Master Thesis

Parameterized Complexity of Constrained and Partial Level Planarity

Marie Diana Sieper

Date of Submission: April 30, 2022
Advisors: Prof. Dr. Alexander Wolff
Prof. Dr. Ignaz Rutter (Universität Passau)

Julius-Maximilians-Universität Würzburg
Lehrstuhl für Informatik I
Algorithmen und Komplexität

Abstract

Level Planarity asks for a planar drawing of a graph G on horizontal lines called levels under the constraint that each vertex can only be drawn on a specific level, also called a level-planar drawing of G.

In the problem Partial Level Planarity (PLP) we are additionally given a level-planar partial drawing $\mathcal{H}$ of a subgraph H of G and want to know whether it is possible to find a level-planar drawing of G which coincides with $\mathcal{H}$ on H. Constrained Level Planarity (CLP) adds further restrictions via a set of constraints on the order of some of the vertices and asks whether it is possible to find a level-planar drawing of G such that all constraints are fulfilled. For all of these versions we can formulate a Radial counterpart, in which the vertices are not assigned onto levels but nested circles. While (Radial) Level Planarity can be solved in polynomial time [BR21], deciding (Radial) PLP and (Radial) CLP is NP hard [BR17].

In this thesis we investigate the parameterized complexity of Constrained and Partial Level Planarity for several parameters. Using that PLP is NP-hard even for seven levels [BR22a] and that every graph which can be drawn planar onto h levels has at most pathwidth h [DFK$^+$08], we show that it is impossible to find an FPT algorithm for any of the problems (Radial) PLP/CLP parameterized by the number of levels, pathwidth or treewidth. We further demonstrate that it is possible to solve CLP and (Radial) PLP for two levels in polynomial time and present an algorithm accomplishing this. In contrast to that, we prove that Radial CLP is NP-hard even for one level. We then give a reduction from 3-Partition to CLP with four levels, and show how a reduction from Brückner and Rutter [BR22a] from 3-Partition to PLP with seven levels can be adapted to Radial PLP with six levels, showing that these numbers are an upper bound for the number of levels needed for these problems to be NP-hard.

For the restricted case of a proper input instance, i.e., every edge connects vertices on adjacent levels, we show that we can solve CLP and PLP with an FPT algorithm parameterized by the vertex cover number of the input graph.

We then describe an expansion technique with which we can reduce PLP to PLP with exactly one vertex per level, and CLP to CLP with at most two vertices and one constraint per level, showing that there can neither exist an FPT algorithm for PLP or CLP parameterized by the maximal number of vertices per level, nor an FPT algorithm for (Proper) CLP parameterized by the maximal number of constraints per level. However, for the restricted problems Proper CLP and Proper PLP we present an FPT algorithm parameterized by the number of vertices per level.

Zusammenfassung

In dem Problem Level Planarity bekommen wir einen Graphen G übergeben, in dem jeder Knoten einem Level zugewiesen ist, und wir wollen wissen, ob wir diesen Graphen levelplanar zeichnen können, also so, dass die Zeichnung planar ist und jeder Knoten auf einer zu seinem Level gehörigen horizontalen Linie liegt.

In dem Problem Partial Level Planarity (PLP) bekommen wir zusätzlich noch eine levelplanare partielle Zeichnung $\mathcal{H}$ von einem Teilgraphen H von G gegeben, und wollen wissen, ob es möglich ist eine levelplanare Zeichnung von G zu finden, die auf H mit $\mathcal{H}$ übereinstimmt. In dem Problem Constrained Level Planarity (CLP) erhalten wir zusätzlich eine Menge an Beschränkungen, die eine partielle Ordnung für Knoten auf jeweils einem Level beschreibt, und wollen wissen, ob es eine levelplanare Zeichnung von G gibt, in der alle diese Beschränkungen erfüllt sind. Für beide Versionen von Level Planarity können wir eine entsprechende radiale Variante formulieren, in der die Knoten nicht horizontalen Leveln, sondern ineinander verschachtelten Kreisen zugewiesen werden. Während (Radial) Level Planarity in Polynomialzeit gelöst werden kann [BR21], ist es NP-schwer (Radial) PLP und (Radial) CLP zu entscheiden [BR17].

In dieser Arbeit untersuchen wir die parametrisierte Komplexität von Constrained und Partial Level Planarity für verschiedene Parameter. Indem wir ausnutzen, dass PLP mit sieben Leveln NP-schwer ist [BR22a], und dass jeder Graph, der planar auf h Leveln gezeichnet werden kann höchstens Pfadweite h hat [DFK+08], zeigen wir, dass es nicht möglich ist, einen FPT-Algorithmus für eines der Probleme (Radial) PLP/CLP parametrisiert nach der Anzahl der Level, Pfadweite oder Baumweite zu finden. Weiterhin zeigen wir, dass es es möglich ist, CLP und (Radial) PLP für zwei Level in Polynomialzeit zu lösen. Im Gegensatz dazu beweisen wir, dass Radial CLP bereits ab einem Level NP-schwer ist. Dann präsentieren wir eine Reduktion von 3-Partition auf CLP mit vier Leveln. Darüber hinaus beschreiben wir, wie eine Reduktion von Brückner und Rutter [BR22a] von 3-Partition nach PLP mit sieben Leveln auf Radial PLP mit sechs Leveln angepasst werden kann. Diese Werte sind also obere Schranken für die Anzahl an Leveln sind, die benötigt werden, damit die entsprechenden Probleme NP-schwer werden.

Für den eingeschränkten Fall, dass der Eingabegraph proper ist (also dass jede Kante Knoten verbindet, die auf benachbarten Leveln liegen), zeigen wir, dass CLP und PLP mit einem FPT-Algorithmus parametrisiert nach der Knotenüberdeckungszahl des Eingabegraphen gelöst werden kann.

Wir beschreiben anschließend ein Ausdehnungsverfahren, mithilfe dessen wir PLP zu PLP mit genau einem Knoten pro Level, und CLP zu CLP mit höchstens zwei Knoten pro Level und höchstens einer Beschränkung pro Level reduzieren können. Damit beweisen wir, dass weder ein FPT-Algorithmus für PLP oder CLP parametrisiert nach der maximalen Anzahl an Knoten pro Level, noch ein FPT-Algorithmus für (Proper) CLP parametrisiert nach der maximalen Anzahl an Beschränkungen pro Level existieren kann. Für den eingeschränkten Fall Proper CLP und Proper PLP präsentieren wir jedoch einen FPT Algorithmus parametrisiert nach der maximalen Anzahl an Knoten

pro Level.

Contents

1 Introduction

Visualizing hierarchical data is ubiquitous in many areas such as network visualization, marketing, business analytics and engineering. Often, this hierarchy is best represented in a top-down fashion to make the order of the data clear and understandable. It can be phrased as the problem of drawing so called level graphs. Since crossings in a drawing greatly impact clarity and readability of visualized data, it is desirable to devise crossing free drawings of this data. The question whether a given level graph can be drawn in a level-planar way arises naturally. Since specific data might contain additional structure besides the levels it is reasonable to require a drawing to fulfil a few additional properties. Such requirements can be represented in various forms. In this work we will consider two of them. The first one aims to expand a given partial drawing of a subgraph, the other one has its restrictions given in the form of a partial order on the set of vertices.

1.1 Related work

Level Planarity asks for a planar drawing of a graph G on horizontal lines called levels under the constraint that each vertex can only be drawn on a specific level, also called a level-planar drawing of G. Di Battista and Nardelli showed that the problem can be solved in polynomial time if G has only one source [DN88a]. Jünger et al. [JLM98] gave a linear-time recognition algorithm for the general case, which was refined to also compute an embedding within the same running time by Jünger and Leipert [JL99].

Radial Level Planarity augments Level Planarity in such a way that the vertices are not assigned to levels but to nested circles. Bachmaier et al. [BBF04] presented a linear-time testing and embedding algorithm for Radial Level Planarity. Brückner and Rutter [BR22b] described another approach for a linear time algorithm solving (Radial) Level Planarity, while pointing out some critical gaps in the preceding linear-time results and providing a detailed survey of previous work on the subject. They also developed a linear time algorithm for testing (Radial) Level Planarity with a fixed embedding [BR21].

Various versions of Level Planarity with additional restrictions have also been considered. Harrigan and Healy [HH08] gave an algorithm with quadratic running time for Level Planarity, in which constraints on the order of incident edges around the vertices could be given. Angelini et al. [ADD$^+$15] considered two restricted versions called t-Level Planarity and Clustered Level Planarity, and showed that both problems are NP-complete in the general case, but become solvable in polynomial time if the input is required to be proper (i.e., every edge connects vertices on adjacent levels). Brückner and Rutter [BR17] introduced Constrained Level Planarity, in which a partial order on the set of vertices is given, and Partial Level Planarity, in which the drawing of a subgraph is fixed. They showed that in the general case, both problems are NP-hard, but are solvable in polynomial time with only one source. Later, Brückner and Rutter [BR22a] offered another NP-hardness proof for both problems, in which it is shown that the problems are already NP-hard with only seven levels.

There also exists a variety of problems similar to Level Planarity, for which FPT algorithms have been found.

In Upward Planarity we are given an acyclic digraph G and ask for a planar drawing of G in which every edge is drawn strictly upward. Garg and Tamassia [GT01] showed that Upward Planarity is NP-hard via reduction of a variant of 3-SAT. Chan [Cha04] presented the first FPT algorithm for Upward Planarity. His algorithm has a running time of $O(t!8^t n(G)^3 + 2^{3\cdot 2^l} t^{3\cdot 2^l} t!8^t)$, where k is the number of triconnected components, and l the number of cut vertices. Healy and Lynch [HL06] gave two further FPT algorithms, one with a running time of $O(2^t \cdot t! \cdot n(G)^2)$, where t again denotes the number of triconnected components of the graph, and the other one with a running time of $O(n(G)^2 + k^4 \cdot (2k)!)$, where $k = |E| - |V|$.

In the problem h-Layered Graph Drawing we are given a graph G, but no predetermined level assignment. The problems asks for a level drawing of G on h levels. Heath and Rosenberg [HR92] showed that this problem is NP-hard. Dujmovic et al. [DFK+08] proved that h-Layered Graph Drawing is in FPT parameterized by h.

1.2 Contribution

We introduce the problems Radial Partial Level Planarity and Radial Constrained Level Planarity and set them in relation to PLP and CLP, see Chapter 2. We are not aware of former work on these problems. We show that the problems (Radial) PLP and (Radial) CLP are not in FPT parameterized by the number of levels, pathwidth or treewidth, see Section 3.1. In a refined analysis on the number of levels needed to achieve NP-hardness (see Section 3.2), we show that it is possible to solve CLP and (Radial) PLP for two levels in polynomial time and present an algorithm doing so. In contrast to that, we prove that Radial CLP is NP-hard even for one level. We give a reduction from 3-Partition to CLP with four levels, and show how a reduction from Brückner and Rutter [BR22a] from 3-Partition to PLP with seven levels can be adjusted to Radial PLP with six levels, showing that these numbers are upper bounds for the numbers of levels needed for these problems to be NP-hard.

For the restricted case of a proper input instance, we show that we can solve CLP and PLP with an FPT algorithm parameterized by the vertex cover number of the input graph, see Chapter 4.

We describe an expansion technique with which we can reduce PLP to PLP with exactly one vertex per level, and CLP to CLP with at most two vertices and one constraint per level, showing that there can neither exist an FPT algorithm for PLP or CLP parameterized by the maximal number of vertices per level, nor an FPT algorithm for (Proper) CLP parameterized by the maximum number of constraints per level. For the restricted problems Proper CLP and Proper PLP, however, we present an FPT algorithm parameterized by the number of vertices per level, see Chapter 5.

Finally, we conclude and outline questions for future work in Chapter 6.

2 Preliminaries

Before we can proceed to the main part of this work, we at first need some formal definitions.

Definition 1 (General terms and conventions)**.** *Let G be a graph (or, in most cases in this work, a digraph). We define $V(G)$ to be the set of vertices and $E(G)$ to be the set of edges of G. Similarly we define $n(G) = |V(G)|$ and $m(G) = |E(G)|$ to be the number of vertices and edges of G, respectively. As shorthand, we will use n, m if G is clear from context.*

For $k \in \mathbb{N}$ we define $[k] = \{1, 2 \ldots, k\} \subset \mathbb{N}$.

We can now state the problem this work focuses on.

Definition 2 (Level Planarity)**.** *Let G be a digraph together with a level assignment $\ell\colon V(G) \to [h]$ such that $\ell(u) \leq \ell(v)$ for each edge $(u, v) \in E(G)$. We say that G is an* h-level graph*. Let further $V_j(G) = \ell(j)^{-1}$ be the vertices of level j.*

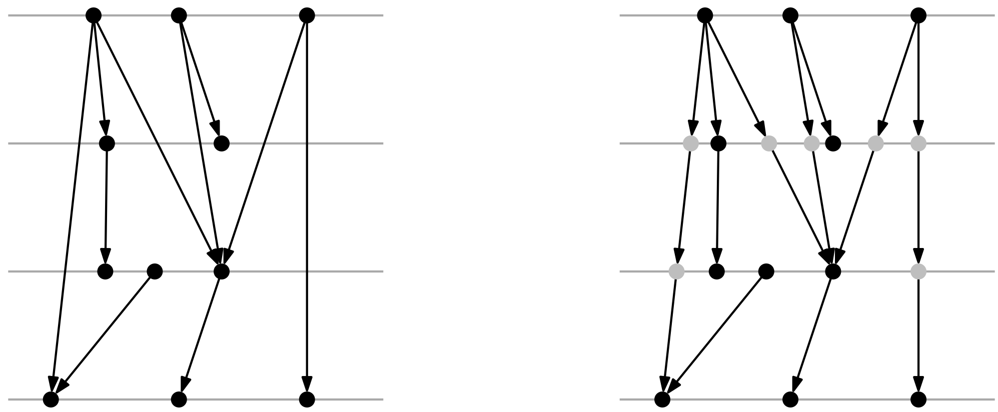
A drawing of G is called a level drawing *if for every level $j \in [h]$ all vertices $v \in V_j(G)$ are drawn on the same horizontal line, the line of level j lies below the line of level j' if and only if $j > j'$, and every edge $e = (u, v)$ is drawn strictly downward ($\ell(u) \leq y(e) \leq \ell(v)$). We say a level drawing of G is a* level-planar drawing *if it is crossing-free, and G is* level-planar *if it admits a level-planar drawing.*

We say that a level graph G is proper *if, for every edge $e = (u, v) \in E(G)$, the incident vertices lie on consecutive levels (that is, if $\ell(v) = \ell(u) + 1$).*

As we can simply subdivide each edge spanning several levels with more vertices without changing planar drawability, we can in most cases assume that our input instance is proper. Figure 2.1 shows the drawing of a level graph before (Figure 2.1a) and after (Figure 2.1b) subdividing all edges spanning over more than one level, and the graph shown in Figure 2.1b is proper. In a level graph G every edge spans at most over $n - 2$ levels, and we can assume that our input instance is planar and thus has at most $O(n(G))$ edges. Hence, this properization step increases the size of G at most quadratically.

Since the direction of the edges is clear from context, we will from now on omit them in drawings. If we want to implement some restrictions on the drawing of a level graph, it is natural to define a partial order on the levels.

Definition 3 (Constrained Level Planarity (CLP))**.** *Let G be an h-level graph. Let further some constraints on the order of vertices in form of a partial order $\prec_j$ for each level $j \in [h]$ be given. In the problem Constrained Level Planarity (CLP) we ask for a level-planar drawing $\mathcal{G}$ of G compatible with $\prec_j$. In order for a level-planar drawing $\mathcal{G}$ to be* compatible *with these constraints, the vertex order of each level j must be a linear extension of $\prec_j$.*



(a) A drawing of a level graph which is not proper due to some edges spanning over more than one level.

(b) A drawing of the same graph, in which every edge spanning more than one level got subdivided.

Fig. 2.1: A visualization of how to properize a graph.

In the following we assume that the partial orders are given in the form of a set $C = \bigcup_{j\in[h]} C_j$, where C_j contains the constraints of level j. A constraint has the form $v \prec v'$, $v, v' \in V$ meaning that in a drawing of G, v must lie before v'. We can observe that not every constraint must be explicitly contained in C. For example $v_1 \prec v_2$ and $v_2 \prec v_3$, $v_1, v_2, v_3 \in V$ also enforce $v_1 \prec v_3$, even if this constraint is not contained in C. We say C is *closed*, if for all $v_1 \prec v_2$ and $v_2 \prec v_3$, $v_1, v_2, v_3 \in V$, $v_1 \prec v_3$ is explicitly contained in C. As we can include all these transitive constraints in quadratic time we can assume in most cases that C is closed.

Another way to implement restrictions on a drawing of a level graph is to fix a partial drawing in advance.

Definition 4 (Partial Level Planarity (PLP))**.** *Let G be an h-level graph and let $\mathcal{H}$ be a level-planar drawing of a subgraph $H \subseteq G$ (also called a* partial drawing *of G). In the problem Partial Level Planarity (PLP) we ask for a level-planar drawing of G compatible with $\mathcal{H}$. In order for a level-planar drawing $\mathcal{G}$ of G to be compatible with $\mathcal{H}$ it must coincide with $\mathcal{H}$ on H.*

If we consider a level-planar drawing $\mathcal{G}$ of a graph G, then every other straight-line level drawing of G in which the vertices lie in the same order as in $\mathcal{G}$ is also planar [DN88b]. We are therefore only interested in the order of vertices and not in their exact vertical placement. This means that proper PLP reduces to a special case of CLP, because all necessary information provided by the partial drawing can be expressed in terms of a total order on the vertices in this partial drawing [BR17]. The converse does not hold, since a partial drawing $\mathcal{H}$ always defines a total order on H. This is because in CLP we can require for a vertex $v \in V(G)$ to lie before or after some vertices $v_1, v_2 \in V(G)$, and leave the relation between v_1, v_2 open, which is not possible in PLP.

While Level Planarity can be solved in polynomial and even linear time [BR22b], CLP and PLP remain NP-hard [BR17].

A problem closely related to Level Planarity is Radial Level Planarity, which – while not the main focus of this work – will be considered and compared to Level Planarity in Chapter 3. We first need to adapt the concept of a (partial) order to circles.

Definition 5 (Partial cyclic order, [GM77])**.** *Let M be a finite set. A* partial cyclic order *on M is a set Λ of ordered triples (x, y, z) out of M^3, x, y, z pairwise unequal, such that*

1. *with a triplet (x, y, z) all cyclic equivalents are also included:* $(x, y, z) \in \Lambda \Rightarrow (y, z, x), (z, x, y) \in \Lambda$
2. *Λ is antisymmetric:* $(x, y, z) \in \Lambda \Rightarrow (z, y, x) \notin \Lambda$
3. *Λ is transitive:* $(x, y, z), (x, z, w) \in \Lambda \Rightarrow (x, y, w) \in \Lambda$

We understand $(x, y, z) \in \Lambda$ such that if we want to place M on a circle, if we start from x and traverse M counter-clockwise, we find y before z.

We say Λ over M is saturated*, if for every triplet $(x, y, z) \in M^3$, x, y, z pairwise unequal, either $(x, y, z) \in \Lambda$ or $(z, y, x) \in \Lambda$ holds. In other words for every triplet a cyclic order is given. A saturated cyclic partial order on M corresponds directly to a cyclic order on M. Let further Λ' be another partial cyclic order on M. We say Λ is extendable to Λ' if $\Lambda \subseteq \Lambda'$.*

In opposition to the corresponding problem on a line, there are partial cyclic orders which cannot be extended to a saturated cyclic order [Meg76]. In fact, it is even NP-hard to decide for a given partial cyclic order Λ whether it is extendable to a saturated cyclic order [GM77].

We can now proceed to the definition of Radial Level Planarity.

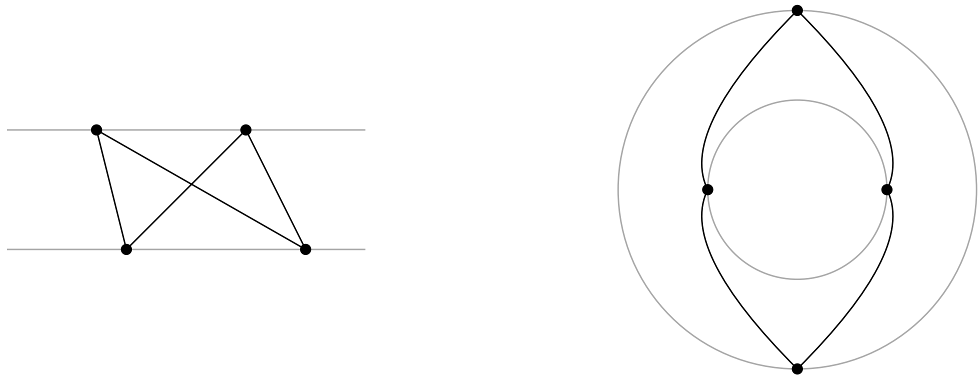
Definition 6 (Radial Level Planarity)**.** *Let G be an h-level graph together with a radial level assignment $\ell \colon V(G) \to [h]$, such that $\ell(u) \leq \ell(v)$ for all edges $(u, v) \in E$. We say that $V_j = \ell(j)^{-1}$ are the vertices of the radial level j.*

A radial level drawing *of G is a drawing in which every vertex $v \in V(G)$ is drawn on a circle with radius $\ell(v)$ around some origin, and every edge $e = (u, v)$ is drawn as a continuous curve between u, v such that every circle with radius r, $\ell(u) \leq r \leq \ell(v)$ and e cross at exactly one point. We say a radial level drawing of G is a* radial level-planar drawing *if it is crossing-free, and G is* radial level-planar *if it admits a radial level-planar drawing.*

Just like with Level Planarity, an h-level graph G is *proper* if every edge e connects only vertices on consecutive radial levels, and we can properize every given instance by subdividing edges.

We can now define some restricted versions of Radial Level Planarity.

Definition 7 (Radial Constrained Level Planarity (Radial CLP))**.** *Let G be an h-level graph. Let further some constraints on the order of vertices in form of a partial*



(a) An example of a graph that is radial level-planar, but not level-planar.

(b) A level-planar drawing of $K_{2,2}$ demonstrating the level function.

Fig. 2.2: A non-planar radial level-planar drawing of $K_{2,2}$.

cyclic order C_j for every radial level j be given and define the set of cyclic constraints $C = \bigcup_{j \in [h]} C_j$ as their union.

In the problem Constrained Radial Level Planarity (radial CLP) *we ask for a radial level-planar drawing $\mathcal{G}$ of G compatible with C. In order for a radial level-planar drawing $\mathcal{G}$ to be* compatible *with C the circular order of vertices on level j in $\mathcal{G}$ must be an extension of C_j for every level j.*

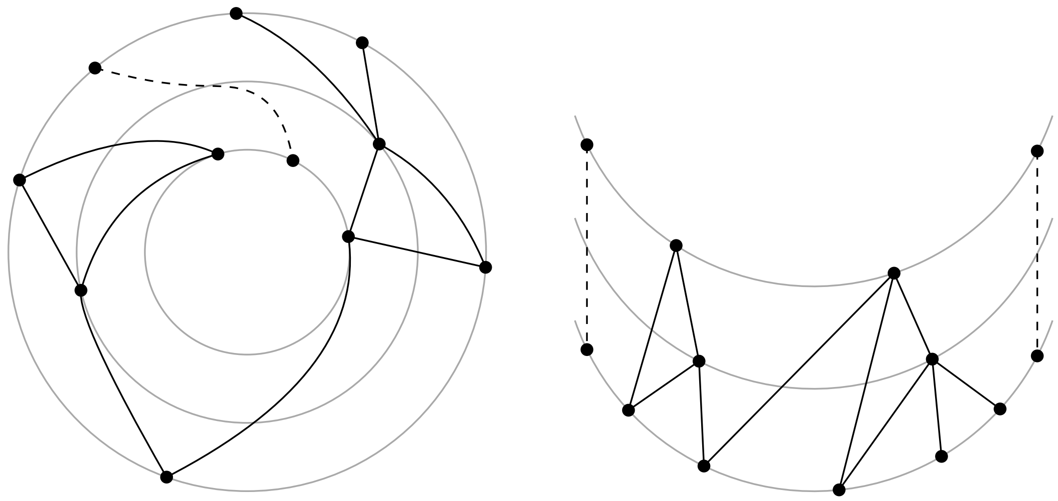
Definition 8 (Radial Partial Level Planarity (Radial PLP))**.** *Let G be an h-level graph and let $\mathcal{H}$ be a partial radial level-planar drawing of a subgraph $H \subseteq G$.*

In the problem Partial Radial Level Planarity (radial PLP) *we ask for a radial level-planar drawing $\mathcal{G}$ of G compatible with $\mathcal{H}$. In order for a level-planar drawing $\mathcal{G}$ of G to be compatible with $\mathcal{H}$ it must must coincide with $\mathcal{H}$ on H.*

We can show that radial PLP reduces to (proper) radial CLP analogue to how it can be shown that PLP reduces to (proper) CLP. To differentiate between Radial Level Planarity and Level Planarity as defined in Definition 2, we will sometimes refer to Level Planarity as Plain Level Planarity.

It is quite obvious that every level graph which is level-planar is also radial level-planar, but the other way around is not true. There exist level graphs which are radial level-planar, but not level-planar. An example for such a graph is the $K_{2,2}$ on two levels as shown in Figure 2.2a. We can see that in every level drawing there exists an edge from the upper left to the lower right vertex and an edge from the upper right to the lower left, so no crossing-free and thus no planar level drawing exists. But Figure 2.2b gives a radial level planar drawing of the $K_{2,2}$. Testing radial level planarity can be done in linear time [BR22b].

Now let G be an h-level graph, and G' be the graph obtained from G by adding one vertex v_1 on level 1, one vertex v_h on level h and an edge (v_1, v_h) (or, if we want a proper instance, a subdivided edge) between them. Then there exists a level-planar drawing of G if and only if there exists a radial level-planar drawing of G', because we can treat the



(a) A radial level-planar drawing of a graph G together with an edge e from the first to the last layer.

(b) A level-planar drawing of G. The circles have been broken up and stretched out.

Fig. 2.3: A visualization of how a level-planar drawing of a graph can be obtained from a radial level-planar drawing of this graph together with an edge (dashed).

edge (v_1, v_h) as a line at which we can split our radial drawing as shown in Figure 2.3. Conversely, it it is easy to see that G stays level-planar if we add the edge and every level-planar graph is also radial level-planar.

Following this thought, we can show that every version of Level Planarity reduces to its respective version of Radial Level Planarity.

Lemma 9. *Let G be an h-level graph together with a partial drawing $\mathcal{H}$ of a subgraph H, and let $\mathcal{H}'$ be a drawing of $H' = H + e$ (where e is constructed as described above) which we obtain from $\mathcal{H}$ by drawing e to the left (or right) of it. Then G has a level-planar drawing respecting $\mathcal{H}$ if and only if G' has a radial level-planar drawing respecting $\mathcal{H}'$.*

Proof. Adding e to the left (or right) of $\mathcal{H}$ guarantees that by obtaining a plain level drawing from the radial level drawing we do not cut through $\mathcal{H}$. The rest follows directly from the construction described above. □

Lemma 10. *Let G be a h-level graph together with a closed set of constraints $C = \bigcup_{j\in[h]} C_j$. Let further G' be the graph obtained from G by adding the vertices v_1, v_h, the edge (v_1, v_h) and then subdividing (v_1, v_h) with a vertex v_j for every inner level j. Generate a partial cyclic order C'_j for every level j by adding a constraint (v_j, x, y) and all its cyclic equivalents to C'_j for every constraint $x \prec y$ in C_j. This way, all the C'_j are partial cyclic orders. Let $C' = \bigcup_{j\in[h]} C'_j$ be the set of all these partial cyclic orders.*

Then G has a level-planar drawing respecting C if and only if G' has a radial level-planar drawing respecting C'.

Proof. Every C'_j is a partial cyclic order because transitivity follows from the fact that

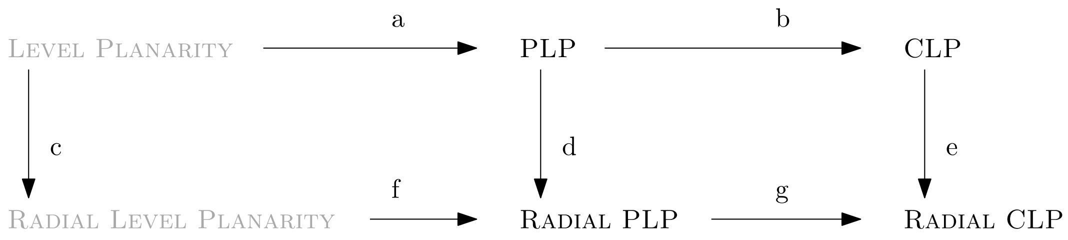


Fig. 2.4: An overview of the introduced problems and their relation to each other. An arrow leads from problem A to problem B if we can directly reduce A to B. Gray problems can be solved in linear time, the black ones are NP-hard.

C is closed and antisymmetry follows from the fact that C as a partial order is antisymmetric. The remaining claim follows directly from the construction described above. □

Figure 2.4 gives the reducibility correspondences discussed in this chapter. The reduction follows from:

a: Trivial.
b: [BR17].
c: Trivial.
d: Lemma 9.
e: Lemma 10.
f: Trivial.
g: Analogue to [BR17].

In particular, this means that Radial CCP and Radial PLP are NP-hard.

3 Number of Levels, Pathwidth and Treewidth

One of the most intrinsic parameters to Level Planarity is the number of levels h. Surprisingly, besides turning out to not be solvable by an PFT algorithm, this parameter also nearly immediately leads to a similar result for the more general parameters pathwidth and treewidth.

3.1 Parametrization by Number of Levels, Pathwidth and Treewidth

We first outline an NP-hardness proof for PLP by Brückner and Rutter [BR22a]. They reduce from 3-Partition, which is NP-hard.

Definition 11 (3-Partition). *Let $m \in \mathbb{N}$ be a positive integer, let A be a multiset with $n = 3m$ positive integers $a_1, \dots, a_{3m}$, and let $B \in \mathbb{Z}^+$ be a bound, such that both $B/4 < a < B/2$ for each $a \in A$ and $\sum_{a \in A} a = m \cdot B$ hold.*

3-Partition then asks whether A can be partitioned into m disjoint sets $A_1, A_2, \dots, A_m$, such that for every $j \in [m]$ the equation $\sum_{a \in A_j} a = B$ holds.

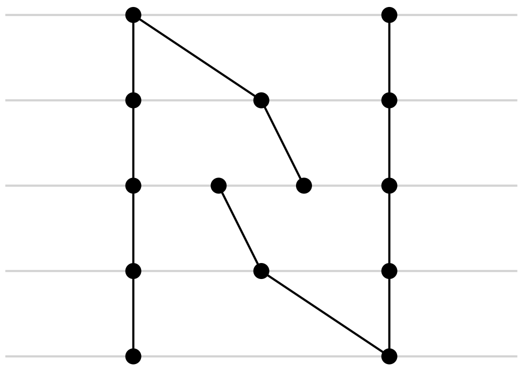
We refer to the sets $A_1, A_2, \dots, A_m$ as buckets*.*

Note that since we required $B/4 < a < B/2$ for each $a \in A$, in a valid solution every bucket A_j, $j \in [m]$ contains exactly three elements.

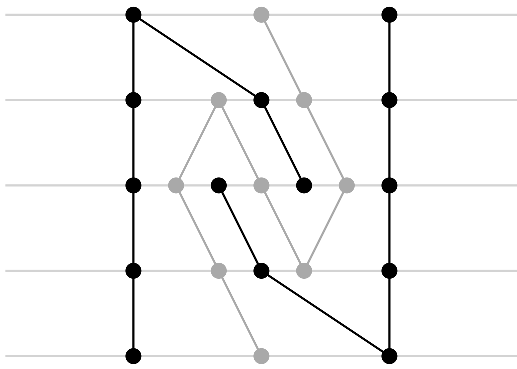
3-Partition is strongly NP-complete, meaning that it stays NP-complete even if we require B to be at most a polynomial of n [GJ75].

The reduction itself is built around a socket gadget as shown in Figure 3.1. The whole asset is fixed in a partial drawing as shown in Figure 3.1a. The key observation is that, for a path (also referred to as *pin*) as shown in Figure 3.1b there is only one possible way to traverse through the socket, and every socket can contain at most one pin.

The reduction is visualized in Figure 3.2. Let (m, A, B) be an instance of 3-partition. Our goal is to subdivide A into m buckets. For each $i \in [m]$ we glue together B/m socket gadgets and add an additional vertex one level above, which we connect to the left- and rightmost bucket line. We then combine m buckets together so that we have used exactly B sockets overall. To represent A, for every element $a \in A$ we combine a pins into a plug p_a. To make sure that all pins out of a plug p_a have to be drawn into one bucket, we combine them at level 6. One level above, we draw three additional vertices r_1, t, r_2. Vertex t gets connected towards the leftmost and the rightmost bucket separator, and additionally to all plugs. We also connect r_1, r_2 to the leftmost and the rightmost bucket



(a) A socket with its given drawing.



(b) A socket (black) used by a pin (grey).

Fig. 3.1: The socket gadget used and unused.

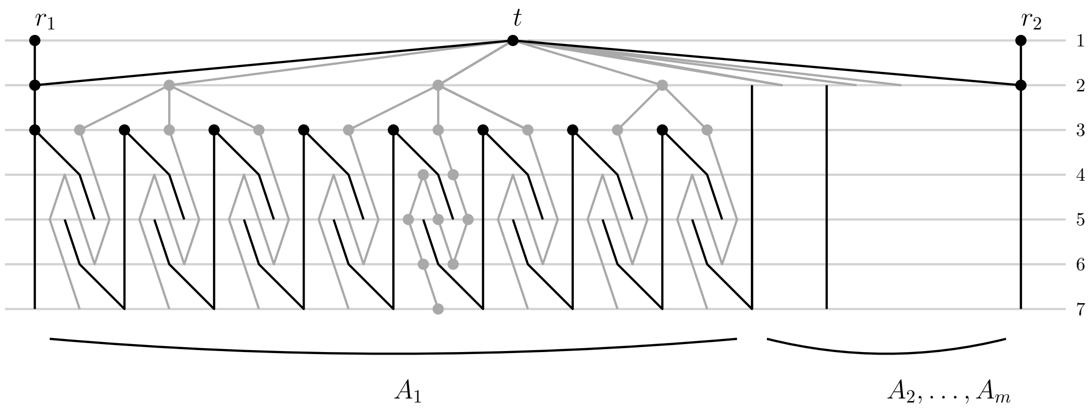


Fig. 3.2: [BR22a]. A visualization of the graph constructed during reduction from 3-Partition to PLP. The black part of the graph is fixed in a partial drawing as shown in this figure. Grey vertices can be moved. Note that this represents a proper graph, in which most vertices have ben omitted in the drawing for clarity.

separator respectively, and demand with the partial drawing the total order $r_1 < r_2 < r_3$ on them.

We have fixed the drawing of everything except the plugs with the partial drawing as described and demonstrated in Figure 3.2 and leave the plugs free, enabling them to move freely between the buckets but ensuring at the same time that no plug is outside of the bucket structure (this is guaranteed by r_1, r_2) and that no plug can stretch over several buckets because of the bucket separators.

We can now observe that with every level drawing of this construction compatible with the partial drawing every plug p_a uses exactly a sockets out of the bucket it sits in, and since $B/4 < a < B/2$ every bucket contains exactly three plugs. So in order to find a solution for (m, A, B) we can define each subset A_j as the elements a_1, a_2, a_3, such that bucket b_j contains the plugs $p_{a_1}, p_{a_2}, p_{a_3}$.

The other way around, if there exists a solution $A_1, A_2, \ldots, A_m$ for (m, A, B), we can draw the constructed graph by for every $a \in A_j$ placing the plug p_a into the bucket b_j.

Since 3-Partition is strongly NP-complete, this gives us a reduction to PLP. What makes it special is that the constructed level graphs all have a constant (seven to be precise) number of levels. This immediately results in the following theorem.

Theorem 12. *Consider the problem PLP and let h denote the number of levels in an input instance. Then, under the assumption that $P \neq NP$, there do not exist any functions $f, g\colon \mathbb{N} \to \mathbb{N}$, such that there exists an algorithm A solving PLP in $O(f(h) \cdot n^{g(h)})$ time.*

In particular there exists no FPT-algorithm solving PLP parameterized by the number of levels.

Proof. Assume there exist such functions f, g. Then there exists an algorithm A' solving 7-level PLP in $O(c_1 \cdot n^{c_2})$ time, c_1, c_2 being constants. Since there exists the method found by [BR22a] to reduce every instance of 3-Partition with size n to a 7-level PLP with polynomial size $poly(n)$, we can solve 3-Partition in $O(poly(n)^{c_2})$ and thus polynomial time, a contradiction if $P \neq NP$.

Since every FPT algorithm solving PLP parameterized by the number of levels would provide some functions f, g with these properties such an algorithm cannot exist. □

We will now extend this result to the parameters pathwidth and treewidth.

Theorem 13. *Consider the problem PLP, and let w denote the pathwidth or treewidth of an input instance. Then, under the assumption that $P \neq NP$, there do not exist any functions $f, g\colon \mathbb{N} \to \mathbb{N}$, such that there exists an algorithm A solving PLP in $O(f(w) \cdot n^{g(w)})$ time.*

In particular there exists no FPT algorithm solving PLP parameterized by pathwidth or treewidth.

Proof. Consider an instance of 3-partition and let G be a level graph obtained from this instance by the reduction onto PLP as described above. Note that the proof that the reduction from 3-Partition to PLP works only guarantees that G is level-planar

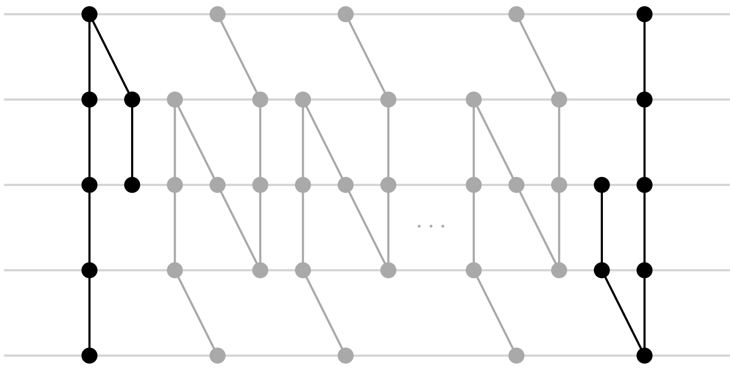


Fig. 3.3: A socket where the partial drawing got disregarded. It can now contain an arbitrary number of pins.

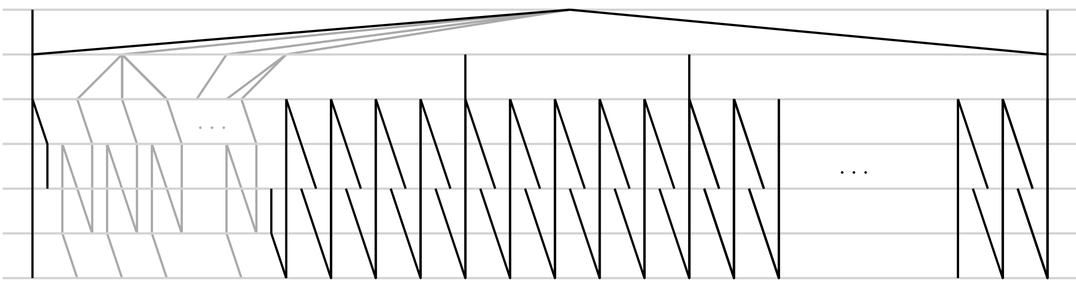


Fig. 3.4: A level-planar drawing of the 3-Partition reduction graph, which is not compatible with the constructed partial drawing. Every constructed graph can be drawn this way regardles of the question whether the original instance of 3-Partition has a solution.

if the starting instance of 3-Partition is solvable. Thus we will show at first that if omitting the partial drawing, every level graph constructed during the reduction from 3-Partition to PLP is level-planar.

Figure 3.3 shows a drawing of a socket where the given partial drawing got disregarded. We can see that we are now able to lead an arbitrary number of pins through the socket. We thus obtain a level-planar drawing of G if we leave most of the partial drawing intact, but lead all pins through the first socket as shown in Figure 3.4. Since every level-planar graph with h levels has at most pathwidth (and thus also at most treewidth) h [DFK$^+$08], we know that we can reduce 3-Partition onto PLP with constant pathwidth and thus treewidth.

Assume now that there exist such functions f, g. Then there exists an algorithm A' solving PLP with constant w in $O(c_1 \cdot n^{c_2})$ time, c_1, c_2 being constants. With this, we can solve 3-Partition3 -Partition in $O(poly(n)^{c_2})$ and thus polynomial time, a contradiction if $P \neq NP$.

Since every FPT-algorithm solving PLP parameterized by the number of levels would provide some functions f, g with these properties such an algorithm cannot exist. ☐

We are now able to generalize these results.

Theorem 14. *Let K be a problem out of PLP, CLP, Radial PLP, Radial CLP and let k denote the number of levels/circles, the pathwidth or the treewidth of an input instance. Then, under the assumption that $P \neq NP$, there do not exist functions $f, g : \mathbb{N} \to \mathbb{N}$, such that there exists an algorithm A solving K in $O(f(k) \cdot n^{g(k)})$ time.*

In particular there exists no FPT-algorithm solving K parameterized by k..

Proof. As this at most squares the size of the input instance, we assume that all of them proper. We thus know that we are able to express PLP as a special case of CLP, Radial CLP, Radial PLP. Therefore, if we had such f, g and an algorithm A solving K in $O(f(k) \cdot n^{g(k)})$ time, this would also solve every instance of PLP in the same time, a contradiction to Theorem 12 or Theorem 13. ☐

3.2 A Refined Analysis of the Number of Levels

Theorem 14 motivates the question whether there exists a number of levels for which we can solve CLP or PLP (and also Radial CLP and Radial PCP) in polynomial time, and if there are at which point the change to NP-hardness is.

Since CLP with only one-level is equal to the problem of extending a partial to a total order, this special case can be solved in quadratic time, and this also follows from lemma 15 shown below since a one-level graph has no edges. It turns out that also the case with two levels is solvable in polynomial time.

To show that we will first prove a lemma which will be useful in several topics.

Lemma 15. *Let G be an h-level graph together with a level assignment $\ell \colon V \to [h]$, let $V' \subseteq V(G)$ be a set of vertices such that $\deg(v) = 0$ for all $v \in V'$, and let $C = \bigcup_{j \in [h]} C_j$ a closed set of constraints. Let further $\mathcal{G}'$ be a level-planar drawing respecting C of the*

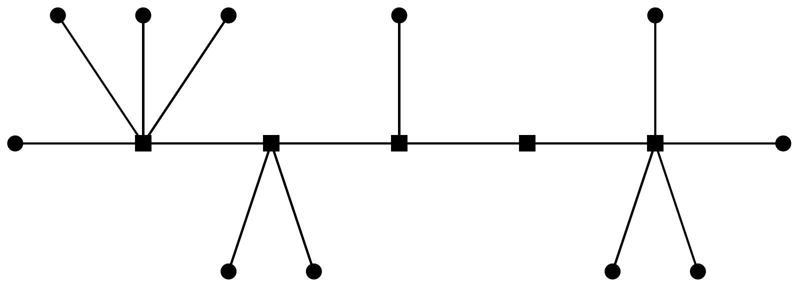


Fig. 3.5: A caterpillar graph. The underlying path p is represented by boxes, the degree 1 set V' is represented by points. Notice that the two vertices on the respective sides of p are not contained in p

subgraph $G' \subseteq G$ containing of all of G except the vertices in V'. Then there exists a level-planar drawing $\mathcal{G}$ of G respecting C, such that $\mathcal{G}$ reduced to G' equals $\mathcal{G}'$.

We can further obtain $\mathcal{G}$ from $\mathcal{G}'$ in $O(n(G)^2)$ time.

Proof. Let $v \in V'$ be an arbitrary vertex not drawn in G' with $\ell(v) = j$. If no vertex v' in G' with $\ell(v') = j$ exists such that there is constraint $(v \prec v') \in C$ we can draw v at the right of level j. Similarly if no vertex v^* in G' with $\ell(v') = j$ exists such that there is a constraint $(v^* \prec v) \in C$ we can draw v at the left of level j. Now assume that there exist vertices $v', v^* \in G$ such that $(v^* \prec v) \in C$ and $(v \prec v') \in C$. We can assume that v' to be leftmost and v^* to be the rightmost vertex on level j with this property. Because C is closed we know that $(v^* \prec v)$ is a constraint in contained explicitly in C, and therefore v^* must be left of v'. We can therefore draw v somewhere in between v^*, v'. Since those two were the leftmost respectively rightmost vertices with these properties, the insertion of v into $\mathcal{G}$ does not contradict any other constraint out of C.

This step can be performed in linear time, thus iterating it yields a level-planar drawing $\mathcal{G}$ of G which can be computed in quadratic time. □

We now take a closer look at the class of graphs which can be drawn level-planar between two levels.

Definition 16 (Caterpillar, [HS73]). *Let G be a graph with at least $n(G) \geq 3$ vertices and let $V' = \{v \in V(G) \mid \deg(v) = 1\} \subseteq V(G)$ be the set of leaves in G with degree 1. Then we call G a* caterpillar *if G becomes a path p through removal of V'.*

Figure 3.5 shows a caterpillar graph and its underlying path p. We will now see that caterpillar graphs are precisely those connected graphs we can draw on two levels.

Lemma 17. *Let G be a connected 2-level graph with $n(G) \geq 3$ together with a level assignment $\ell \colon V(G) \to [2]$. Then G is level-planar if and only if G is a caterpillar and $\ell v \neq \ell u$ for every edge (u, v) in G.*

In particular, every 2-level graph which is level-planar consists of caterpillars, singular vertices and singular edges (a component consisting out of two vertices and an edge between them).

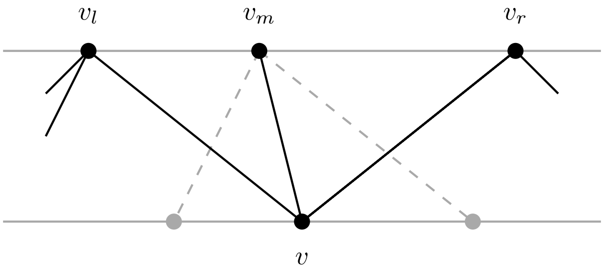


Fig. 3.6: A vertex v together with its set of neighbours. We can see that only the leftmost neighbour v_l and the rightmost vertex v_r can have neighbours besides v, since every further neighbour of a middle neighbour v_m would generate a crossing.

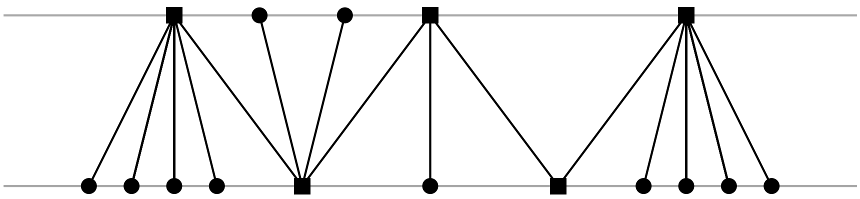


Fig. 3.7: The caterpillar out of Figure 3.5 drawn on two levels.

Proof. Let G be a connected 2-level graph with $n(G) \geq 3$ together with a level assignment $\ell\colon V(G) \to [2]$. Note that the condition $\ell v \neq \ell u$ follows directly from the fact that G is a level-graph.

If G is a caterpillar we can draw the underlying path p of G as zigzag line between the levels, and afterwards we can insert all leaves of a path vertex $v \in V$ in the interval between its (at most two) neighbours on the path. Now let G be level-planar, $\mathcal{G}$ be a level-planar drawing of G, and let v in G be a vertex with $\deg(v) \geq 2$. We assume that without loss of generality $\ell(v) = 1$. Let v_l be the leftmost and v_r be rightmost neighbour of v in $\mathcal{G}$. We can observe that only v_l, v_r can have another adjacent vertex besides v, because for every other neighbour v_m of v each edge (v_m, x) would necessarily cross (v_l, v) or (v_r, v), as can be seen in Figure 3.6. Thus v has at most two neighbours with a degree ≥ 2. We can also see that G cannot contain a cycle, since every cycle in G would necessarily create a crossing in a $2 - \mathit{level}$ drawing. Thus G is a tree in which the subset of non leaves forms paths. We cannot have more than one path since G is connected, thus the subset of non leaves forms one path and G is a caterpillar.

Therefore every graph which can be drawn level-planar on two levels is build up out of caterpillars, singular vertices and singular edges. □

Figure 3.7 shows a 2-level drawing of the caterpillar shown in Figure 3.5. We can observe that for the drawing of the underlying path p we have essentially two options, because we can draw p from left to right or the other way around. We will now see that we will be able to consider all components of a 2-level graph independently.

Lemma 18. *Let G_1 be two 2-level caterpillar graph and let G_2 be a 2-level caterpillar graph or a component consisting out of single edge. Let further $\mathcal{G}$ be a level-planar* *drawing of G_1 and G_2. Then on both level 1 and level 2 the vertices out of G_1 are drawn successively, and they are drawn before the vertices out of G_2 on level 1 if and only if they are drawn befor the vertices out of G_2 on level 2.*

Further, in every level-planar drawing $\mathcal{G}$ of a 2-level graph G without single vertices the following condition holds: the vertices of every component are drawn successive on each level and the order of these components is the same on both levels.

In particular, every partial order $\prec$ in a level-planar 2-level graph G induces $\prec_v$ induces a partial order $\prec'$ on the non-singular components out of G.

Proof. Let $\mathcal{G}$ be a drawing of G_1, G_2 and let v_1, v_1' in G_1 and v_2 in G_2 be some vertices such that they occur from left to right in the order v_1, v_2, v_1' on level 1 or 2. Since G_2 has at least 2 vertices v_2 has to have an incident edge e. Since v_1 and v_2 are connected through a path in G_1 e has to cross this path, and $\mathcal{G}$ cannot be level-planar. So the vertices out of G_1, G_2 must be drawn succesively on both levels, and the vertices are drawn before the vertices out of G_2 on level 1 if and only if they are drawn before the vertices out of G_2 on level 2 because both components have at least one edge and they would cross otherwise. This statement holds if we also allow for G_1 to be a single edge.

Iterating this proof gives us that in every level-planar drawing $\mathcal{G}$ of a 2-level graph G without single vertices the vertices of every component are drawn successive on each level and the order of these components is the same on both levels.

This means that the non-singular components can be ordered from left to right. Let now $G_1, \ldots, G_l$ denote the non-singular components of G. Then for every constraint of the form $v_x \prec v_y$, v_x a vertex in G_x and v_y a vertex in G_y we know that the component G_x must be drawn before G_y. Let $\prec_1, \prec_2$ be partial orders on level 1 and level 2. If G is level-planar $\prec_1$ and $\prec_2$ do not stand in conflict with each other and therefore induce a partial order $\prec'$on the components $G_1, \ldots, G_l$. □

Lemma 18 assures us that we can test level-planarity of restricted 2-level-graphs by concentrating on the components. We will now see that these can be tested in polynomial time.

Lemma 19. *Let G be a connected 2-level graph with at least 2 vertices together with a closed set of constraints $C = C_1 \cup C_2$. Then we can test whether there exists a level-planar drawing $\mathcal{G}$ of G respecting C and compute such a drawing in $O(n(G)^2)$ time.*

Proof. We know from Lemma 17 that for G in order to be level-planar it must hold that G is a caterpillar or a single edge. Testing whether G is a caterpillar is straightforward and can be done in $O(n(G))$ time. If G consists of a single edge, it is level-planar and can be drawn in a unique way. Since there is only one vertex on each level the constraints $C_1 = C_2 = \varnothing$ need to be empty and thus the drawing is consistent with C.

Now let G be a caterpillar and p be its path containing all vertices with degree at least 2. We have already seen that there are exactly two options to draw p level-planar: as a zig-zag line from left to right ar the other way around. Since these are only two options, we can test for both if they can be extended to a level-planar drawing of G.

Now let v be a vertex out of p and let L denotes its set of neighbours with degree 1. As we have seen in the proof of Lemma 17 all vertices of L_v must be drawn successively between the path neighbours of v or – if v is an endpoint of p – at one of the sides of G. Therefore we can choose any order of L_v compatible with C (such an order always exists since partial orders do not contain contradictions) and draw L_v at its respective place. This can be done in $O(n^2)$ time because $|C| \in O(n(G)^2)$. We can now test whether at least one of these two drawings (one for each direction of p) is level-planar and consistent with C by testing every constraint $c \in C$ in $O(n(G)^2)$ time. If G is level-planar, one of them needs to be and we can return it. Otherwise, G is not level-planar and we can stop. □

We can now formulate an algorithm solving 2-level CLP.

Algorithm 1: 2-level graph G, level function ℓ, closed constraints C)

$K_1, \ldots, K_l \leftarrow$ non-singular components of G
$\mathcal{G} \leftarrow$ empty drawing
if $k \neq 0$ **then**
    $C_K \leftarrow$ Set of constraints on $\{K_1, \ldots, K_l\}$ induced by C
    if C_k inconsistent **then**
        return no
    for $j = 1$ to l **do**
        if K_j not 2-level planar **then**
            return no
        $\mathcal{K}_j \leftarrow$ level-planar drawing of K_j
    Add all $\mathcal{K}_j$ to $\mathcal{G}$ in an order consistent with C_K
Add all single vertices to $\mathcal{G}$
return $\mathcal{G}$

Theorem 20. *Algorithm 1 has a running time in $O(n(G)^3)$ and solves 2-level CLP. In particular, PLP and CLP with two levels can be solved in polynomial time.*

Proof. It is not obvious that all operations are well defined and we therefore give an explanation for these at first: We can always construct an induced set of constraint C_k as described in the proof of Lemma 18 (however, it might be inconsistent). We can test for all the non-singular components K_j whether they are level-planar consistent with C and construct such a drawing $\mathcal{K}_j$ if they are because of Lemma 19. The level-planar drawings $\mathcal{K}_, \ldots, \mathcal{K}_l$ can then always be combined into a single level-planar drawing because of Lemma 18. Lemma 15 guarantees us that we can add all single vertices to the drawing at the end.

It is now easy to see that every drawing $\mathcal{G}'$ returned by Algorithm 1 is a level-planar drawing of G and respects C, since the combined drawing of the $\mathcal{K}_j$ is a level-planar drawing drawing of all non-singular components consistent with C.

It is only left to show that if the input G is in CLP Algorithm 1 always computes a drawing, i.e., that we never return *no*. With Lemma 15 it suffices to show that every input graph G is in CLP if there exists a partial graph G' containing all non-singular components of G that is level-planar respecting C. We saw in Lemma 18 that for every level-planar drawing there exists a consistent order $\prec_K$ on the set of non-singular components induced by C. So for G level-planar, C_k is always consistent. Further if G is level-planar respecting C, all components must be too. This gives us that we never return *no* and algorithm 1 computes a level-planar drawing $\mathcal{G}$ of G if G is level-planar respecting C.

The running time of our algorithm depends on the following operations: We can identify all components in linear time. Generating a partial order C_K and testing its consistency can be done in $O(n(G)^2)$ time because $|C| \in O(n(G)^2)$. Testing the level-planarity of and drawing a component K_j requires $O(n(G)^2)$ time according to Lemma 19, so performing this step for all components requires at most $O(n(G)^3)$ time. Combining these drawings and adding the single vertices can be done in $O(n^2)$ time because of Lemma 15. This results in a running time of $O(G(n)^3)$, thus Algorithm 1 requires only a polynomial amount of time. □

We can extend this result to Radial PLP.

Theorem 21. *Radial PLP with only two levels is solvable in polynomial time.*

Proof. Let G be a 2-level graph together with a level assignment $\ell\colon V(G) \to [2]$ and let $\mathcal{H}$ be a level-planar partial drawing of G. We have to consider a few cases:

For the first case we assume that $\mathcal{H}$ contains at least one edge $e = (v_1, v_2)$. This edge must have endpoints on both level 1 and level 2. Therefore e cuts through all levels and we can break $\mathcal{H}$ at e by duplicating e into two edges $e = (v_1, v_2)$ and $e_d = (v_{1,d}, v_{2,d})$ and transforming $\mathcal{H}$ into a plane level-planar drawing $\mathcal{H}'$ of G as was described in Chapter 2 and demonstrated in Figure 2.3. Now $\mathcal{H}'$ induces a set of constraints C on G if we consider G together with e_d as a plain level-graph. For each level $i \in [2]$ and every vertex $v \in V_i$ we add the constraints $(v_i < v)$ and $(v < v_{i,d})$ to C. For every level-planar drawing $\mathcal{G}'$ of G' respecting C we know that the edges e, e_d are at the leftmost respectively rightmost side of $\mathcal{G}'$. So if there exists a level-planar drawing $\mathcal{G}'$ of G' respecting C we know that we can fuse e, e_d and obtain a radial level-planar drawing $\mathcal{G}$ of G respecting $\mathcal{H}$. The other way around, if there exists a radial level-planar drawing $\mathcal{G}$ of G we can break $\mathcal{G}$ at e the same way we did with $\mathcal{H}$, and obtain a plain level-planar drawing $\mathcal{G}'$ respecting C. The constructed problem is an instance of 2-level CLP, which is solvable in polynomial time as shown in Theorem 20.

We now consider the case that $\mathcal{H}$ does not contain an edge but a vertex v with a degree of at least one in G. We assume without loss of generality that $\ell(v) = 1$. Let $e = (v, v')$ be an incident edge of v. Then there are only $O(n(G))$ possibilities for v' to be placed on level 2 inside of $\mathcal{H}$. This gives us at most $O(n(G))$ new partial drawings $\mathcal{H}_j$, and if there exists a level-planar drawing $\mathcal{G}$ of G respecting $\mathcal{H}$ then $\mathcal{G}$ also respects exactly one of the $\mathcal{H}_j$. So we can apply the first case to all $\mathcal{H}_j$ and G is radial level-planar respecting $\mathcal{H}$ if and only if G is radial level-planar respecting one of the $\mathcal{G}_j$, and every radial level-planar drawing respecting one of the $\mathcal{H}_j$ clearly also respects $\mathcal{H}$.

Now let us assume that $\mathcal{H}$ contains only single vertices but G contains at least one edge $e = (v_1, v_2)$. For both v_1, v_2 there are only $O(n(G))$ possibilities to be placed inside of $\mathcal{H}$. This gives us at most $O(n(G)^2)$ new partial drawings $\mathcal{H}_j$, and if there exists a level-planar drawing $\mathcal{G}$ of G respecting $\mathcal{H}$ then $\mathcal{G}$ also respects exactly one of the $\mathcal{H}_j$. So we can apply the first case to all $\mathcal{H}_j$ and G is radial level-planar respecting $\mathcal{H}$ if and only if G is radial level-planar respecting one of the $\mathcal{G}_j$, and every radial level-planar drawing respecting one of the $\mathcal{H}_j$ clearly also respects $\mathcal{H}$.

For the last case assume that G does not contain an edge. In this case we can add all vertices not already contained in $\mathcal{H}$ at an arbitrary place and gain a radial level-planar drawing $\mathcal{G}$ of G respecting $\mathcal{H}$. This can be done in linear time.

Since these were all possible cases Radial PLP with only two levels is solvable in polynomial time. □

So far the NP-hardness results have pretty similar between the Radial and the Plain versions of Level Planarity. Thus, the following result is even more surprising, as it provides a major gap between Radial CLP and CLP.

Theorem 22. *Radial CLP is NP-hard even for one level.*

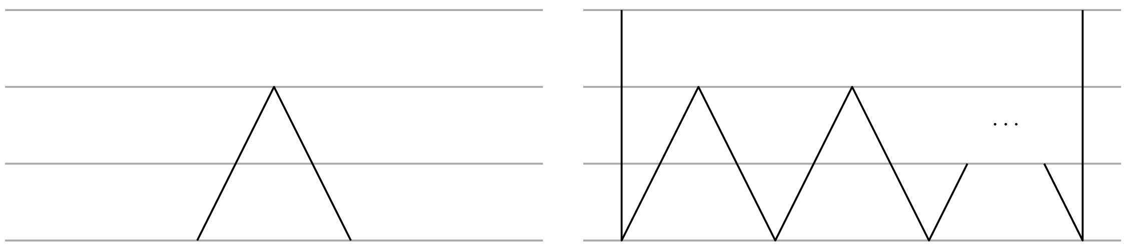
Proof. Radial CLP with only one level is equivalent to the problem of finding a saturated partial cyclic order Λ' containing a given partial cyclic order Λ. This problem is NP-hard [GM77]. □

In Section 3.1 we saw that 7-level PLP is NP-hard via reduction from 3-Partition. As we can reduce PLP to Radial PLP without increasing the number of levels, this holds for Radial PLP as well. However, if we reduce the NP-hardness of Radial PLP directly from 3-Partition we can even decrease the number of required levels.

Theorem 23. *Radial PLP is NP-hard even for six levels.*

Proof. Recall the NP-hardness reduction from 3-Partition to PLP [BR22a]. Observe that, in this reduction, the first layer was only needed to ensure that the plugs are not drawn next to the bucket construction but inside of it. If we perform the same reduction onto Radial PLP we can identify the leftmost and the rightmost edge chain of the partial drawing, i.e., the left side of the leftmost bucket and the right side of rightmost bucket with each other since the reduction ensured that nothing could be drawn outside of it in any level-planar drawing. By combining these two sides we now have a radial partial drawing with no outside at which the plugs can be drawn, so the first level can be omitted. This leaves six levels. The rest follows from the correctness proof of Brückner and Rutter [BR22a]. □

Using a similar approach, we could show that CLP is NP-hard already for six levels by requiring all plugs to lie between the leftmost and rightmost socket vertices. However, the following approach gives us an even better bound.



(a) A mountain.

(b) A mountain chain.

Fig. 3.8: The drawings of a mountain and a mountain chain. The chain is made up of mountains glued together by their forth-level vertices and with an edge wall from level 1 to level 4 at both ends. We can see that we have some kind of valley between adjacent mountains in the chain and between the left- respectively rightmost mountain and the neighboured walls. The vertices in the drawing are omitted for clarity.

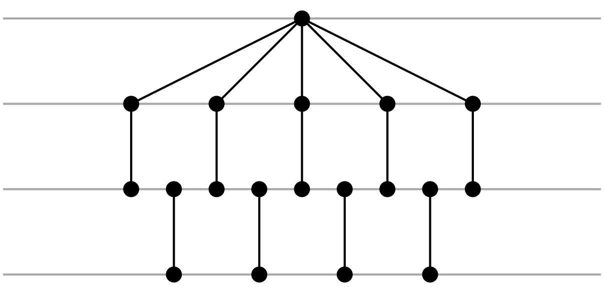


Fig. 3.9: A 4-clip. The order of all clip vertices on level 3 is fixed by constraints. Therefore in the third level there is always precisely one vertex with an adjacent vertex in level 4 between two vertices with adjacent vertices in level 2.

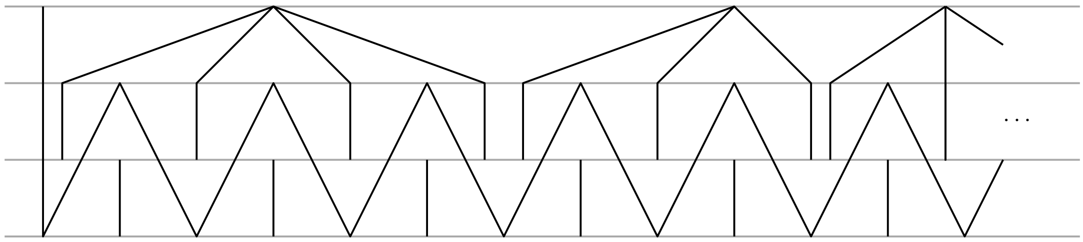


Fig. 3.10: A demonstration of how we can draw a mountain chain together with a number of clips compact as possible, i.e., in such a way that every mountain contains a clip edge from level 3 to level 4.

Definition 24 (Mountain Chain)**.** *A* mountain *as shown in Figure 3.8a is a graph consisting out of five vertices* v_1, v_2, v_3, v_4, v_5 *with a level assignment* $\ell(v_1) = \ell(v_5) = 4$, $\ell(v_2) = \ell(v_4) = 3$ *and* $\ell(v_3) = 2$ *and edges* $(v_3, v_2), (v_3, v_4), (v_2, v_1), (v_4, v_5)$*. We can combine k mountains into a* mountain k-chain *as shown in Figure 3.8b by identifying in each case a* v_1 *of one mountain with a* v_5 *of another one, and adding a wall consisting out of an edge chain from level* 4 *to level* 1 *to the two unpaired vertices on the sides. We can see that each mountain k-chain contains* $k+1$ valleys*, one between each pair of adjacent mountains and one between the outer mountains and the adjacent walls, respectively.*

We will in the following always assume that the drawing of a mountain chain is fixed by a saturated partial order C on it.

Definition 25 (k-Clip)**.** *A k-clip as shown in Figure 3.9 is graph consisting out of k edges between level* 3 *and level* 4*,* $k+1$ *edges between level* 2 *and level* 3 *and a vertex m on level* 1 *which is adjacent to all the clip vertices on level* 2*. We fix the drawing of each clip with constraints in such a way that on level* 3*, there is always precisely one vertex adjacent to a level-*4 *vertex between two vertices which are adjacent to level-*2 *vertices. The* 4*-clip in Figure 3.9 is drawn in the unique way that it respects these constraints.*

We will now see how these structures interact with each other.

Lemma 26. *Let* P_1, P_2 *be two clips. Then in every* 4*-level-planar drawing containing both* P_1, P_2 *the do not cross, i.e., on every level either all vertices of* P_1 *lie before all vertices of* P_2 *or the other way around.*

Let now M be a k-mountain chain and $\{P_1, \ldots, P_l\}$ *be a set of clips with clip number* $k_1, \ldots, k_l$ *respectively, and let* $\mathcal{G}$ *a* 4*-level-planar drawing of M and the* $P_1, \ldots, P_k$*. Then in* $\mathcal{G}$*, in every mountain of M there is drawn at most one edge between level* 3 *and level* 4 *out of all clips.*

If we further define constraints enforcing that all clips must be drawn between the walls of M, there exists a 4*-level-planar drawing* $\mathcal{G}$ *of M and the* $P_1, \ldots, P_l$ *respecting these new constraints if and only the sum of all clip numbers* $\sum_{j \in [l]} k_j \leq k$ *is at most the number of mountains in M.*

Proof. Let at first P_1, P_2 two clips in a level-planar drawing $\mathcal{G}$ and let $v_1 \in P_1, v_2 \in P_2$ be their unique level-1 vertices. We assume without loss of generality that in $\mathcal{G}$ v_1 is drawn to the left of v_2. We can then immediately see that on the levels $2, 3$ all vertices of the clip P_1 must be drawn before all vertices of the clip P_2. For the level 4 vertices of a clip it holds that they must be drawn in the same order as their adjacent vertices on level 3, therefore the statement also follows for level 4.

Now let M be a k-mountain chain, $\{P_1, \ldots, P_l\}$ be a set of clips with clip number $k_1, \ldots, k_l$ respectively, and let $\mathcal{G}$ a 4-level-planar drawing of M and the $P_1, \ldots, P_k$. Let e_1, e_2 be two edges between level 3 and 4 out of two (not necessarily distinct) clips $P, P' \in \{P_1, \ldots, P_l\}$. We notice that between e_1, e_2 there must lie at least one edge leading from level 2 to level 3. This edge cannot be drawn inside of a mountain, so there must lie a at least one valley between e_1, e_2. So inside of every mountain there can be drawn at most one clip edge.

Now let again M be a k-mountain chain and $\{P_1, \ldots, P_l\}$ be a set of clips with clip number $k_1, \ldots, k_l$ respectively. If we enforce with constraints that every clip must lie between the walls of M, every edge inside of a clip which leads from level 3 to level 4 must lie inside of a mountain, and because we already showed that there can be at most one of these inside of a mountain and the total number of clip edges between level 3 and level 4 equals $\sum_{j \in [l]} k_j$, we know that $\sum_{j \in [l]} k_j \leq k$ must hold if there exists a level-planar drawing of M and the $P_1, \ldots, P_l$ respecting the new constraints.

Figure 3.10 demonstrates that and how we can draw the clips $P_1, \ldots, P_l$ inside of M in such a way that every mountain contains an edge and all constraints are fulfilled. Therefore if $\sum_{j \in [l]} k_j \leq k$ we can construct such a 4-level planar drawing of the $P_1, \ldots, P_l$ inside of M. ☐

We can now describe a reduction from 3-Partition to 4-level CLP

Let $m \in \mathbb{N}$ be a positive integer, A be a multiset with $n = 3m$ elements, and $B \in \mathbb{Z}^+$ be a bound, such that both $B/4 < a < B/2$ for all $a \in A$ and $\sum_{a \in A} a = m \cdot B$ hold.

We can now construct a graph G in the following way: For every $a \in A$ we construct an a-clip P_a which we add to G. We then construct m mountain chains $M_1, \ldots, M_m$ with B mountains each and combine them into an extended mountain chain by identifying the right wall of M_j with the left wall of M_{j+1} for all $j \in [m+1]$, giving us $m \cdot B$ mountains in total, which we also add to G. Finally we enforce with constraints that in every level-planar drawing of G the vertices of the left wall of M_1 must be leftmost vertices and the vertices of the right wall of M_m must be rightmost vertices in the drawing.

This leads to a reduction from 3-Partition to CLP with four levels.

Lemma 27. *Let $m \in \mathbb{N}$ be a positive integer, A be a multiset with $n = 3m$ elements, and $B \in \mathbb{Z}^+$ be a bound, such that both $B/4 < a < B/2$ for all $a \in A$ and $\sum_{a \in A} a = m \cdot B$ hold. Let further G be the 4-level graph together with the set of constraints C constructed with (m, A, B) as described above.*

Then there exists a level-planar drawing $\mathcal{G}$ of G if and only if there exist m disjoint sets $A_1, A_2, \ldots, A_m$, such that for all $j \in [m]$ the equation $\sum_{a \in A_j} a = B$ holds.

Proof. Let $m \in \mathbb{N}$ be a positive integer, A be a multiset with $n = 3m$ elements, and $B \in \mathbb{Z}^+$ be a bound, such that both $B/4 < a < B/2$ for all $a \in A$ and $\sum_{a \in A} s(a) = m \cdot B$ hold. Let further G be the 4-level graph together with the set of constraints C constructed with (m, A, B, s) as described above.

If there exists a level-planar drawing $\mathcal{G}$ of G we know that all $m \times B$ mountains in G contain exactly one clip edge between level 3 and 4, because there exist exactly $\sum_{a \in A} a$ of these, each mountain can contain at most one (see Lemma 26) and every clip must lie entirely between the leftmost and the rightmost wall of the extended mountain chain. We can further see that a clip must lie entirely in one mountain chain, because it can cross one of the inner mountain walls in the extended mountain chain. We can now construct m sets $A_1, A_2, \ldots, A_m$ by assigning to every A_j for $j \in [m]$ exactly those $a \in A$ for which the clip P_a lies entirely in the mountain chain M_j. Then for every $j \in [m]$ it holds that $\sum_{a \in A_j} a = B$.

Now let $A_1, A_2, \ldots, A_m$ be disjoint sets such that $\sum_{a \in A_j} a = B$ holds for all $j \in [m]$. We know that we can draw every mountain chain M_j level-planar if we draw exactly the P_a into M_j for which $a \in A_j$ because of Lemma 26. Since $\bigcup_{j \in [m]} A_j = A$, all clips $P_a, a \in A$ can be drawn simultaneously by this method. This results in a level-planar drawing $\mathcal{G}$ of G. □

We can now state the following lower bound on required number of levels such that CLP gets NP-hard.

Theorem 28. *CLP is NP-hard even for four levels.*

Proof. We showed in Lemma 27 that we can reduce 3-Partition to CLP with four levels. Since 3-Partition is strongly NP-complete, this gives us that it is NP-hard to solve CLP even for 4 levels. □

We sum up what we found out about the complexity of our problems regarding their number of levels h in the following table. The information is given in the form of x/y, where x denotes the largest number of levels for which we know that we can solve the problem in polynomial time and y denotes the smallest number of levels for which we know that the problem is NP-hard. The numbers between are of open complexity status.

Level Planarity	*Plain*	*Radial*
Partial	*2 (Thm. 20) / 7 ([BR22a])*	*2 (Thm. 21) / 6 (Thm. 23)*
Constrained	*2 (Thm. 20) / 4 (Thm. 28)*	*— / 1 (Thm. 22)*

Of special interest seems to be the case with three levels. It seems quite possible, that PLP is solvable in polynomial time for three levels, but CLP is not. Therefore, a solution for the question which of these problems can be solved in polynomial time for three levels could provide a useful insight into the differences between PLP and CLP.

4 Vertex Cover Number

In this chapter, we take a look at the vertex cover number of the input graph G.

Definition 29 (Vertex Cover)**.** *Let G be a graph, and let $\mathcal{C} \subseteq V(G)$ be a set with the property that for each edge $e = \{u, v\} \in E(G)$ $v \in \mathcal{C}$ or $u \in \mathcal{C}$. We then say that $\mathcal{C}$ is a* vertex cover *of G. If, for any vertex cover $\mathcal{C}'$ of G with $\mathcal{C}' \subseteq \mathcal{C}$, it holds that $|\mathcal{C}'| \leq |\mathcal{C}|$, then we say that $\mathcal{C}$ is* minimal. *If there exists no vertex cover $\mathcal{C}'$ of G such that $|\mathcal{C}'| < |\mathcal{C}|$ then $\mathcal{C}$ is a* minimum *vertex cover of G and G has* vertex cover number *$|\mathcal{C}|$.*

Deciding whether G has vertex cover number at most k for a given $k \in \mathbb{N}$ is NP-hard [Kar72]. On the other hand, there exist several 2-approximation algorithms for finding a minimum vertex cover. For example identifying a maximal Matching M of G and assigning all vertices involved in M to a set C yields a 2-approximation. This is the case because every edge of these edges needs to be covered by a different vertex (thus C is at most twice as big compared to a minimum vertex cover), and every edge e gets covered since otherwise we could grow M with e.

The aim of this chapter is to analyse CLP and PLP parameterized by the vertex cover number of the graph G.

Now let $\mathcal{C}$ be a vertex cover of G. For every subset $X \subseteq \mathcal{C}$, let

$$V_X = \{v \in V(G) \setminus \mathcal{C} \mid N(v) = X\}$$

be the set of vertices in $V(G) \setminus \mathcal{C}$ that are adjacent precisely to the vertices X. Note that this means that every vertex in $V(G) \setminus \mathcal{C}$ is contained in precisely one of the V_X.

We claim the following about the set V_X.

Lemma 30. *Let G be a planar graph, and let $\mathcal{C}$ be a vertex cover of G. Let further $X \subseteq \mathcal{C}$ be a subset of the vertex cover with $|X| \geq 3$. Then it holds that $|V_X| \leq 2$.*

Proof. Assume that $\{v_1, v_2, v_3\} \subseteq V_X$ and let $\{v_1', v_2', v_3'\} \subseteq X$. Then these vertices form the complete bipartite graph $K_{3,3}$ which is hence a minor of G. But G was planar by assumption. This is to a contradiction. Therefore, $|V_X| \leq 2$ must hold. ☐

This gives us a bound on the size of most of the V_X. The following two lemmas give us a similar result for the case $|X| = 2$:

Lemma 31. *Let G be a planar graph with a proper level assignment $\ell\colon V(G) \to [h]$, and let $\mathcal{C}$ be a vertex cover of G. Let further $X \subseteq \mathcal{C}$ be a subset of the vertex cover with $|X| = \{x, y\}$. If G is level-planar and the two vertices x, y lie on the same level ($\ell(x) = \ell(y)$) then it holds that $|V_X| \leq 2$.*

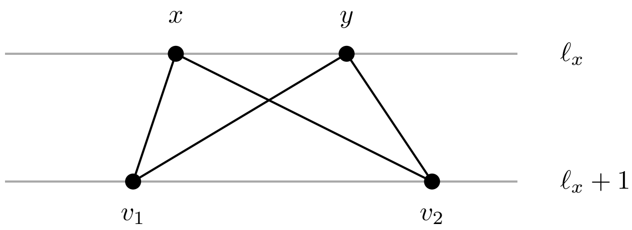


Fig. 4.1: A crossing generated by more than one vertex with level $\ell_x + 1$ if $|X| = 2$. We can see that the crossing generated by this constellation remains even if we swap v_1, v_2 or x, y.

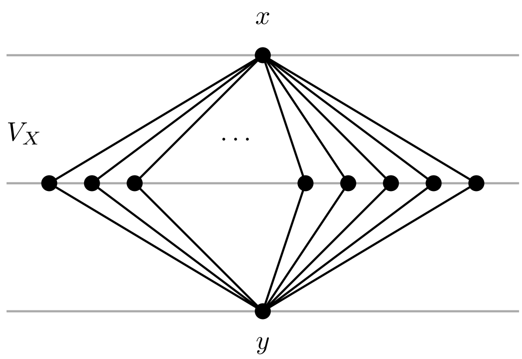


Fig. 4.2: An example of a level-planar structure with a subset $X = \{x, y\} \subseteq \mathcal{C}$ of the vertex cover $\mathcal{C}$ and with V_X arbitrarily large.

Proof. Let $\ell_x = \ell(x)$ denote the level of both vertices in X. Because G is proper, for every vertex $v \in V_X$, either $\ell(v) = \ell_x + 1$ or $\ell(v) = \ell_x - 1$ must hold.

Now let $v_1, v_2 \in V_X$ be two neighbours of X such that $\ell(v_1) = \ell(v_2) = \ell_x + 1$. We can see that in every level drawing of G this enforces a crossing because in every level drawing the vertices v_1, v_2, x, y have to form a constellation as can be seen in Figure 4.1. Hence, there can be at most one vertex $v \in V_X$ with $\ell(v) = \ell_x + 1$.

With the same argument we can show that there exists at most one vertex $v \in V_X$ with $\ell(v) = \ell_x - 1$. Since these were the only two options, it holds that $|V_X| \leq 2$. □

Unfortunately if $|X| = 2$ and the two vertices $x, y \in X$ do not lie on the same level, then V_X can be arbitrarily large as can be seen in Figure 4.2. The following lemma shows us how we can deal with this problem.

Lemma 32. *Let G be a planar graph with a proper level assignment $\ell \colon V(G) \to [h]$, let C be a closed set of constraints on $V(G)$, and let $\mathcal{C}$ be a minimal but not necessarily minimum vertex cover of G. Further let $X \subseteq \mathcal{C}$ be a subset of the vertex cover with $|X| = x, y$, $|V_X| \geq 1$ and with $\ell(x) \neq \ell(y)$.*

Then there exists a graph $G' \subseteq G$ with the same vertex cover $\mathcal{C}$ and a new (and without loss of generality closed) set of constraints on $V(G')$ with the following properties:

1. *For every set of vertices $X' \subseteq \mathcal{C}$ and for the associated set of vertices $V'_{X'} \subseteq V(G') \setminus$*

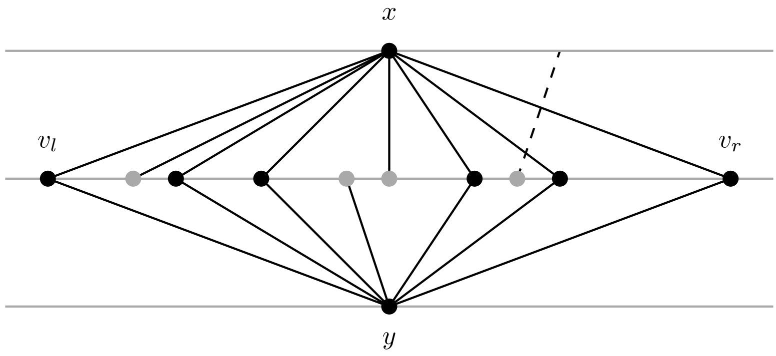


Fig. 4.3: A visualisation of the situation in Lemma 4.2. We have a set of vertex cover vertices $X = \{x, y\}$ and a drawing of the whole graph. The vertices v_l, v_r are the left- and the rightmost vertex of V_X in the drawing, respectively. Note that each vertex $v \notin V_X$ (grey) drawn between v_l, v_r must be in V_x or V_y, since no vertex of degree 0 is contained in the graph and every edge (e.g. the dashed edge) between v and a vertex not in X creates a crossing with an edge incident to v_l or v_r.

$\mathcal{C}$ consisting of precisely the vertices connected to X', it holds that $|V'_{X'}| \leq |V_{X'}|$.

2. *For the set of vertices V'_X it holds that $|V'_X| = 1$.*
3. *G is level-planar respecting C if and only if G' is level-planar respecting C'.*

Given G we can compute a new graph G' with the above properties in polynomial time.

Proof. Let G be a planar graph with a proper level assignment $\ell \colon V(G) \to [h]$, let C be a closed set of constraints on $V(G)$ and let $\mathcal{C}$ be a minimal, but not necessarily minimum vertex cover of G. Let further $X \subseteq \mathcal{C}$ be a subset of the vertex cover with $X = \{x, y\}$, $|V_X| \geq 1$ and with the two vertices $x, y \in X$ lying on different levels, i.e., $\ell(x) \neq \ell(y)$.

We assume without loss of generality that G contains no singletons. (Otherwise we know that, because of Lemma 15, G is level-planar if and only if G without all its singletons is level-planar, and since $\mathcal{C}$ is a minimal vertex cover we know that no singletons are contained in $\mathcal{C}$. Therefore the V_X stay the same for all $X \subseteq \mathcal{C}$, except for $V_\varnothing$. Since then $V_\varnothing = \varnothing$, this is no problem for our following result.

Since $|V_X| \geq 1$, there exists at least one vertex $v \in V_X$ that is connected to both x and y by definition of V_X. Since $\ell(x) \neq \ell(y)$ and our level-assignment is proper, one of the x, y must lie one level above v and the other one must lie one level below v. We therefore know that $\ell(x) = \ell(y) + 2$ or $\ell(x) = \ell(y) - 2$ must hold. We assume without loss of generality that $\ell(x) = \ell(y) - 2$. Then for all vertices $v' \in V_X$, it holds that $\ell(v') = \ell(x) + 1$, i.e., all vertices in V_X lie on the same level between x, y.

Assume that G is level-planar, let $\mathcal{G}$ be a level-planar drawing of G, and let $v_l, v_r \in V_X$ be the leftmost respectively rightmost vertex out of V_X in $\mathcal{G}$. Let further $v \notin V_X$ be a vertex drawn somewhere between v_l, v_r in $\mathcal{G}$. We know from assumption that v s not singleton, and therefore hast at least one neighbour. This neighbour needs to be either x

or y (every other neighbour would necessarily generate a crossing with one of the edges from v_l or v_r to x or y, and it cannot be both since then $v \in V_X$ would hold). Since $\mathcal{C}$ is minimal it holds that $v \notin \mathcal{C}$. A visualization of this situation can be seen in in Figure 4.3. Therefore either $v \in V_{\{x\}}$ or $v \in V_{\{y\}}$ holds.

We can observe now that we can reorder all vertices between v_l, v_r arbitrarily without changing the level-planarity of the drawing. If there exist constraints $v_1 \prec_C v$ and $v \prec_C v_2$ for some $v_1, v_2 \in V_X$, then v must lie somewhere between the V_X in every level-planar drawing of G respecting C. Since C is closed we can now reorder the vertices between v_l, v_r in such a way that the vertices $v \notin V_X$, $v_1 \prec_C v$ and $v \prec_C v_2$ for some $v_1, v_2 \in V_X$ are the only ones lying inside of X.

This shows that if G is level-planar there exists a level-planar drawing $\mathcal{G}$ of G such that between the V_X lie only vertices which have constraints of both forms $v_1 \prec_C v$ and $v \prec_C v_2$ for some $v_1, v_2 \in V_X$ (and we will further denote this set by V_X^*).

Contracting V_X^* into a single vertex v_{new} yields a new graph G' with the claimed properties (1) and (2). We further define the new set of constraints C' in such a way that it is consistent with C on $V(G) \setminus V_X^*$ and that for each $v \in V(G) \setminus V_X^*$ there exists a constraint of the form $v \prec_{C'} v_{\text{new}}$ (respectively $v_{\text{new}} \prec_{C'} v$) if there exists a constraint of the form $v \prec_C v'$ (respectively $v' \prec_C v$) for a $v' \in V_X^*$.

We can immediately see that G' is level-planar if G is level-planar because every level-planar drawing $\mathcal{G}$ of G respecting C can be transformed into a level-planar drawing $\mathcal{G}'$ of G' respecting C' by the above method.

If on the other hand there exists a level-planar drawing $\mathcal{G}'$ for G' respecting C', we can replace v_{new} and its incident edges with the vertices in V_X^* ordered in some way respecting C and because of the way we defined C' no vertex in V_X^* has a constraint conflict of a vertex outside of V_X^*. This therefore gives us a level-planar drawing $\mathcal{G}$ of G.

Therefore, G is level-planar if and only if G' is level-planar and property (3) holds.

The possibility to construct a such a graph G' for a given G in polynomial time follows directly from the definition of G'. □

We know have a way to handle the size of all V_X with $|X| \geq 2$. What is left to do is to deal with the leaves.

Lemma 33. *Let G be a planar graph with a proper level assignment $\ell\colon V(G) \to [h]$, let C be a closed set of constraints on $V(G)$ and let $\mathcal{C}$ be a minimum but not necessarily minimal vertex cover of G. Let further $V^* = \bigcup V_{\{v\}}, v \in \mathcal{C}$ be the set of vertices in $V(G) \backslash \mathcal{C}$ neighboured to exactly one vertex in $\mathcal{C}$, let $H \subseteq G$ be the subgraph of G containing all but the vertices V^* and their incident edges and let $\mathcal{H}$ be a level-planar drawing of H respecting C.*

Then it is possible to test whether there exists a level-planar drawing $\mathcal{G}$ of G which coincides with $\mathcal{H}$ on H in $O^*(2^{|\mathcal{C}|^2})$ time.

Proof. The idea of this proof is to show that we can define new constraints with the property that for every level j we can insert the leaves lying on this level if and only if there exists an order of level j consistent with all new constraints.

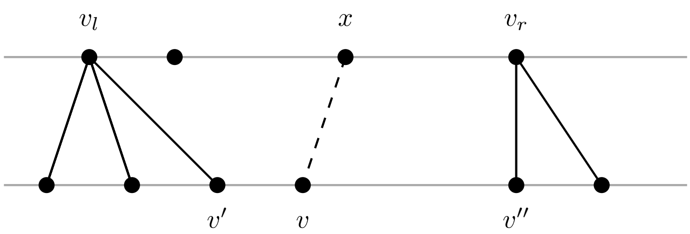


Fig. 4.4: A visualization of the situation in the proof of Lemma 33 where we want to ensure that the newly inserted edge (x, v) does not generate a crossing with an edge already drawn. The vertex v_l is the rightmost vertex to the left of x with an edge (v_l, v') to some vertex v' on level $j + 1$, and v_r is the leftmost vertex to the right of x with an edge (v_r, v'') to some vertex v' on level $j+1$. The vertex v' is the rightmost neighbour of v_l on level $j + 1$ and v'' is the leftmost neighbour of v_r on level $j + 1$. We can see that e crosses an already existing edge if and only if e crosses either (v_l, v') or (v_r, v'').

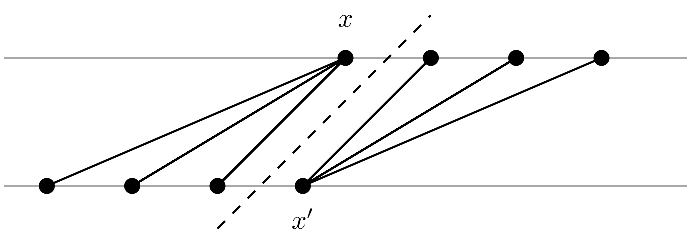


Fig. 4.5: A visualization of the situation in the proof of Lemma 33 where we want to ensure that leafs of vertex cover vertices x, x' on adjacent levels do not generate a crossing. The set $V_{x'}$ must lie either entirely to the left or entirely to the right of x (and in the same way the set V_x must lie either entirely to the right or entirely to the left of x', depending on the choice of side made before).

We therefore take a look at all sorts of crossings which can occur while inserting the leaves. Let $x \in \mathcal{C}$ be a vertex out of the vertex cover on level j and let $v \in V_x$ be a leaf adjacent to x. Assume without loss of generality that $e = (x, v) \in E(G)$ (otherwise, $(v, x) \in E(G)$ and the proof is analogue).

The first kind of crossing that can occur while inserting (x, v) is a crossing between e and an edge already present in the drawing. To prevent that, let v_l be the rightmost vertex to the left of x with an edge (v_l, v') to some vertex v' on level $j + 1$, and let v_r be the leftmost vertex to the right of x with an edge (v_r, v'') to some vertex v' on level $j + 1$. Let further v' be the rightmost neighbour of v_l on level $j + 1$ and v'' be the leftmost neighbour of v_r on level $j + 1$. This situation is visualized in Figure 4.4. We can see that e crosses an already existing edge if and only if e crosses either (v_l, v') or (v_r, v''). Therefore, in every level-planar drawing the constraints $v' \prec v$ and $v \prec v''$ must be fulfilled, and enforcing them by adding them to C guarantees that e generates no crossing with an already existing edge.

All other possible crossings we need to prevent can only occur between two newly added edges. We can observe that for $v, v' \in V_x$ the edges (x, v) and (x, v') never cross. It therefore suffices to consider edges of the form (x', v') or (v', x'), $x' \in \mathcal{C}$, $v' \in V_{x'}$, $x \neq x'$.

We will at first show how to prevent crossings with edges of the form (x', v'), where x, x' both lie on level j. Let assume without loss of generality that x lies to the left of x'. We can now see that in order to generate a level-planar drawing all vertices of V_x must lie entirely to the left of all vertices out of $V_{x'}$, and if this is fulfilled no crossing between the edge e and an edge (x', v') occurs. Therefore it is necessary and sufficient to add the constraints $v \prec v'$ for all $v \in V_x$, $v' \in V_{x'}$ to C in order to prevent these crossings.

So far, all additional constraints could be generated and added simultaneously in polynomial time. This changes in the last case we need to consider. Let $x' \in \mathcal{C}$ be a vertex cover vertex on level $j + 1$ with an non-empty set of neighboured leaves $V_{x'} \neq \varnothing$. This situation is visualized in Figure 4.5. We can see now that the set $V_{x'}$ must lie either entirely to the left or entirely to the right of x (and in the same way the set V_x must lie either entirely to the right or entirely to the left of x', depending on the choice of side made before). This is ensured by enforcing either the additional constraints $v \prec x'$ for all $v \in V_x$ and $x \prec v'$ for all $v' \in V_{x'}$, or the constraints $x' \prec v$ for all $v \in V_x$ and $v' \prec x$ for all $v' \in V_{x'}$, and one of these situations must hold in every level-planar drawing of G. There at most $|\mathcal{C}|^2$ pairs $x, x' \in \mathcal{C}$ for which this situation can occur, there are at most $O(2^{|\mathcal{C}|^2})$ combinations of constraints of this kind. It is now easy to see that there exists a level-planar drawing of G if and only if there exists an order for the levels respecting at least one of the generated set of constraints.

All of this can be tested in $O^*(2^{|\mathcal{C}|^2})$ time. □

We can now state the main result of this chapter:

Theorem 34. *Proper CLP and Proper PLP are in FPT parameterized by Vertex Cover Number.*

Proof. Let G be a graph with Vertex Cover Number k together with a proper level assignment $\ell\colon V(G) \to [h]$ and a closed set of constraints C. We will show that there exists an FPT algorithm with which we can test whether G is level-planar parameterized by Vertex Cover Number.

We can assume without loss of generality that G is planar (otherwise we immediately know that G cannot be level-planar) and that G has no singular vertices (because of Lemma 15 we know G is level-planar if and only if G without all its singular vertices is level-planar).

We can now find a Vertex Cover $\mathcal{C}$ for G with size at most $2 \cdot k$ in polynomial time, and we can assume without loss of generality that $\mathcal{C}$ is minimal (otherwise we could just delete vertices out of $\mathcal{C}$ until the resulting set cannot be reduced any further).

From the Lemmata 30, 31 and 32 we know that we can assume that $|V_X| \leq 2$ holds for every $|X| \geq 2$. (This is already guaranteed for most cases according to the Lemmata 30 and 31. If another case occurs, Lemma 32 guarantees us that we can reduce G to a new graph G' with the desired property in polynomial time.)

Define the set of leaves $V^* = \{v \in V(G) \setminus \mathcal{C} \mid \deg(v) = 1\} \subset V(G) \setminus \mathcal{C}$ as all vertices not contained in $\mathcal{C}$ but neighboured to precisely vertex out of $\mathcal{C}$. We now consider the subgraph $H \subseteq G$ containing all of G except V^* and its incident edges. From the observations made above it follows that H contains at most $|C| + 2 \cdot 2^|C| \leq 2 \cdot k + 2^{2 \cdot k+1}$ vertices, and we could generate H in polynomial time. Therefore, the size of H depends only on k and there exists a function $f\colon \mathbb{N} \to \mathbb{N}$ such that we can construct all possible level-planar drawings of H by brute force in $O(f(k))$ time.

For every level-planar drawing $\mathcal{H}$ of H we can further test in $O^*(2^{|\mathcal{C}|^2})$ time whether we can add the remaining leaves V^* because of Lemma 33.

If there exists at least one level-planar drawing $\mathcal{H}$ of H which can be expanded to a level-planar drawing $\mathcal{G}$ of G respecting C, then G is level-planar. The other way around it is easy to see that if G is level-planar, there must exist at least one level-planar drawing $\mathcal{H}$ of H which can be expanded to a level-planar drawing $\mathcal{G}$ of G respecting C.

We can therefore test whether G is level-planar respecting the constraints C with an FPT-algorithm parameterized by Vertex Cover Number.

Since Proper PLP can be reduced to Proper CLP while maintaining the same Vertex Cover Number, this result holds for Proper PLP as well. □

Unfortunately, it is not clear if and how one can expand this result to the non-proper cases. We can see that we heavily relied on a proper level assignment in the Lemmata 31, 32 and 33. The statement out of Lemma 31 becomes wrong if we omit the assumption that the level-assignment is proper, and it is unclear whether Lemmata 32 and 33 can be generalized to this case. The general question whether CLP and PLP are in FPT parameterized by Vertex Cover Number therefore remains open.

5 Maximal Number of Vertices per Level

In this chapter we investigate another parameter intrinsic to Level Planarity: the maximal number of vertices per level. However, it turns out that in the general case CLP and PLP are not in FPT parameterized by this number. We will see that this changes if we require our input instance to be proper.

To show that general CLP is not in FPT, we will prove that we can reduce the problem whether a given instance of CLP or PLP is drawable to the problem with at most two vertices per level. Brückner and Rutter [BR22b] provided a technique with which we can reduce every h-level graph to a (non-proper) $O(n)$-level graph with exactly one vertex per level. We now describe a simplified version of this reduction technique, which is demonstrated in Figure 5.1:

Theorem 35 ([BR22b]). *Let G be an h-level graph together with a level assignment $\ell\colon V(G) \to [h]$, and $j \in [h]$ be an arbitrary level with y vertices on it. We can expand level j by replacing it with y new levels $j_1, j_2, \ldots, j_y$ and distributing the y vertices to these new levels. Let G'_j be the instance obtained in this way, and let G' be the instance in which every level is expanded with this method. Let further G'' be the (proper) instance obtained from G' in which every edge crossing k layers is subdivided k times.*

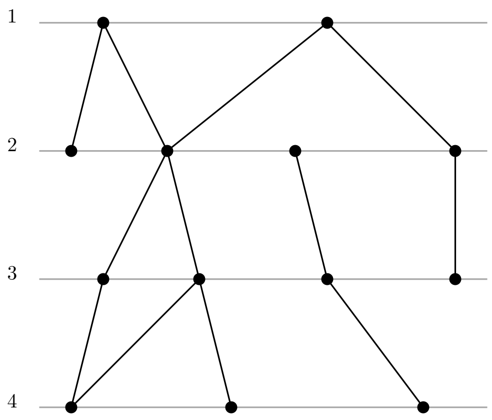
Then the sizes of G'_j, G' and G'' are polynomial in the size of G, and G, G'_j, G', G'' are all level-planar if and only if one of them is level-planar. Note further that in G' each level contains at most one vertex. Having a level-planar drawing $\mathcal{G}$ of G, we can perform all these expansion steps on $\mathcal{G}$ without changing the embedding.

It is not obvious for our constrained setting how to transfer the additional set of constraints or the partial drawing. As only proper PLP reduces to a special case of CLP, we have to consider these problems separately. For PLP we can expand the partial drawing as we did with the input graph.

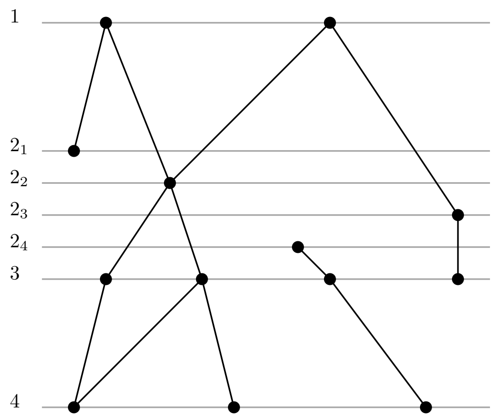
Theorem 36. *Let G be an h-level graph together with a level assignment $\ell\colon V(G) \to [h]$ and a partial drawing $\mathcal{H}$ of a subgraph $H \subseteq G$. Let G' be a new graph with only one vertex per level, obtained by expansion of all levels as described in Theorem 35. Let further $\mathcal{H}'$ be a drawing of H respecting the new level assignment of G' but with the same embedding as $\mathcal{H}$. Then G has a level-planar drawing respecting $\mathcal{H}$ if and only if G' has a level-planar drawing respecting $\mathcal{H}'$.*

Furthermore there exists no FPT-algorithm solving PLP parameterized by the maximum number of vertices per level under the assumption that $P \neq NP$.

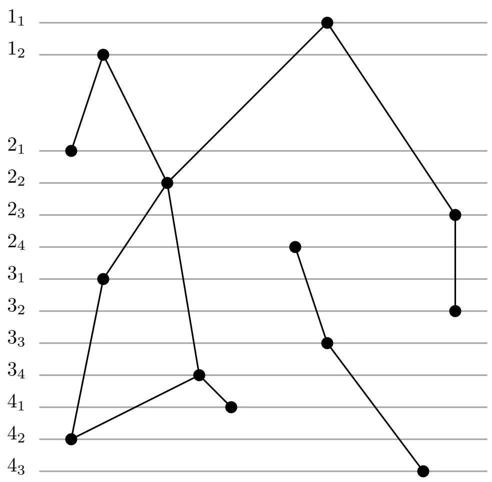
Proof. Assume that there exists a level-planar drawing $\mathcal{G}$ of G respecting $\mathcal{H}$. If we perform the expansion steps described in Theorem 35 on every level, we gain a level-planar drawing $\mathcal{G}'$ of G'. Let $\mathcal{H}'$ be the part of $\mathcal{G}'$ that corresponds to H. Clearly $\mathcal{G}'$ respects



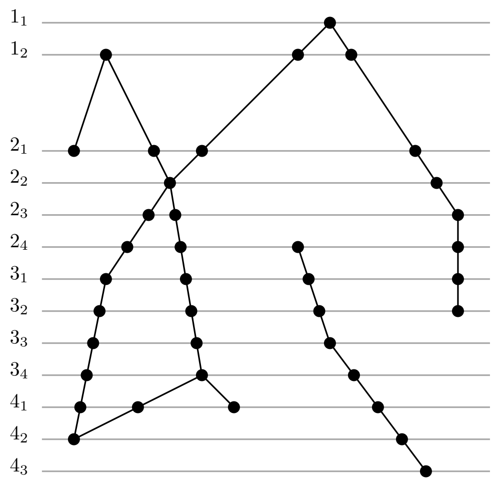
(a) A level-graph with at most four vertices per level.



(b) The same level-graph in which line 2 has been expanded.



(c) A fully expanded graph. Every level contains at most one vertex.



(d) A proper expanded graph. Every edge that spans several levels got subdivided.

Fig. 5.1: The simplified version of the expansion technique described by Brückner and Rutter [BR22b].

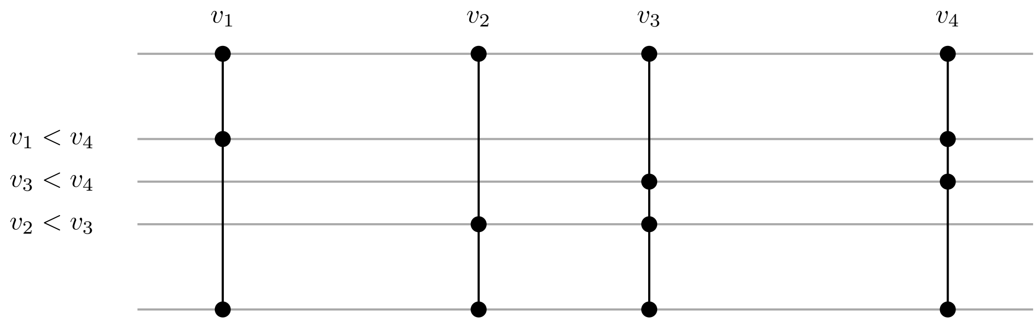


Fig. 5.2: A level after constraint expansion.

$\mathcal{H}'$. Since the expansion steps can be performed without changing the embedding of either drawing $\mathcal{G}, \mathcal{H}$ or $\mathcal{H}'$ is equivalent to $\mathcal{H}$. Thus $\mathcal{G}'$ is a level-planar drawing of G' respecting $\mathcal{H}'$.

Now assume that there exists a level-planar drawing $\mathcal{G}''$ of G'' respecting $\mathcal{H}'$. By Theorem 35, we know that there exists a level-planar drawing $\mathcal{G}$ of G with the same embedding as $\mathcal{G}''$. Since the expansion steps can be performed without changing the embedding of either drawing, $\mathcal{G}$ constructed in this way coincides with $\mathcal{H}$ on H. Thus $\mathcal{G}$ is a level-planar drawing of G respecting $\mathcal{H}$.

Since we can reduce the question whether there exists a level planar drawing for a given PLP-instance to the class of level-graphs with at most one vertex per level, and PLP is NP-hard [BR17], there cannot exist an FPT-algorithm solving PLP parameterized by the number of vertices per level under the assumption that P $\neq$ NP. □

For CLP we have to deal with our constraints first. Let G be an h-level graph together with a level assignment $\ell \colon V(G) \to [h]$, let $C = \bigcup_{j \in [h]} C_j$ be a set of constraints and let $j \in [h]$ be an arbitrary level with the constraints C_j. We can now expand level j in the following way:

At first generate a duplicate level j_d of level j and all vertices on it, and place this new level j_d directly under level j. For every vertex v on level j, we denote the duplicate of v with v_d.

Then for every vertex v on level j we leave all incoming edges of the former v intact as they are, add an edge (v, v_d) and reoriginate all former outgoing edges of v from v_d. For every constraint $c = (v < v')$ in C_j, we then insert a new level j_c in between the levels j, j_d, subdivide the edge between the vertices v, v' and their respective duplicates v_d, v'_d with the two vertices v_c, v'_c, remove c from C_j and add a new constraint $c_d = (v_c < v'_c)$ to the constraint set $C_{j,c}$ of level j_c. We call such an expansion step a *constraint expansion* of level j, and denote the resulting graph by G^*_j. Figure 5.2 shows a level after constraint expansion. The level and all corresponding vertices have been duplicated, and every constraint is realized on a new distinct level in between. Let further G^* be the graph obtained from G in which every level got constraint expanded, and $G^{*'}$ be the graph obtained from G^* by expanding all the levels without constraints as described in Theorem 35. Note that this means that we expand precisely all levels except those of the form j_c

for a constraint $c \in C_j$.

Having defined this constraint expansion, we can now formulate the following Theorem:

Theorem 37. *Let G be an h-level graph together with a level assignment $\ell\colon V(G) \to [h]$ and let $C = \bigcup_{j\in[h]} C_j$ be a set of constraints for G. Let further $j \in [h]$ be an arbitrary level of G.*

Then in every level-planar drawing of the graph G_j^* (obtained from G by constraint expansion of level j) respecting the new constraints, the vertices lie in the same order on both level j and its duplicate level j_d, and they respect all constraints originally present in C_j. Further, the size of the graph $G^{*'}$ is polynomial in the size of G, every level of $G^{*'}$ contains at most two vertices, and there exist level-planar drawings for all of the $G_j^*, G^*, G^{*'}$ their respective set of constraints if and only if there exists a level-planar drawing $\mathcal{G}$ of G respecting C.

Furthermore there exists no FPT-algorithm solving CLP parameterized by the maximum number of vertices per level under the assumption that $P \neq NP$.

Proof. Let G_j^* be a graph obtained from G by constraint expansion of level j. We will show at first that in every level-planar drawing $\mathcal{G}_j^*$ of G_j^*, the vertices formerly on level j lie in the same order on both the original level j and the duplicate level j_d, and respect all constraints originally present in C_j.

If we take a look at the (possibly subdivided) lines between the vertices and their duplicates, we can observe that they can not cross at any point, so they form a number of parallel lines, with the same line order at any horizontal cut in this interval. This immediately implies that the original vertices and their duplicates have the same vertex order. Furthermore, every level with a constraint in between the levels j, j_d enforces its order on the vertices with which we subdivided these lines, and with that on the overall line order. We can therefore contract the levels j, j_d and all levels in between into a single level fulfilling C_j in every drawing of G_j^*.

On the other side, given a level-planar drawing of G we can directly expand level j in the drawing in accordance with the constraint expansion rules and gain a level-planar drawing of G_j^*.

This directly results in G having a level-planar drawing respecting C if and only if G_j^* has a level-planar drawing respecting its new set of constraints, and – through iteration of this argument – if and only if G^* has a level-planar drawing respecting its new set of constraints. Theorem 35 finally gives us that G^* has a level-planar drawing respecting its set of constraints if and only if $G^{*'}$ has a level-planar drawing respecting its set of constraints.

Since every constraint expansion generates only $O(|C|)$ many new levels, the size of C is at most quadratic in the number of vertices n and we perform this constraint expansion step at most n times, the size of $G^{*'}$ is polynomial in the size of G, and Theorem 35 guarantees that the size of $G^{*'}$ is polynomial in the size of G^*, and thus also polynomial in the size of G.

Furthermore, every level in $G^{*'}$ with constraints contains only two vertices after the constraint expansion, and every other level contains only one vertex after the expansion

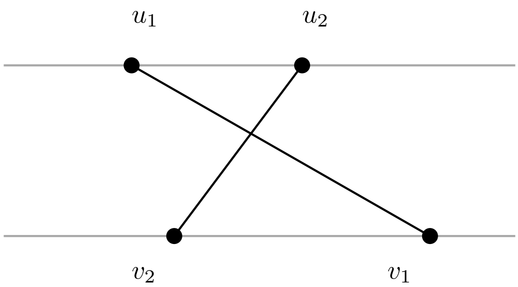


Fig. 5.3: Two edges $(u_1, v_1), (u_2, v_2)$ which lie between the same levels cross if and only if u_1 lies before u_2 and v_1 after v_2; or if u_1 lies after u_2 and v_1 before v_2.

method from Theorem 35, therefore every level contains at most two vertices.

The last thing we need to show is that there cannot exist an FPT-algorithm for CLP parameterized by the number of vertices per level. Since we can reduce the question whether there exists a level planar drawing for a given CLP-instance to the class of level-graphs with at most two vertices per level, and CLP is NP-hard [BR17], there can not exist an FPT-algorithm solving CLP parameterized by the number of vertices per level under the assumption that P $\neq$ NP. □

As a side effect this gives us the following corollary:

Corollary 38. *Assuming P $\neq$ NP, there exists no FPT-algorithm for CLP parameterized by the maximum number of constraints per level. This does not change if we require our input instance to be proper.*

Proof. Theorem 37 shows that we can reduce the question whether an instance of CLP is level-planar on the instances with at most one constraint per level. Such an instance can be made proper by subdividing every edge spanning several levels, without increasing the number of constraints, therefore there cannot exist an FPT-algorithm solving (proper) CLP parameterized by the maximum number of constraints per level. □

It is noteworthy that in order to prove theorems 36 and 37, we required a lot of edges to pass over several levels. We can prohibit this phenomenon by requiring the input graph to be proper. It turns out that this restricted case lies in FPT.

We first need a way to test planarity of certain embeddings.

Lemma 39. *Let $G = (V, E)$ be an h-level graph together with a level function ℓ and constraints C, with at most k vertices per level. Let further $j \in [h-1]$ and P_j, P_{j+1} be a linear order for the vertices V_j respectively V_{j+1} of level $j, j+1$. respecting C.*

Then we can test in $O(k^2)$ time whether the drawing of G restricted to $V_j \cup V_{j+1}$ and the edges between defined by P_j, P_{j+1} is planar.

Proof. To test planarity it suffices to check if there are any crossing edges. Two edges $(u_1, v_1), (u_2, v_2)$ which lie between the same levels have to cross if and only if u_1 lies before u_2 and v_1 after v_2; or if u_1 lies after u_2 and v_1 before v_2, as shown in Figure 5.3.

Given two edges this property can be tested in $O(1)$ time. As we can have at most a linear number of edges in a planar graph, we can test all edge pairs in $O(k^2)$ time. □

Algorithm 2: (h-level graph G, level function ℓ, constraints C)

if $h = 0$ **then**
    return yes
$M_1 \leftarrow \varnothing$
foreach permutation P of L_1 consistent with C **do**
    $M_1 \leftarrow M_1 \cup \{P\}$
for $i = 2$ to h **do**
    $M_i \leftarrow \varnothing$
    foreach permutation P of L_i consistent with C **do**
        for $P' \in M_{i-1}$ **do**
            if P' and P together with edges between layers $i - 1$ and i planar **then**
                $M_i \leftarrow M_i \cup P'$
if $M_h = \varnothing$ **then**
    return no
return yes

Using this consistency test, we can formulate Algorithm 2.

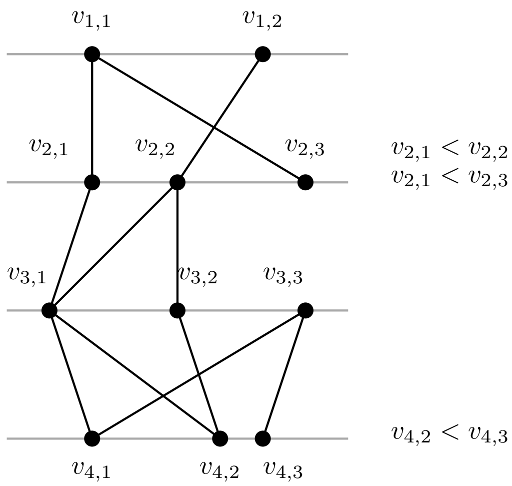
The idea behind this algorithm is the following: For each level, we compute all possible permutations, and test afterwards which of them are compatible with each other. If there exists a selection of permutations compatible with each other covering all levels those form a level-planar drawing of the graph.

Figure 5.4 shows a level graph (5.4a) and a level-planar drawing (5.4b) of it according to the computation of Algorithm 2 visualized by the tree in Figure 5.5. In this tree, all considered permutations (exactly those compatible with the constraints) are considered. All permutations in the first level are considered reachable. Those in the following levels are considered reachable if they are compatible with a reachable permutation in the previous level (and are otherwise pictured grey). The existence of a reachable permutation on the last level is equivalent to the existence of a level-planar drawing.

Theorem 40. *Let G be a graph together with a level function $\ell\colon V(G) \to [h]$ and a set of constraints C be a proper instance of CLP with at most k vertices per level. Then Algorithm 2 decides whether there exists a drawing $\mathcal{G}$ of G compatible with C in $O(n \cdot (k!)^2 \cdot k^4)$ time.*

In particular there exists an FPT-algorithm for Proper CLP (and thus Proper PLP) parameterized by k.

Proof. We will show at first that the algorithm returns *yes* if and only if the input is level-planar.



(a) A representation of a level graph G. Some constraints are given at the side.

(b) A level-planar drawing of G compatible with the given constraints.

Fig. 5.4: An instance of Proper CLP, on which we demonstrate Algorithm 2.

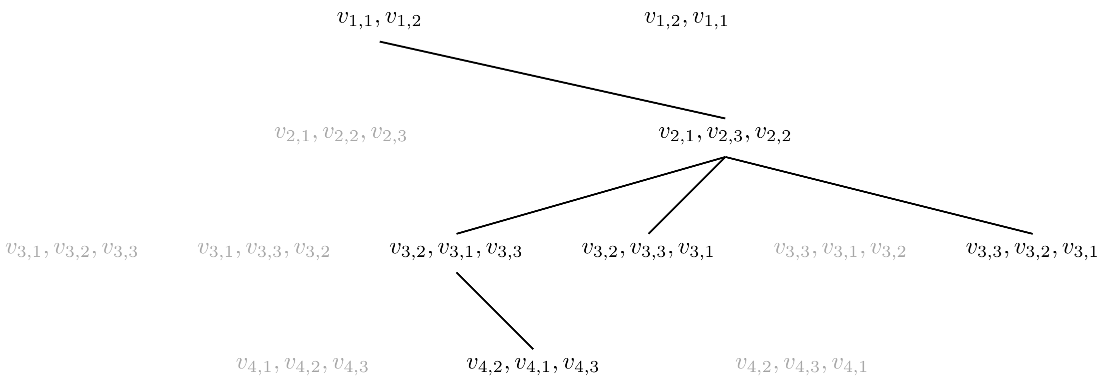


Fig. 5.5: A computation tree demonstrating how Algorithm 2 works on the graph G shown in Figure 5.4. On each level we consider all permutations compatible with C. Every permutation contained in a set M_i is drawn black, all others are drawn gray. An edge consists from permutation P to permutation P', if they can be drawn level-planar together and P is contained in a set M_i. We see that G can be drawn level-planar since there exists a permutation in M_4, and the (in this case) only possible level-planar drawing of G can be seen in Figure 5.4b.

If $h = 0$, G must be the empty graph which is trivially level-planar. This case is checked in the beginning. We will therefore assume in the rest of this correctness proof that $h \geq 1$, and write G^j for the graph consisting exactly of the first j levels out of G and the edges between them.

We will show that the following invariant is true at the beginning of each iteration of the second *for* loop:

For the set M_{i-1} the equation $M_{i-1} \neq \varnothing$ holds if and only if there exists at least one level-planar drawing of G^{i-1}, and there exists a level-planar drawing of G^{i-1} respecting C in which level $i-1$ has the total order P if and only if $P \in M_{i-1}$.

Since we can draw every 1-level graph with exactly those permutations which are compatible with C, and these are exactly those we add to M_1 in the first *for* loop, the invariant is fulfilled at the beginning of the first iteration of the second *for* loop.

Now let the invariant be true at the start of an iteration process and let $2 \leq i \leq h$. Since G_i is proper, every level planar drawing of G_i consists of a level-planar drawing of G_{i-1} and a level-planar drawing of the graph consisting of levels $i-1, i$ and the edges between those two levels, while these two drawings must coincide at level $i-1$ and both must be compatible with C. We know that according to the invariant the permutations of level $i-1$ for which a level-planar drawing exists are precisely those in M_{i-1}, so it suffices to check for every permutation P of level i if it is consistent with C and if there exists a permutation $P' \in M_{i-1}$ such that the level drawing of levels $i-1, i$ and the edges between those two levels is planar if the levels are ordered as in P respectively P'. We add precisely those permutations for which this condition is fulfilled. Therefore at the end of the iteration step M_i contains permutation P' of level i if and only if there exists a level-planar drawing of G_i in which level i is ordered as in P', and the invariant holds.

The iteration process of the second *for* loop ends if $i = h + 1$, so according to the invariant $G^h = G$ has a level-planar drawing compatible with C if and only if $M_h \neq \varnothing$.

This proves that Algorithm 2 correctly decides whether there exists a level-planar drawing of G.

It remains to show that it also has the claimed running time. Since there are at most k vertices per level, we can generate all level permutations in $O(k!)$ time, and for each permutation P we require at most $O(k^2)$ time to check if P is consistent with C (since there are at most $O(k^2)$ constraints per level). Given two permutations P, P' consistent with C for two successive levels we can test in $O(k^2)$ time if the respective levels can be drawn level-planar together if we order their vertices according to P, P' as shown in Lemma 39.

Overall this gives us a running time of $O(k! \cdot k^2 + h \cdot k! \cdot k^2 \cdot k! \cdot k^2)$, which is in $O(n \cdot (k!)^2 \cdot k^4)$ as $h \in O(n)$, thus Algorithm 2 has the claimed running time and is an FPT-algorithm for PROPER CLP parameterized by k. □

Note that although Algorithm 2 decides only whether the input has a level-planar drawing and does not compute a drawing itself, we could easily gain a drawing from the M_j, if for every permutation we also save a predecessor. Starting from a reachable permutation P in M_h and choosing the line of predecessors then leads to a level-planar drawing.

6 Conclusion

In this thesis we investigated Constrained and Partial Level Planarity parameterized by several interesting parameters. The problem turned out to not be in FPT parameterized by the number of levels, pathwidth or treewidth. Since these problems are solvable in polynomial time for one level, this lead to the question how many levels are necessary to make these problems (and their radial analogues) NP-hard. While Radial CLP proved to be NP-hard even for one level, we showed that it is possible to solve CLP and (Radial) PLP with at most two levels in polynomial time. Brückner and Rutter [BR22a] showed that PLP is NP hard with seven levels, which we improve to four levels for CLP and six levels for Radial PLP. The remaining levels between these bounds remain of unclear complexity status. Improving these bounds, or even solving the question at which levels these problems become NP-hard and if this number is different for CLP and (Radial) PLP is an interesting subject for future work.

The next parameter we considered was Vertex Cover Number. We proved that Proper CLP (and thus Proper PLP) is in FPT parameterized by Vertex Cover Number. However, it is not clear if and how this approach can be generalized to CLP without requiring that the input graph is proper. Therefore, the question whether there exists an FPT algorithm for this parameter remains open.

Another parameter we considered was the maximal number of vertices per level. By expanding the given graphs, we demonstrated that PLP can be reduced to PLP with at most one vertex per level and CLP can be reduced to CLP with at most two vertices per level, and that therefore there does not exist a general FPT algorithm for PLP or PLP parameterized by the maximal number of vertices per level. However we presented an FPT algorithm for Proper CLP (and thus Proper PLP).

There are a few more parameters we did not investigate in this thesis but which would be an intriguing subject for future work.

A promising parameter seems to be the number of sources (i.e., the number of vertices with no incoming edge). Current algorithms for Level Planarity are based on algorithms for Level Planarity with only one source present [BR21]. Further, there exists an algorithm solving CLP in polynomial time if there is only one source present [BR17], which has a very similar structure to the one source Level Planarity algorithm. Trying to combine these approaches seems promising in order to obtain an FPT algorithm for CLP parameterized by the number sources.

Other interesting parameters we did not research are the treedepth of an input instance or the number of constraints respectively the size of a partial drawing.

Finally, whether these results (especially those for vertex cover number and the maximal number of levels) can be generalized to Radial CLP and Radial PLP is a compelling question.

Bibliography

[ADD$^+$15] 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). *Theoretical Computer Science*, 571:1–9, 2015, 10.1016/j.tcs.2014.12.019.

[BBF04] Christian Bachmaier, Franz J. Brandenburg, and Michael Forster: Radial level planarity testing and embedding in linear time. In Giuseppe Liotta (editor): *Graph Drawing*, pages 393–405, Berlin, Heidelberg, 2004. Springer Berlin Heidelberg.

[BR17] Guido Brückner and Ignaz Rutter: Partial and constrained level planarity. *Proceedings of the Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 2000–2011, 2017, 10.1137/1.9781611974782.130.

[BR21] Guido Brückner and Ignaz Rutter: Radial level planarity with fixed embedding. *Journal of Graph Algorithms and Applications*, 25(1):353–366, 2021, 10.7155/jgaa.00561.

[BR22a] Guido Brückner and Ignaz Rutter: Partial and constrained level planarity. Unpublished manuscript, 2022.

[BR22b] Guido Brückner and Ignaz Rutter: Simplified level-planarity algorithms. Unpublished manuscript, 2022.

[Cha04] Hubert Chan: A parameterized algorithm for upward planarity testing. In Susanne Albers and Tomasz Radzik (editors): *Proceedings of tthe European Symposium on Algorithms (ESA)*, volume 3221 of *Lecture Notes in Computer Science*, pages 157–168, 2004, 10.1007/978-3-540-30140-0_16.

[DFK$^+$08] Vida Dujmovic, Michael R. Fellows, Matthew Kitching, Giuseppe Liotta, Catherine McCartin, Naomi Nishimura, Prabhakar Ragde, Frances Rosamund, Sue Whitesides, and David R. Wood: On the parameterized complexity of layered graph drawing. *Algorithmica*, 52:267–292, 2008, 10.1007/s00453-007-9151-1.

[DN88a] Giuseppe Di Battista and Enrico. Nardelli: Hierarchies and planarity theory. *IEEE Transactions on Systems, Man, and Cybernetics*, 18(6):1035–1046, 1988, 10.1109/21.23105.

[DN88b] Giuseppe Di Battista and Enrico Nardelli: Hierarchies and planarity theory. *IEEE Transactions on Systems, Man, and Cybernetics*, 18:1035–1046, 1988, 10.1109/21.23105.

[GJ75] Michael R. Garey and David S. Johnson: Complexity results for multiprocessor scheduling under resource constraints. *SIAM Journal on Computing*, 4(4):397–411, 1975, 10.1137/0204035.

[GM77] Zvi Galil and Nimrod Megiddo: Cyclic ordering is np-complete. *Theoretical Computer Science*, 5(2):179–182, 1977, 10.1016/0304-3975(77)90005-6, ISSN 0304-3975.

[GT01] Ashim Garg and Roberto Tamassia: On the computational complexity of upward and rectilinear planarity testing. *SIAM Journal of Computing*, 31(2):601–625, 2001, 10.1137/S0097539794277123.

[HH08] Martin Harrigan and Patrick Healy: Practical level planarity testing and layout with embedding constraints. In Seok Hee Hong, Takao Nishizeki, and Wu Quan (editors): *Graph Drawing*, pages 62–68, Berlin, Heidelberg, 2008. Springer Berlin Heidelberg, 10.1007/978-3-540-77537-9_9.

[HL06] Patrick Healy and Karol Lynch: Two fixed-parameter tractable algorithms for testing upward planarity. *International Journal of Foundations of Computer Science*, 17(5):1095–1114, 2006, 10.1142/S0129054106004285.

[HR92] Lenwood S. Heath and Arnold L. Rosenberg: Laying out graphs using queues. *SIAM Journal on Computing*, 21(5):927–958, 1992, 10.1137/0221055.

[HS73] Frank Harary and Allen J. Schwenk: The number of caterpillars. *Discrete Mathematics*, 6(4):359–365, 1973, 10.1016/0012-365X(73)90067-8, ISSN 0012-365X.

[JL99] Michael Jünger and Sebastian Leipert: Level planar embedding in linear time. In Jan Kratochvíl (editor): *Graph Drawing*, volume 1731 of *LNCS*, pages 72–81. Springer, 1999, 10.1007/3-540-46648-7_7.

[JLM98] Michael Jünger, Sebastian Leipert, and Petra Mutzel: Level planarity testing in linear time. In Sue H. Whitesides (editor): *Graph Drawing (LNCS,volume 1547)*, volume 1547 of *LNCS*, pages 224–237. Springer, 1998, 10.1007/3-540-37623-2_17.

[Kar72] Richard M. Karp: Reducibility among combinatorial problems. In Raymond E. Miller, James W. Thatcher, and Jean D. Bohlinger (editors): *Proceedings of the Symposium on Complexity of Computer Computations*, The IBM Research Symposia Series (IRSS), pages 85–103. Springer US, 1972, 10.1007/978-1-4684-2001-2_9.

[Meg76] Nimrod Megiddo: Partial and complete cyclic orders. *Bulletin of the American Mathematical Society*, 82(2):274–276, 1976, 10.1016/0304-3975(77)90005-6.

Erklärung

Hiermit versichere ich die vorliegende Abschlussarbeit selbstständig verfasst zu haben, keine anderen als die angegebenen Quellen und Hilfsmittel benutzt zu haben, und die Arbeit bisher oder gleichzeitig keiner anderen Prüfungsbehörde unter Erlangung eines akademischen Grades vorgelegt zu haben.

Würzburg, den April 30, 2022

. . . .
Marie Diana Sieper