coordinate of a vertex below x (if there no such vertex, we set $\ell' = \ell - 1$). We say that the line $L_{(\ell+\ell')/2}$ is *directly below x*. The line *directly above x* is defined symmetrically.

We proceed in two main steps. In the first main step, we show that there *exists* a refined visibility extension Λ^* of Γ^*. To this end, we start with a visibility extension Λ of Γ^* and describe an incremental strategy that performs a total of $\mathcal{O}(k)$ augmentation steps, in each of which a new ear is added that takes over the role of a crucial ear in Γ^*. (The description of this first main step is deferred to the appendix of the full preprint version [28].) In the second main step, we discuss the construction of the desired family $\mathcal{F}$. To this end, let Λ_{core} be the subdrawing of Λ^* induced by the core of Λ^* with respect to C. The drawing Λ_{core} is uniquely described by $\mathcal{G}$, C, the set of visibility edges of Λ_{core} (and Λ^*), the set of crucial ears of Λ_{core} (and Λ^*) together with their level assignments and their incident edges, and a compatible edge ordering of the nonhorizontal edges of Λ_{core}. The graph $\mathcal{G}$, as well as the vertex cover C are given, so it suffices to enumerate all possible options for the remaining elements.

There are $m_{\text{vis}} \in \mathcal{O}(k)$ visibility edges by Lemma 7 and each of these visibility edges joins a pair of vertices in C. Hence, there are at most $\binom{k}{2}^{m_{\text{vis}}} \subseteq k^{\mathcal{O}(k)} \subseteq 2^{\mathcal{O}(k \log k)}$ possible options for choosing the set of visibility edges. To enumerate the set of crucial ears along with their level assignments, we mimic the aforementioned incremental strategy for constructing Λ^*: we first enumerate all options to pick the pair of neighbors of the first new vertex along with its level, then, for each of these options, we enumerate all options to pick the pair of neighbors of the the second vertex along with its level, etc., until we have obtained all options to pick the desired $\mathcal{O}(k)$ vertices together with their levels. More precisely, suppose we have already enumerated all options to pick the first i vertices together with their neighbors and levels. For each of these options, to enumerate all options to pick the next vertex v', we go through all ways to pick its two neighbors $u, w \in C$ and through all ways to pick the level of v'. There are $\mathcal{O}(k)$ pairs of vertices in C with ears (by Lemma 7). To bound the number of ways to pick the level of v', we make use of the fact that whenever the incremental strategy for constructing Λ^* places a new vertex v', it is assigned to a new level directly above or below a level of one of the following categories:

- a level with a vertex in C ($\mathcal{O}(k)$ possibilities),
- a level with a vertex in $V_{\geq 3}(C)$ ($\mathcal{O}(k)$ possibilities by Corollary 3),
- a level of a vertex that does not belong to $\mathcal{G}$, i.e., a level used for one of the already placed vertices ($\mathcal{O}(i) \subseteq \mathcal{O}(k)$ possibilities),
- a level with a top-most or bottom-most vertex of $\mathcal{G}$ ($\mathcal{O}(1)$ possibilities),
- a level with a top-most top ear, a top-most bottom ear, a bottom-most top ear, or a bottom-most bottom ear of some pair of vertices in C ($\mathcal{O}(k)$ possibilities by Lemma 7),
- a level with a top-most or bottom-most vertex of a connected component that contains a vertex of C in the graph obtained by removing u and w from the current graph ($\mathcal{G}$ together with the visbility edges and the already added vertices) ($\mathcal{O}(k)$ possibilities).

In total, for a fixed pair of neighbors u, w, there are thus $\mathcal{O}(k)$ options to pick a level for v'. We immediately discard level assignments for which v' is no ear. By multiplying with the number of ways to choose the neighbors, we obtain $\mathcal{O}(k^2)$ options to choose v' and its level. Multiplying the number of options for all $\mathcal{O}(k)$ steps together, we obtain a total number of $k^{\mathcal{O}(k)} \subseteq 2^{\mathcal{O}(k \log k)}$ ways to create the set of crucial ears along with their levels. By multiplying with the number of ways to choose the visibility edges, we obtain a total of $2^{\mathcal{O}(k \log k)}$ options to choose the graph that corresponds to Λ_{core}. For each of these options we enumerate all $k^{\mathcal{O}(k)} \subseteq 2^{\mathcal{O}(k \log k)}$ permutations of the set of non-horizontal edges and, interpreting the permutation as a compatible edge order, try to construct a level planar drawing for which this order is compatible (cf. Section 2). If we succeed, we check whether the drawing is conform with $(\prec_i)_i$ and can be augmented with the horizontal visibility edges. If so, we include the drawing in the set $\mathcal{F}$ of reported drawings. The size of the thereby constructed set $\mathcal{F}$ is bounded by $2^{\mathcal{O}(k \log k)}$ and it is guaranteed to contain Λ_{core} by construction. ◀

Step 2: Inserting transition vertices. We now describe how to insert the transition vertices into the core-induced subdrawing Λ_{core} of the (refined) visibility extension Λ^*. Our plan is to first show that in Λ^*, every transition vertex is placed "very close to" some visibility edge. Intuitively, this means that the visibility edges of Λ_{core} act as placeholders near which the transition vertices have to be placed. We will describe a procedure that does so while carefully taking into account the given partial orderings $\prec_i$ and prove its correctness by means of an (somewhat technical) exchange argument. To formalize the notion of "very close to", let e be an edge of a level planar drawing joining two vertices a, b such that there is a degree-2 vertex t with neighbors a, b and $\gamma'(b) < \gamma'(t) < \gamma'(a)$ where γ' is the level assignment. We say that t is drawn in the *vicinity* of e with respect to a vertex set V' if the simple closed curve formed by e and the two edges incident to t does not contain a vertex of V' in its interior.

▶ **Lemma 9** (⋆). *Let $\mathcal{G} = (G = (V, E), \gamma, (\prec_i)_i)$ be a constrained level graph without isolated vertices, let C be a vertex cover of G, let Γ^* be a constrained level planar drawing of $\mathcal{G}$, let Λ^* be a refined or non-refined visibility extension of Γ^* with respect to C, and let Λ_{core} the subdrawing of Λ^* induced by the core of Λ^* with respect to C. There is an algorithm* *that, given $\mathcal{G}$, C and Λ_{core}, inserts all transition vertices $V^{t}_{=2}(C)$ (and their incident edges) into vicinities (with respect to C) of visibility edges in Λ_{core} in polynomial time such that the resulting drawing $\Lambda^{t}_{\text{core}}$ can be extended to a drawing whose restriction to G is a constrained level planar drawing of $\mathcal{G}$.*

Step 3: Inserting leaves and ears. In this step, we start with the output of Step 2 and finalize our drawing by placing all the vertices that are still missing.

▶ **Lemma 10.** *Let $\mathcal{G} = (G = (V, E), \gamma, (\prec_i)_i)$ be a constrained level graph without isolated vertices, let C be a vertex cover of G, let Γ^* be a constrained level planar drawing of $\mathcal{G}$, let Λ^* be a refined or non-refined visibility extension of Γ^* with respect to C in which each transition vertex is placed in the vicinity of some visibility edge with respect to C, and let Λ_{core} ($\Lambda^{t}_{\text{core}}$) be the subdrawing of Λ^* induced by the core (and the transition vertices) of Λ^* with respect to C. There is an algorithm that, given $\mathcal{G}$, C, Λ_{core} and $\Lambda^{t}_{\text{core}}$, extends Λ_{core} to a drawing Γ whose restriction to G is a constrained level planar drawing of $\mathcal{G}$ in $2^{\mathcal{O}(k^2 \log k)} n^{\mathcal{O}(1)}$ time, where $n = |V|, k = |C|$.*

Proof. The only vertices of G that are missing in $\Lambda^{t}_{\text{core}}$ are the leaves and the non-crucial ears with respect to C (in case Λ^* is a *refined* visibility extension, the non-crucial ears are exactly the ears of $\mathcal{G}$). Our plan to insert them into our drawing is as follows. We begin by introducing more structure in $\Lambda^{t}_{\text{core}}$ and Λ^* by adding some additional visibility edges and making some normalizing assumptions, which will simplify the description of the upcoming steps. In particular, this step will ensure that for each missing ear, there are essentially only (up to) two possible placements, which will allow us to enumerate all possible ear placements (so-called ear orientations) on a given level in FPT-time. We then describe a partition of the plane into so-called cells in a way that is very reminiscent of the well-known trapezoidal decomposition from the field of computational geometry, cf. [11]. We merge some cells into so-called channels, which correspond to connected y-monotone regions in which the missing leaves along with their incident edges will be drawn (a region is *y-monotone* if its intersection with every horizontal line is connected). We then introduce (and describe an enumerative process that constructs in FPT-time) a so-called traversal sequence that is compatible with Λ^*, which is a sequence of sets of channels with several useful structural properties related to Λ^*. In particular, this sequence, in some sense, sweeps the plane from left to right in a way where for each edge incident to a leaf in Λ^*, at some point there is a channel that contains it. Exploiting the properties of the traversal sequence, we then describe how to construct a so-called insertion sequence for the leaves on a given level with respect to a given placement of the ears of that level in polynomial time. Such an insertion sequence does not necessarily exist for every placement of ears, but we are guaranteed to find one by enumerating all possible ear placements of the level. This computation is performed independently for each level. Finally, we show how to construct in polynomial time the desired drawing Γ when given an insertion sequence along with its ear placement for each level. Notably, the final step can be executed even if some of the ear placements are different from the ones used in Λ^*. Let us proceed to formalize these ideas.

Augmenting and normalizing Λ_{core}, $\Lambda^{t}_{\text{core}}$, and Λ^*. Let $e = (u, v)$ be a visibility edge of Λ^* (and $\Lambda^{t}_{\text{core}}$) that has at least one transition vertex in its vicinity. In both $\Lambda^{t}_{\text{core}}$ and Λ^*, we add two copies of e; one directly to the left of the leftmost transition vertex and the other directly to the right of the rightmost transition vertex in the vicinity of e, which is possible to do in a y-monotone fashion and without introducing any crossings; see Figure 2(a). Note that the region enclosed by these two edges only contains transitions vertices of u and v, as well as leaves of u or v. We repeat this operation for all visibility edges e.

The following steps are illustrated in Figure 2(b). Let f be a face of Λ^t_{core} that is bounded by four vertices v_1, v_2, u_1, u_2 where $v_1, v_2 \in C$ and $u_1, u_2 \in V_{\{v_1,v_2\}}$ are either both left ears or both right ears. Without loss of generality, assume they are both left ears with $\gamma(v_1) \le \gamma(v_2) < \gamma(u_1) < \gamma(u_2)$. We add copies of the edges (v_1, u_2) and (v_2, u_2) in f in both $\Lambda^{\text{t}}_{\text{core}}$ and Λ^*, which can be done without introducing crossings. These edges partition f into three regions. Note that in Λ^* these regions only contain leaves and non-crucial ears. Without loss of generality, we will assume that each leaf in f is either placed in the region bounded by (v_1, u_2) and its copy or the region bounded by (v_2, u_2) and its copy (note that a leaf v that is adjacent to v_1 cannot have a constraint of the form $w \prec_i v$, where w is a non-crucial ear in f; the situation for leaves adjacent to v_2 is symmetric). Thus, the remaining (central) third region only contains non-crucial ears and is henceforth called an *ear-face* of $\Lambda^{\text{t}}_{\text{core}}$. We repeat this modification for all faces such as f.

For the remainder of the proof, $\Lambda^{\text{t}}_{\text{core}}$ and Λ^* are used to refer to the thusly augmented and normalized drawings. We also add all the new edges to Λ_{core} and use Λ_{core} to refer to this augmentation. Note that this implies that it now suffices to search for a drawing of $\mathcal{G}$ in which every non-crucial ear is placed in an ear-face, whereas no leaf is placed in an ear-face.

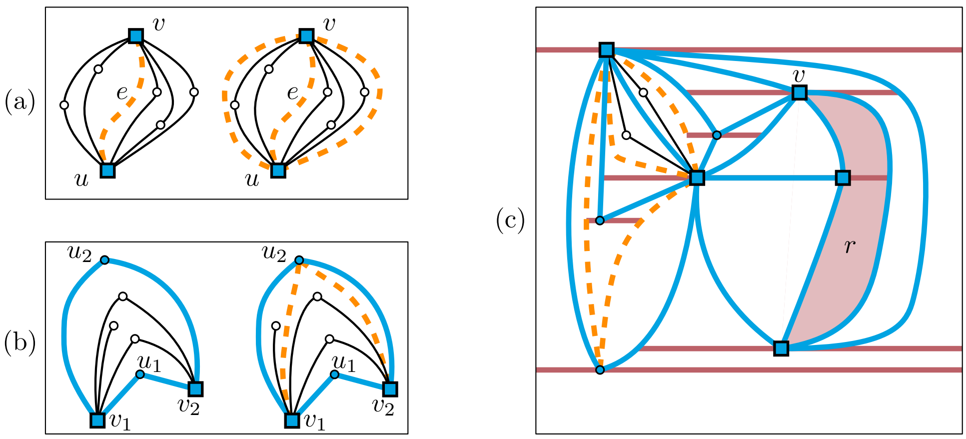


Figure 2 Like in all figures, filled square vertices belong to a vertex cover C of the depicted graph and filled (round or square) vertices belong to the core of the shown drawing with respect to C. Figures (a) and (b) visualize how Λ_{core}, $\Lambda^{\text{t}}_{\text{core}}$, and Λ^* are augmented by (a) enclosing transition vertices and (b) creating ear-faces with visibility edges (dashed). Moreover, (b) illustrates our normalizing assumption, i.e., the leaf can be moved to the exterior of the ear-face without violating any constraints. (c) The drawing $\Lambda^{\text{t}}_{\text{core}}$ with its additional visibility edges (dashed) from the augmentation step and the horizontal rays and segments (thick, red) from the cell decomposition. The shaded region corresponds to a channel (v, r, R) from $v \in C$ to the cell r with $|R| = 2$.

Decomposition into cells. We will now describe a partition of the plane that essentially corresponds to a trapezoidal decomposition (cf. [11]) of Λ_{core}; for illustrations refer to Figure 2(c): for each vertex v in Λ_{core}, shoot a horizontal ray from v to the left until hitting an edge or vertex of Λ_{core}, then add the corresponding segment to Λ_{core}. In case the ray does not intersect any part of Λ_{core}, add the ray itself to Λ_{core}. Perform a symmetric augmentation by shooting a horizontal ray from v to the right. The maximal connected regions of the resulting partition of the plane are henceforth called *cells*. We consider the cells to be closed. Note that each cell is y-monotone and bounded by up to two horizontal segments or rays and up to two y-monotone curves. By Lemma 7, Λ_{core} has $\mathcal{O}(k)$ vertices and edges (note that the augmentation step copies each edge at most twice) and, hence, it has $\mathcal{O}(k)$ faces. Consequently, the number of cells is also $\mathcal{O}(k)$ since the insertion of a single segment or ray can only increase the number of faces (or, rather, maximal connected regions) by one.

Channels. Let $v \in C$, let R be a set of cells that do not belong to ear-faces, and let $r \in R$. Further, assume that R contains a cell that is incident to v. The triple $c = (v, r, R)$ is a *channel from v to r* if it is possible to draw a y-monotone curve in $\Lambda^{\mathrm{t}}_{\mathrm{core}}$ in the interior of the union of R that intersects each cell in R and does not cross any edge of $\Lambda^{\mathrm{t}}_{\mathrm{core}}$; as illustrated in Figure 2(c). We say that *c can be used* by a leaf $w \in V_{\{v\}}$ and that the edge e_w incident to *w can be drawn in c* if e_w can be drawn in $\Lambda^{\mathrm{t}}_{\mathrm{core}}$ in the union of R without any crossings and there is no channel (v, r', R') with this property for which $R' \subset R$. Further, we say that *c is used* by w if it can be used by w and the edge incident to w is drawn in the union of R in Λ^*. We use U to denote the set of all channels and $U_{\mathrm{used}} \subseteq U$ to denote the set of all channels that are used. The connectivity of $\Lambda^{\mathrm{t}}_{\mathrm{core}}$ (cf. Lemma 7) can be used to show:

▷ Claim 11 ($\star$). $|U_{\mathrm{used}}| \leq |U| \in \mathcal{O}(k^2)$.

Traversal sequences. Let $\mathcal{U} = (\mathcal{U}_1, \mathcal{U}_2, \ldots, \mathcal{U}_m)$ be a sequence of sets of channels. We say $\mathcal{U}$ is a *traversal sequence* if the following properties are fulfilled (see Figure 3 for illustrations):

(T1) Let $i \in [m]$ and let $(v, r, R) \in \mathcal{U}_i$ and $(v', r', R') \in \mathcal{U}_i$ with $v \neq v'$. Then the intersection of the line $L_{\gamma(v)}$ with the interior of the union of R' is empty.

(T2) Let c be a channel and let $a \leq b$ be two indices such that $c \in \mathcal{U}_a$ and $c \in \mathcal{U}_b$. Then for every $a \leq i \leq b$, $c \in \mathcal{U}_i$.

We say a channel u is *active* in $\mathcal{U}_i$ if it is contained in it, and otherwise it is *inactive* in it. We say a traversal sequence $\mathcal{U} = (\mathcal{U}_1, \mathcal{U}_2, \ldots, \mathcal{U}_m)$ is *compatible* with Λ^* if the following conditions are satisfied (refer again to Figure 3 for illustrations):

(C1) For every channel c, there exists an $i \in [m]$ such that $c \in \mathcal{U}_i$ if and only if c is used.

(C2) There exists a compatible edge ordering $\prec^{\mathrm{e}}$ for the restriction of Λ^* to its nonhorizontal edges (recall that some visibility edges are horizontal) such that:

  a. Let e, e' be two edges that are incident to leaves and where $e \prec^{\mathrm{e}} e'$, let c be the channel used by e, and let c' be the channel used by e'. Then there exist indices i, i' such that $i \leq i'$ and $c \in \mathcal{U}_i$, $c' \in \mathcal{U}_{i'}$.

  b. Let $c_1, c_2 \in U_{\mathrm{used}}$ such that for every edge e_1 using c_1 and for every edge e_2 using c_2 we have $e_1 \prec^{\mathrm{e}} e_2$. Then there is no index $i \in [m]$ such that $\mathcal{U}_i$ contains both c_1 and c_2 (and every index for which c_1 is active is smaller than every index where c_2 is active).

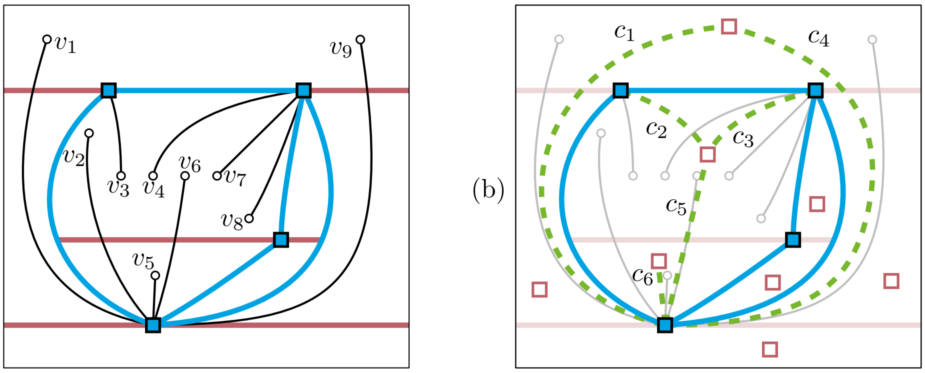
  c. For every pair of used channels c_1, c_2 such that c_2 is being used by an edge e that succeeds all edges that use c_1 in $\prec^{\mathrm{e}}$ there exists an index i such that $c_2 \in \mathcal{U}_i$ and $c_1 \notin \mathcal{U}_i$ (and c_1 is active for some index smaller than i).

▷ Claim 12 ($\star$). There exists a traversal sequence $\mathcal{U} = (\mathcal{U}_1, \mathcal{U}_2, \ldots, \mathcal{U}_m)$ that is compatible with Λ^* and whose length is $m \in \mathcal{O}(k^2)$. Moreover, there is an algorithm that, given $\mathcal{G}$, C and $\Lambda^{\mathrm{t}}_{\mathrm{core}}$, computes a set of $2^{\mathcal{O}(k^2 \log k)}$ traversal sequences that contains $\mathcal{U}$ in $2^{\mathcal{O}(k^2 \log k)} n^{\mathcal{O}(1)}$ time.

▷ Claim 13 ($\star$). Let $\mathcal{U} = (\mathcal{U}_1, \mathcal{U}_2, \ldots, \mathcal{U}_m)$ be a traversal sequence that is compatible with Λ^*, let $i \in [m]$, and let v be a leaf. Then $\mathcal{U}_i$ contains at most channel that can be used by v.

Ear orientations. Let $i \in [h]$ and let $V_i^{\mathrm{e}} \subseteq V_i$ be all non-crucial ears on the ith level. Further, consider a mapping $s \colon V_i^{\mathrm{e}} \to \{\text{left}, \text{right}\}$. We say that s is an *ear orientation* of level i. We say that s is *valid* if it is possible to insert the ears V_i^{e} (on the line L_i) along with their incident edges into $\Lambda^{\mathrm{t}}_{\mathrm{core}}$ such that the resulting drawing Λ is crossing-free, no constraint is violated (i.e., if $u \prec_i v$, then u is placed to the left of v), and for every $v \in V_i^{\mathrm{e}}$, we have that v is a left ear if and only if $s(v) = \text{left}$. We say that Λ is *induced* by s. Note that for any ear $v \in V_i^{\mathrm{e}}$ there is at most one left ear-face and at most one right ear-face into which it can be

(a) (b)



(c)

	$\mathcal{U}_1$	$\mathcal{U}_2$	$\mathcal{U}_3$	$\mathcal{U}_4$	$\mathcal{U}_5$	$\mathcal{U}_6$	$\mathcal{U}_7$	$\mathcal{U}_8$	$\mathcal{U}_9$
c_1									
c_2									
c_3									
c_4									
c_5									
c_6									

Figure 3 Like in all figures, filled square vertices belong to a vertex cover C of the depicted graph. (a) The drawing Λ^* together with its cell decomposition. (b) The dashed (green) "edges" represent the used channels $c_1 \dots, c_6$ of Λ^* leading from vertices of C to cells in R, which are represented by unfilled (red) squares. (c) A traversal sequence compatible with Λ^* for which Property (C2) is satisfied for any compatible edge ordering that contains the edges of $v_1, v_2, \dots, v_9$ in this order.

inserted without introducing crossings. Hence, a valid ear orientation uniquely describes the ear-face in which each ear is placed. Further, note that no two ears of V_i^{e} can be placed in the same ear-face without introducing crossings. In contrast, whenever an ear orientation assigns only one ear to a given ear-face, it is possible to place the ear without introducing crossings. These properties make it easy to test whether a given ear orientation is valid and, if so, construct the (unique) induced drawing in polynomial time. In view of Lemma 5, this means we can enumerate all valid ear orientations of a given level in $2^{2k} n^{\mathcal{O}(1)}$ time.

Insertion sequences. Let $\mathcal{U} = (\mathcal{U}_1, \mathcal{U}_2, \dots, \mathcal{U}_m)$ be a traversal sequence that is compatible with Λ^*. Further, let $i \in [h]$, let s be a valid ear orientation of level i, and let Λ be its induced drawing. Finally, let $\mathcal{Q} = (Q_1, Q_2, \dots, Q_q)$ be a sequence with $Q_t = (v, j)$, $v \in V_i \cap V_{=1}$, and $1 \le j \le m$ for all $1 \le t \le q$. We say $\mathcal{Q}$ is an *insertion sequence* for i, s, and $\mathcal{U}$ if the following conditions are fulfilled (for an example, see Figure 4):

(I1) Let $v \in V_i \cap V_{=1}$. Then there exists at most one index t such that $v \in Q_t$.

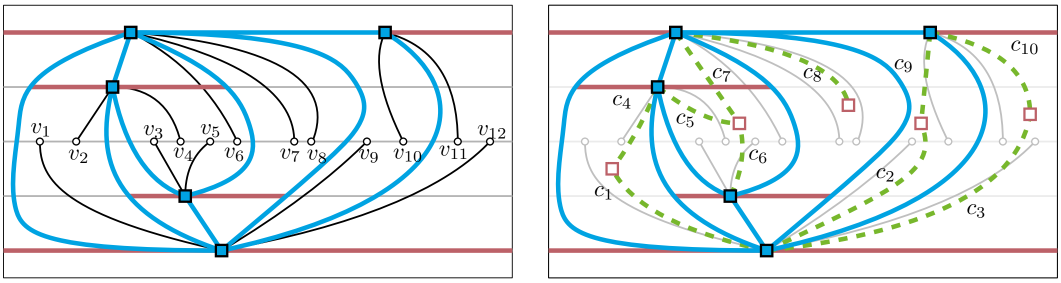
(I2) Let $Q_x = (v_x, j_x) \in \mathcal{Q}$ and $Q_y = (v_y, j_y) \in \mathcal{Q}$ with $x < y$. Then $j_x \le j_y$.

(I3) Let $Q_x = (v, j) \in \mathcal{Q}$. Then there exists a channel $c \in \mathcal{U}_j$ that can be used by v.

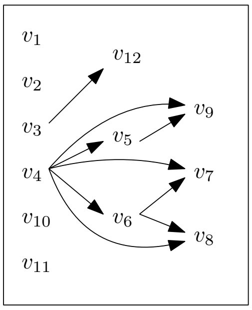
(I4) Let $Q_x = (v, j) \in \mathcal{Q}$. Then for every $w \in V_i \cap V_{=1}$ with $w \prec_i v$, there exists an index $x' < x$ such that $w \in Q_{x'}$.

(I5) Let $Q_x = (v, j) \in \mathcal{Q}$ and let $(v, r, R) \in \mathcal{U}_j$ be the (unique, by Claim 13) channel usable by v in $\mathcal{U}_j$. Then for every $w \in V_i \setminus V_{=1}$ that is not a transition vertex in r and where $v \prec_i w$, w is to the right of r or on the right boundary of r. Symmetrically, for every $w \in V_i \setminus V_{=1}$ that is not a transition vertex in r and where $w \prec_i v$, w is to the left of r or on the left boundary of r.

Let $\mathcal{Q} = (Q_1, Q_2, \dots, Q_q)$ be an insertion sequence for i, s, and $\mathcal{U}$. We say a leaf $v \in V_i \cap V_{=1}$ is *choosable* with regard to $\mathcal{Q}$ if (i) there exists an index j, such that $(Q_1, Q_2, \dots, Q_q, (v, j))$ is an insertion sequence for i, s, and $\mathcal{U}$ as well and (ii) there exists no pair v', j', with $v' \in V_i \cap V_{=1}$ and $j' < j$ such that $\mathcal{Q} = (Q_1, Q_2, \dots, Q_q, (v', j'))$ is an insertion sequence.



(a) (b)



	$\mathcal{U}_1$	$\mathcal{U}_2$	$\mathcal{U}_3$	$\mathcal{U}_4$	$\mathcal{U}_5$	$\mathcal{U}_6$	$\mathcal{U}_7$	$\mathcal{U}_8$	$\mathcal{U}_9$	$\mathcal{U}_{10}$	$\mathcal{U}_{11}$	$\mathcal{U}_{12}$
c_1	v_1											
c_2									v_9 v_{12}			
c_3												
c_4		v_2										
c_5				v_4								
c_6			v_3	v_5								
c_7						v_6 v_7 v_8						
c_8												
c_9										v_{11} v_{10}		
c_{10}												

(c) (d)

Figure 4 Like in all figures, filled square vertices belong to a vertex cover C of the depicted graph. (a) The drawing Λ^* together with its cell decomposition. (b) The dashed (green) "edges" represent the used channels $c_1, \dots, c_{10}$ of Λ^* leading from vertices of C to cells in R represented by unfilled (red) squares. Figure (c) shows the constraints between the vertices of the third level depicted in (a) and Figure (d) illustrates the insertion sequences $\mathcal{Q} = ((v_1, 1), (v_2, 2), (v_3, 3), (v_4, 4), (v_5, 4), (v_6, 6), (v_7, 6), (v_8, 6), (v_9, 9), (v_{12}, 9), (v_{11}, 10), (v_{10}, 10))$ for this level and the depicted traversal sequence $\mathcal{U} = (\mathcal{U}_1, \dots, \mathcal{U}_{12})$. Note that the order of the vertices in $\mathcal{Q}$ is different from the one in Λ^*.

▷ Claim 14 (★). Let $\mathcal{U} = (\mathcal{U}_1, \mathcal{U}_2, \dots, \mathcal{U}_m)$ be a traversal sequence that is compatible with Λ^* and let $\prec^{\mathrm{e}}$ be a compatible edge ordering for the restriction of Λ^* to its nonhorizontal edges, for which Property (C2) is fulfilled for $\mathcal{U}$. Further, let $i \in [h]$ and let s be the valid ear orientation of level i that is used in Λ^*. For every $q \in \{0, 1, \dots, |V_i \cap V_{=1}|\}$, there exists an insertion sequence $\mathcal{Q}_q = (Q_1, Q_2, \dots, Q_q)$ for i, s, and $\mathcal{U}$ such that the following two properties are satisfied (for an example, see Figure 4):
Interval property: For every vertex $v \in V_i \cap V_{=1}$ there exists at most one nonempty maximal interval $[a, b]$ where $0 \le a \le b \le q$ such that v is choosable with regard to $\mathcal{Q}_j$ if and only if $a \le j \le b$. If such an interval $[a, b]$ exists, then $b = q$ or $v \in Q_{b+1}$. Conversely, if v occurs in some entry of $\mathcal{Q}_q$, then the interval exists.
Dominance property: Let $\prec_i^l$ be the restriction of $\prec^{\mathrm{e}}$ to edges incident to leaves on level i. Further, let $Q_\ell = (v_\ell, j_\ell) \in \mathcal{Q}_q$, let $v_{\ell'} \in V_i \cup V_{=1}$, and let e_ℓ $(e_{\ell'})$ be the edge incident to v_ℓ $(v_{\ell'})$. Then, if $e_\ell \prec_i^l e_{\ell'}$ or $v_\ell = v_{\ell'}$, we have $j_\ell \le j'$, where j' is the maximum index such that the channel c' used by $v_{\ell'}$ (in Λ^*) is in $\mathcal{U}_{j'}$.
Moreover, $\mathcal{Q}_k$ is a prefix of $\mathcal{Q}_{k+1}$ for all $0 \le k \le |V_i \cap V_{=1}| - 1$. Finally, there is an algorithm that, given $\mathcal{G}$, C, $\Lambda^{\mathrm{t}}_{\mathrm{core}}$, $\mathcal{U}$, i and s, computes $\mathcal{Q}_{|V_i \cap V_{=1}|}$ in polynomial time.

Computing the drawing. When given a traversal sequence that is compatible with Λ^*, we can utilize Claim 14 to obtain a valid ear orientation together with an insertion sequence for a given level by simply trying to apply the algorithm corresponding to Claim 14 for all valid ear orientations of that level. We do so for each level and then use the gathered information to construct the desired drawing by means of the following claim. We remark that when the algorithm corresponding to Claim 14 successfully terminates, it is guaranteed to return an insertion sequence for the given valid ear orientation. It might output an insertion sequence even if the given valid ear orientation is not the one used in Λ^*. However, this does not invalidate our strategy as the following claim does *not* require that the given ear orientations are the ones used in Λ^*.

▷ Claim 15 (⋆). There is an algorithm that, given $\mathcal{G}$, C, Λ_{core}, $\Lambda^{\text{t}}_{\text{core}}$, a traversal sequence $\mathcal{U} = (\mathcal{U}_1, \mathcal{U}_2, \ldots, \mathcal{U}_m)$ that is compatible with Λ^*, and, for each level $i \in [h]$, a valid ear orientation s^i, as well as an insertion sequence $\mathcal{Q}^i = (Q^i_1, Q^i_2, \ldots, Q^i_{q^i})$ for i, s^i, and $\mathcal{U}$ such that $q^i = |V_i \cap V_{=1}|$ (that is, $\mathcal{Q}^i$ contains all leaves of level i), computes an extension of Λ_{core} whose restriction to G is a constrained level planar drawing of $\mathcal{G}$ in polynomial time.

Wrap-up. In the beginning of (and throughout the) proof of Lemma 10, we have already sketched how the individual pieces of the proof fit together. We formally summarize our strategy in the proof of the following claim.

▷ Claim 16 (⋆). There is an algorithm that, given $\mathcal{G}$, C, Λ_{core} and $\Lambda^{\text{t}}_{\text{core}}$, extends Λ_{core} to a drawing Γ whose restriction to G is a constrained level planar drawing of $\mathcal{G}$ in $2^{\mathcal{O}(k^2 \log k)} n^{\mathcal{O}(1)}$ time.

This concludes the proof of Lemma 10. ◀

Summary. In the beginning of Section 4, we have already sketched how Lemmas 4 and 8–10 can be combined to obtain Theorem 1. We formally summarize:

▶ **Theorem 17** (⋆). *There is an algorithm that, given a constrained level graph $\mathcal{G} = (G = (V, E), \gamma, (\prec_i)_i)$ and a vertex cover C of G, either constructs a constrained level planar drawing of $\mathcal{G}$ or correctly reports that such a drawing does not exist in time $2^{\mathcal{O}(k^2 \log k)} \cdot n^{\mathcal{O}(1)}$, where $n = |V|$ and $k = |C|$.*

Given that a smallest vertex cover can be obtained in FPT-time with respect to its size [10], our main result (Theorem 1) follows from Theorem 17.

5 Discussion

We have shown that CLP is FPT when parameterized by the vertex cover number. A speed-up to polynomial time or a generalization to the smaller graph parameters (in particular, tree-depth, path-width, tree-width, and feedback vertex set number) is not possible, even if restricting to OLP or proper CLP instances.

Recall from Section 1 that in the LEVEL PLANARITY variant PARTIAL LEVEL PLANARITY (PLP), a given level planar drawing of a subgraph of the input graph $\mathcal{G}$ has to be extended to a full drawing of $\mathcal{G}$, which can be seen as a generalization of OLP and, in the proper case, a specialization of CLP. An instance of PLP can always be turned into an equivalent proper instance (and, thus, a CLP instance) by subdividing each edge on each level it passes through. However, in general this operation will (dramatically) increase the vertex cover number of the instance. Hence, our techniques cannot (directly) be applied. It thus is an interesting problem to study the parameterized complexity of PLP with respect to the vertex cover number.

References

1 Patrizio Angelini, Giordano Da Lozzo, Giuseppe Di Battista, Fabrizio Frati, Maurizio Patrignani, and Ignaz Rutter. Beyond level planarity. In Yifan Hu and Martin Nöllenburg, editors, *Graph Drawing and Network Visualization - 24th International Symposium, GD 2016, Athens, Greece, September 19-21, 2016, Revised Selected Papers*, volume 9801 of *Lecture Notes in Computer Science*, pages 482–495. Springer, 2016. doi:10.1007/978-3-319-50106-2_37.

2 Patrizio Angelini, Giordano Da Lozzo, Giuseppe Di Battista, Fabrizio Frati, and Vincenzo Roselli. The importance of being proper: (in clustered-level planarity and t-level planarity). *Theor. Comput. Sci.*, 571:1–9, 2015. doi:10.1016/j.tcs.2014.12.019.

3 Andrei Asinowski, Tillmann Miltzow, and Günter Rote. Quasi-parallel segments and characterization of unique bichromatic matchings. *Journal of Computational Geometry*, 6:185–219, 2015. doi:10.20382/jocg.v6i1a8.

4 Christian Bachmaier, Franz-Josef Brandenburg, and Michael Forster. Radial level planarity testing and embedding in linear time. *J. Graph Algorithms Appl.*, 9(1):53–97, 2005. URL: http://jgaa.info/accepted/2005/BachmaierBrandenburgForster2005.9.1.pdf, doi:10.7155/JGAA.00100.

5 Christian Bachmaier and Wolfgang Brunner. Linear time planarity testing and embedding of strongly connected cyclic level graphs. In Dan Halperin and Kurt Mehlhorn, editors, *Algorithms - ESA 2008, 16th Annual European Symposium, Karlsruhe, Germany, September 15-17, 2008. Proceedings*, volume 5193 of *Lecture Notes in Computer Science*, pages 136–147. Springer, 2008. doi:10.1007/978-3-540-87744-8_12.

6 Michael J. Bannister, William E. Devanny, Vida Dujmović, David Eppstein, and David R. Wood. Track layouts, layered path decompositions, and leveled planarity. *Algorithmica*, 81(4):1561–1583, 2019. doi:10.1007/S00453-018-0487-5.

7 Václav Blažej, Boris Klemz, Felix Klesen, Marie Diana Sieper, Alexander Wolff, and Johannes Zink. Constrained and ordered level planarity parameterized by the number of levels. In Wolfgang Mulzer and Jeff M. Phillips, editors, *40th International Symposium on Computational Geometry, SoCG 2024, June 11-14, 2024, Athens, Greece*, volume 293 of *LIPIcs*, pages 20:1–20:16. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2024. doi:10.4230/LIPICS.SOCG.2024.20.

8 Guido Brückner and Ignaz Rutter. Partial and constrained level planarity. In Philip N. Klein, editor, *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2017, Barcelona, Spain, Hotel Porta Fira, January 16-19*, pages 2000–2011. SIAM, 2017. doi:10.1137/1.9781611974782.130.

9 Guido Brückner and Ignaz Rutter. An spqr-tree-like embedding representation for level planarity. In Yixin Cao, Siu-Wing Cheng, and Minming Li, editors, *31st International Symposium on Algorithms and Computation, ISAAC 2020, December 14-18, 2020, Hong Kong, China (Virtual Conference)*, volume 181 of *LIPIcs*, pages 8:1–8:15. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2020. doi:10.4230/LIPIcs.ISAAC.2020.8.

10 Marek Cygan, Fedor V. Fomin, Lukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michal Pilipczuk, and Saket Saurabh. *Parameterized Algorithms*. Springer, 2015. doi:10.1007/978-3-319-21275-3.

11 Mark de Berg, Otfried Cheong, Marc J. van Kreveld, and Mark H. Overmars. *Computational geometry: algorithms and applications, 3rd Edition*. Springer, 2008. URL: https://www.worldcat.org/oclc/227584184.

12 N. G. de Bruijn. Problems 17 and 18. *Nieuw Archief voor Wiskunde*, 2:67, 1954. Answers in *Wiskundige Opgaven met de Oplossingen* 20:19–20, 1955.

13 Giuseppe Di Battista and Enrico Nardelli. Hierarchies and planarity theory. *IEEE Trans. Systems, Man, and Cybernetics*, 18(6):1035–1046, 1988. doi:10.1109/21.23105.

14 Vida Dujmović, Michael R. Fellows, Matthew Kitching, Giuseppe Liotta, Catherine McCartin, Naomi Nishimura, Prabhakar Ragde, Frances A. Rosamond, Sue Whitesides, and David R.

Wood. On the parameterized complexity of layered graph drawing. *Algorithmica*, 52(2):267–292, 2008. doi:10.1007/S00453-007-9151-1.

15 Michael Forster and Christian Bachmaier. Clustered level planarity. In Peter van Emde Boas, Jaroslav Pokorný, Mária Bieliková, and Julius Stuller, editors, *SOFSEM 2004: Theory and Practice of Computer Science, 30th Conference on Current Trends in Theory and Practice of Computer Science, Merin, Czech Republic, January 24-30, 2004*, volume 2932 of *Lecture Notes in Computer Science*, pages 218–228. Springer, 2004. doi:10.1007/978-3-540-24618-3_18.

16 Radoslav Fulek, Michael J Pelsmajer, Marcus Schaefer, and Daniel Štefankovič. Hanani–Tutte, monotone drawings, and level-planarity. In *Thirty Essays on Geometric Graph Theory*, pages 263–287. Springer, 2013. doi:10.1007/978-1-4614-0110-0_14.

17 Michael R Garey and David S Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman & Co., 1990.

18 Ashim Garg and Roberto Tamassia. On the computational complexity of upward and rectilinear planarity testing. *SIAM J. Comput.*, 31(2):601–625, 2001. doi:10.1137/S0097539794277123.

19 Leonidas J. Guibas and F. Francis. Yao. On translating a set of rectangles. In *Proc. 12th Annual ACM Symposium Theory of Computing (STOC 1980)*, pages 154–160, 1980. doi:10.1145/800141.804663.

20 Leonidas J. Guibas and F. Francis. Yao. On translating a set of rectangles. In F. P. Preparata, editor, *Computational Geometry*, volume 1 of *Advances in Computing Research*, pages 61–77. JAI Press, Greenwich, Conn., 1983.

21 Lenwood S. Heath and Sriram V. Pemmaraju. Recognizing leveled-planar dags in linear time. In Franz-Josef Brandenburg, editor, *Graph Drawing, Symposium on Graph Drawing, GD '95, Passau, Germany, September 20-22, 1995, Proceedings*, volume 1027 of *Lecture Notes in Computer Science*, pages 300–311. Springer, 1995. doi:10.1007/BFb0021813.

22 Lenwood S. Heath and Arnold L. Rosenberg. Laying out graphs using queues. *SIAM J. Comput.*, 21(5):927–958, 1992. doi:10.1137/0221055.

23 Seok-Hee Hong and Hiroshi Nagamochi. Convex drawings of hierarchical planar graphs and clustered planar graphs. *J. Discrete Algorithms*, 8(3):282–295, 2010. doi:10.1016/j.jda.2009.05.003.

24 Michael Jünger, Sebastian Leipert, and Petra Mutzel. Pitfalls of using PQ-trees in automatic graph drawing. In Giuseppe Di Battista, editor, *Graph Drawing, 5th International Symposium, GD '97, Rome, Italy, September 18-20, 1997, Proceedings*, volume 1353 of *Lecture Notes in Computer Science*, pages 193–204. Springer, 1997. doi:10.1007/3-540-63938-1_62.

25 Michael Jünger, Sebastian Leipert, and Petra Mutzel. Level planarity testing in linear time. In Sue Whitesides, editor, *Graph Drawing, 6th International Symposium, GD'98, Montréal, Canada, August 1998, Proceedings*, volume 1547 of *Lecture Notes in Computer Science*, pages 224–237. Springer, 1998. doi:10.1007/3-540-37623-2_17.

26 Boris Klemz. Convex drawings of hierarchical graphs in linear time, with applications to planar graph morphing. In Petra Mutzel, Rasmus Pagh, and Grzegorz Herman, editors, *Proc. 29th Ann. Europ. Symp. Algorithms (ESA'21)*, volume 204 of *LIPIcs*, pages 57:1–57:15. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2021. doi:10.4230/LIPIcs.ESA.2021.57.

27 Boris Klemz and Günter Rote. Ordered level planarity and its relationship to geodesic planarity, bi-monotonicity, and variations of level planarity. *ACM Trans. Algorithms*, 15(4):53:1–53:25, 2019. Conference version in Proc. GD'17. doi:10.1145/3359587.

28 Boris Klemz and Marie Diana Sieper. Constrained level planarity is FPT with respect to the vertex cover number. arXiv report, 2024. arXiv:2404.16723.

29 Ignaz Rutter. Personal communication, 2022.

30 Andreas Wotzlaw, Ewald Speckenmeyer, and Stefan Porschen. Generalized k-ary tanglegrams on level graphs: A satisfiability-based approach and its evaluation. *Discrete Applied Mathematics*, 160(16-17):2349–2363, 2012. doi:10.1016/j.dam.2012.05.028.