Ordered Level Planarity and Its Relationship to Geodesic Planarity, Bi-Monotonicity, and Variations of Level Planarity

BORIS KLEMZ, Freie Universität Berlin, Germany
GÜNTER ROTE, Freie Universität Berlin, Germany

We introduce and study the problem ORDERED LEVEL PLANARITY which asks for a planar drawing of a graph such that vertices are placed at prescribed positions in the plane and such that every edge is realized as a y-monotone curve. This can be interpreted as a variant of LEVEL PLANARITY in which the vertices on each level appear in a prescribed total order. We establish a complexity dichotomy with respect to both the maximum degree and the level-width, that is, the maximum number of vertices that share a level. Our study of ORDERED LEVEL PLANARITY is motivated by connections to several other graph drawing problems.

GEODESIC PLANARITY asks for a planar drawing of a graph such that vertices are placed at prescribed positions in the plane and such that every edge e is realized as a polygonal path p composed of line segments with two adjacent directions from a given set S of directions which is symmetric with respect to the origin. Our results on ORDERED LEVEL PLANARITY imply $\mathcal{NP}$-hardness for any S with $|S| \geq 4$ even if the given graph is a matching. MANHATTAN GEODESIC PLANARITY is the special case where S contains precisely the horizontal and vertical directions. Katz, Krug, Rutter and Wolff claimed that MANHATTAN GEODESIC PLANARITY can be solved in polynomial time for the special case of matchings [GD'09]. Our results imply that this is incorrect unless $\mathcal{P} = \mathcal{NP}$. Our reduction extends to settle the complexity of the BI-MONOTONICITY problem, which was proposed by Fulek, Pelsmajer, Schaefer, and Štefankovič.

ORDERED LEVEL PLANARITY turns out to be a special case of T-LEVEL PLANARITY, CLUSTERED LEVEL PLANARITY, and CONSTRAINED LEVEL PLANARITY. Thus, our results strengthen previous hardness results. In particular, our reduction to CLUSTERED LEVEL PLANARITY generates instances with only two non-trivial clusters. This answers a question posed by Angelini, Da Lozzo, Di Battista, Frati and Roselli.

CCS Concepts: • **Mathematics of computing** → *Graph theory*; *Graph algorithms*; • **Human-centered computing** → **Graph drawings**; • **Theory of computation** → **Problems, reductions and completeness**; *Graph algorithms analysis*.

Additional Key Words and Phrases: Graph drawing, Level Planarity, orthogeodesic drawings, point-set embeddability, NP-hardness, upward drawings

1 INTRODUCTION

In this paper we introduce ORDERED LEVEL PLANARITY and study its complexity. We establish connections to several other graph drawing problems (see Figure 1), which we survey in this first section.

We proceed from general problems to more and more constrained ones: Section 1.1 recalls the original version of LEVEL PLANARITY. Section 1.2 discusses several constrained variations of the problem. ORDERED LEVEL PLANARITY is defined in Section 1.3. The closely related problems GEODESIC PLANARITY and BI-MONOTONICITY are discussed in Section 1.4.

Section 1.5 summarizes the main results of this paper and gives an overview of the remaining chapters.

A preliminary version of this paper appeared in the proceedings of Graph Drawing and Network Visualization 2017 [22].
Authors' addresses: Boris Klemz, Freie Universität Berlin, Institut für Informatik, Takustraße 9, Berlin, 14195, Germany, klemz@inf.fu-berlin.de; Günter Rote, Freie Universität Berlin, Institut für Informatik, Takustraße 9, Berlin, 14195, Germany, rote@inf.fu-berlin.de.

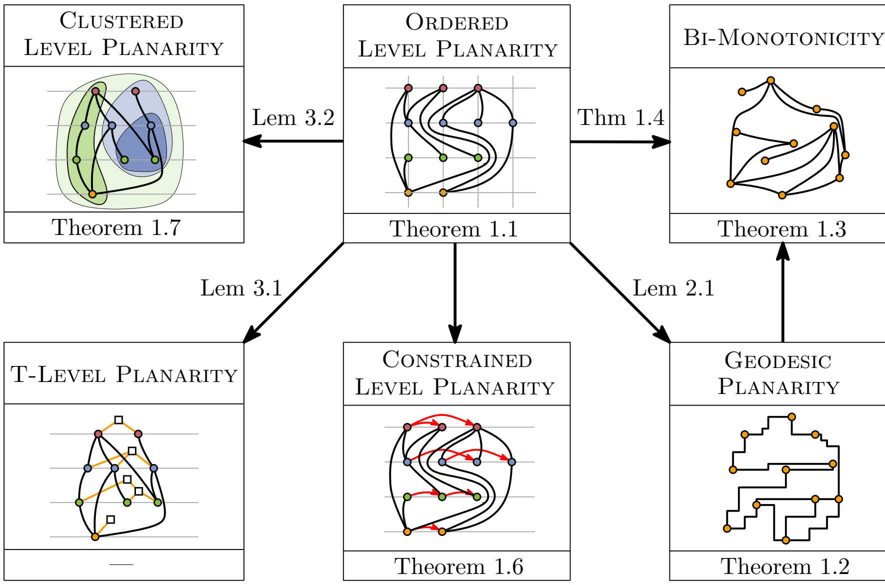


Fig. 1. Ordered Level Planarity is a special case of several other graph drawing problems.

1.1 Upward Planarity and Level Planarity

Upward Planarity. An *upward* planar drawing of a directed graph is a plane drawing (i.e., a crossing-free drawing in the plane) where every edge $e = (u, v)$ is realized as a y-monotone curve that goes upward from u to v. Such a drawing provides a natural way of visualizing a partial order on a set of items. The problem Upward Planarity of testing whether a directed graph has an upward planar drawing is $\mathcal{NP}$-complete [13]. However, if the y-coordinate of each vertex is prescribed, the problem can be solved in polynomial time [20]. This is captured by the notion of level graphs.

Level Planarity. A *level graph* $\mathcal{G} = (G, \gamma)$ is a directed graph $G = (V, E)$ together with a *level assignment*, i.e. a surjective map $\gamma\colon V \rightarrow \{0, \dots, h\}$ with $\gamma(u) < \gamma(v)$ for every edge $(u, v) \in E$. Value h is the *height* of $\mathcal{G}$. The vertex set $V_i = \{v \mid \gamma(v) = i\}$ is called the i-th *level* of $\mathcal{G}$ and $\lambda_i = |V_i|$ is its *width*. The *level-width* λ of $\mathcal{G}$ is the maximum width of any level in $\mathcal{G}$. A *level* planar drawing of $\mathcal{G}$ is an upward planar drawing of G where the y-coordinate of each vertex v is $\gamma(v)$, see Figure 2(b). The horizontal line with y-coordinate i is denoted by L_i. The problem Level Planarity asks whether a given level graph has a level planar drawing, see Figures 2(a–b).

The study of the complexity of Level Planarity has a long history [10, 12, 18–20], culminating in a linear-time algorithm by Jünger, Leipert and Mutzel [20]. Their algorithm is based on work for the special case of single-source level graphs by Di Battista and Nardelli [10]. There was an earlier attempt by Heath and Pemmaraju [18] to extend the work by Di Battista and Nardelli [10] to general level graphs; however, Jünger et al. [19] pointed out gaps in this construction. All these approaches utilize PQ-trees. Various simpler but asymptotically slower approaches to solve Level Planarity have been considered, see the work of Fulek, Pelsmajer, Schaefer, and Štefankovič [12] for one of these approaches (cf. Section 1.4) and a more comprehensive summary. Level Planarity has been extended to drawings of level graphs on surfaces different from the plane [1, 4, 5]. In particular, Radial Level Planarity [4], Cyclic Level Planarity [1, 5] and Torus Level Planarity [1] arrange levels on a standing cylinder, a rolling cylinder, and a torus, respectively.

Proper Instances. An important special case are *proper* level graphs, that is, level graphs in which $\gamma(v) = \gamma(u) + 1$ for every edge $(u, v) \in E$. Instances of Level Planarity can be assumed to be proper without loss of generality by subdividing long edges [10, 20]. However, in variations of Level Planarity where we impose additional constraints, the assumption that instances are proper can have a strong impact on the complexity of the respective problems [2]. The definition of proper instances naturally extends to the following variations of level graphs.

1.2 Level Planarity with Various Constraints

Clustered Level Planarity. Forster and Bachmaier [11] introduced a version of Level Planarity that allows the visualization of vertex clusterings. A *clustered* level graph $\mathcal{G}$ is a triple $(G = (V, E), \gamma, T)$ where (G, γ) is a level graph and T is a *cluster hierarchy*, i.e. a rooted tree whose leaves are the vertices in V. Each internal node of T is called a *cluster*. We call the cluster of the root *trivial* as it contains all vertices. All other clusters are called *non-trivial*. The *vertices* of a cluster c are the leaves of the subtree of T rooted at c. A cluster hierarchy is *flat* if all leaves have distance at most two from the root, i.e. if non-trivial clusters are not nested. A *clustered* level planar drawing of a clustered level graph $\mathcal{G}$ is a level planar drawing of (G, γ) together with a closed simple curve for each cluster that encloses precisely the vertices of the cluster such that the following conditions hold: (i) no two cluster boundaries intersect, (ii) every edge crosses each cluster boundary at most once, and (iii) the intersection of any cluster with the horizontal line L_i through level V_i is either a line segment or empty for any level V_i, see Figure 2(f). The problem Clustered Level Planarity asks whether a given clustered level graph has a clustered level planar drawing. Forster and Bachmaier [11] presented an $O(h|V|)$-time algorithm for a special case of proper clustered level graphs, where h is the height of $\mathcal{G}$. Angelini, Da Lozzo, Di Battista, Frati and Roselli [2] provided a quartic-time algorithm for all proper instances. The general version of Clustered Level Planarity is $\mathcal{NP}$-complete even for clustered level graphs with maximum degree $\Delta = 2$ and level-width $\lambda = 3$; and for 2-connected series-parallel clustered level graphs [2]. In the current paper, we further strengthen these previous results (Theorem 1.7).

T-Level Planarity. This variation of Level Planarity considers consecutivity constraints for the vertices on each level. A T-*level* graph $\mathcal{G}$ is a triple $(G = (V, E), \gamma, \mathcal{T})$ where (G, γ) is a level graph and $\mathcal{T} = (T_0, \ldots, T_h)$ is a set of trees where the leaves of T_i are V_i. A T-*level planar* drawing of a T-level graph $\mathcal{G}$ is a level planar drawing of (G, γ) such that, for every level V_i and for each node u of T_i, the leaves of the subtree of T_i rooted at u appear consecutively along L_i. The problem T-Level Planarity asks whether a given T-level graph has a T-level planar drawing. Wotzlaw, Speckenmeyer, and Porschen [23] introduced the problem and provided a quadratic-time algorithm for proper instances with constant level-width. Angelini et al. [2] give a quartic-time algorithm for proper instances with unbounded level-width. For general T-level graphs the problem is $\mathcal{NP}$-complete [2] even for T-level graphs with maximum degree $\Delta = 2$ and level-width $\lambda = 3$; and for 2-connected series-parallel T-level graphs.

Constrained Level Planarity. Very recently, Brückner and Rutter [6] explored a variant of Level Planarity in which the left-to-right order of the vertices on each level has to be a linear extension of a given partial order. They refer to this problem as Constrained Level Planarity and they provide an efficient algorithm for single-source level graphs and show $\mathcal{NP}$-completeness for connected proper level graphs.

1.3 A Common Special Case

Ordered Level Planarity. We introduce a natural variant of Level Planarity that specifies a total order for the vertices on each level. An *ordered* level graph $\mathcal{G}$ is a triple $(G = (V, E), \gamma, \chi)$

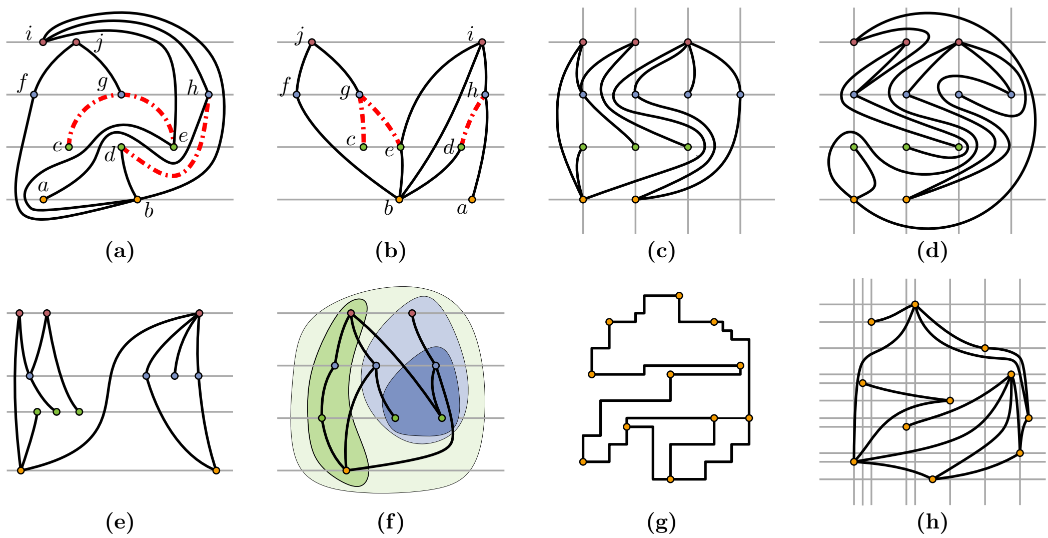


Fig. 2. In Level Planarity the order of the vertices of a common level V_i is not fixed. Finding a good ordering is an essential part of finding a solution. The ordering suggested in **(a)** is not realizable as the edge (d, h) cannot be drawn without crossing (c, g) or (e, g). **(b)** A level planar drawing of (a). As shown in this paper, fixing the ordering (Ordered Level Planarity) renders the problem intractable. **(c)** An ordered level drawing of the instance given in **(d)**. **(e)** An equivalent drawing for the relaxed version of the problem. **(f)** A clustered level drawing. **(g)** A manhattan geodesic drawing. **(h)** A bi-monotone drawing.

where (G, γ) is a level graph and $\chi : V \longrightarrow \{0, \ldots, \lambda - 1\}$ is a *level ordering* for G. We require that χ maps each level V_i $(= \gamma^{-1}(i))$ bijectively to $\{0, \ldots, \lambda_i - 1\}$. An *ordered* level planar drawing of an ordered level graph $\mathcal{G}$ is a level planar drawing of (G, γ) where for every $v \in V$ the x-coordinate of v is $\chi(v)$. Thus, the position of every vertex is fixed. The problem Ordered Level Planarity asks whether a given ordered level graph has an ordered level planar drawing, see Figures 2(c–d). In this paper, we show that Ordered Level Planarity is a common special case of all the Level Planarity variants defined in Section 1.2 (Theorem 1.5); and we provide a complexity dichotomy with respect to both the level-width and the maximum degree (Theorem 1.1).

Order and Realizability. In the above definition, the x-coordinates assigned via χ merely act as a convenient way to encode a total order for the vertices of each level V_i. Similarly, the y-coordinates assigned via γ encode a total preorder (i.e. a total ordering that allows ties) for the set of all vertices. In terms of realizability, the problem is equivalent to a generalized version where χ and γ range over arbitrary real numbers. In other words, the fixed vertex positions can be any points in the plane. All reductions and algorithms in this paper carry over to these generalized versions, if we pay the cost for presorting the vertices according to their coordinates. There is another equivalent version that is even more relaxed: we only require that the vertices appear according to the prescribed orderings without insisting on specific coordinates, see Figures 2(c–e). For the sake of visual clarity, many of the figures in this manuscript make use of this last equivalence, i.e. the vertices are arranged according to the orderings, but do not necessarily appear at the corresponding exact coordinates.

1.4 Geodesic Planarity and Bi-Monotonicity

Geodesic Planarity. Let $S \subset \mathbb{Q}^2$ be a finite set of directions which is symmetric with respect to the origin, i.e. for each direction $s \in S$, the reverse direction $-s$ is also contained in S. A plane drawing of a graph is *geodesic* with respect to S if every edge is realized as a polygonal path p composed of line segments with two adjacent directions from S. Two directions of S are *adjacent* if they appear consecutively in the projection of S to the unit circle. The name *geodesic* comes from the fact that such a path p is a shortest path with respect to some polygonal norm (a norm whose unit ball is a centrally symmetric polygon), which depends on S. An instance of the decision problem Geodesic Planarity is a 4-tuple $\mathcal{G} = (G = (V, E), x, y, S)$ where G is a graph, x and y map from V to the reals and S is a set of directions as stated above. The task is to decide whether $\mathcal{G}$ has a *geodesic drawing*, that is, G has a geodesic drawing with respect to S in which every vertex $v \in V$ is placed at $(x(v), y(v))$.

Katz, Krug, Rutter, and Wolff [21] study Manhattan Geodesic Planarity, which is the special case of Geodesic Planarity where the set S consists of the two horizontal and the two vertical directions, see Figure 2(g). Geodesic drawings with respect to this set of directions are also referred to as *orthogeodesic* drawings [14, 15]. Katz et al. [21] show that a variant of Manhattan Geodesic Planarity in which the drawings are restricted to the integer grid is $\mathcal{NP}$-hard even if G is a perfect matching. The proof is by reduction from 3-Partition and makes use of the fact that the number of edges that can pass between two vertices on a grid line is bounded. In contrast, they claim that the standard version of Manhattan Geodesic Planarity is polynomial-time solvable for perfect matchings [21, Theorem 5]. To this end, they sketch a plane sweep algorithm that maintains a linear order among the edges that cross the sweep line. When a new edge is encountered it is inserted as low as possible subject to the constraints implied by the prescribed vertex positions. When we asked the authors for more details, they informed us that they are no longer convinced of the correctness of their approach. Theorem 1.2 of our paper implies that the approach is indeed incorrect unless $\mathcal{P} = \mathcal{NP}$.

Bi-Monotonicity. Fulek, Pelsmajer, Schaefer, and Štefankovič [12] present a Hanani–Tutte theorem for *y-monotone* drawings, that is, upward drawings in which all vertices have distinct y-coordinates. They accompany their result with a simple and efficient algorithm for Y-Monotonicity, which can be defined as (Ordered) Level Planarity restricted to instances with level-width $\lambda = 1$. Moreover, they show that, even without the restriction on λ, Level Planarity is equivalent to Y-Monotonicity by providing an efficient reduction from Level Planarity. Altogether, this results in a simple quadratic time algorithm for Level Planarity.

Fulek et al. [12] propose the problem Bi-Monotonicity and leave its complexity as an open problem. Bi-Monotonicity combines Y-Monotonicity and X-Monotonicity, which is defined analogously to Y-Monotonicity. More precisely, the input of Bi-Monotonicity is a triple $\mathcal{G} = (G = (V, E), x, y)$ where G is a graph, and x and y are *injective* maps from V to the reals. The task is to decide whether $\mathcal{G}$ has a planar *bi-monotone* drawing, that is, a plane drawing in which edges are realized as curves that are both x-monotone and y-monotone, and in which every vertex $v \in V$ is placed at $(x(v), y(v))$, see Figure 2(h).

Bi-Monotonicity is very similar to Manhattan Geodesic Planarity. One difference is that Manhattan Geodesic Planarity imposes an implicit bound on the number of adjacent edges leading in similar directions, i.e. a vertex can have at most two neighbors in a single quadrant. The overall degree of each vertex is at most four. On the other hand, Bi-Monotonicity requires the coordinate mappings x and y to be injective. When both these additional constraints are satisfied, the problems are equivalent. In this paper, we exploit this relationship between the two problems in order to settle the question by Fulek et al. [12] regarding the complexity of Bi-Monotonicity (Theorem 1.3).

1.5 Main results

In Section 4 we study the complexity of Ordered Level Planarity. While Upward Planarity is $\mathcal{NP}$-complete [13] in general but becomes polynomial-time solvable [20] for prescribed y-coordinates, we show that prescribing both x-coordinates and y-coordinates renders the problem $\mathcal{NP}$-complete. We complement our result with efficient approaches for some special cases of ordered level graphs and, thereby, establish a complexity dichotomy with respect to the level-width and the maximum degree.

Theorem 1.1. *Ordered Level Planarity is $\mathcal{NP}$-complete, even for acyclic ordered level graphs with maximum degree $\Delta = 2$ and level-width $\lambda = 2$. The problem can be solved in linear time if the given level graph is proper; or if the level-width is $\lambda = 1$; or if $\Delta^+ = \Delta^- = 1$, where Δ^+ and Δ^- are the maximum in-degree and out-degree respectively.*

Ordered Level Planarity, especially if restricted to instances with $\lambda = 2$ and $\Delta = 2$, is an elementary problem that readily reduces to several other graph drawing problems. The remainder of this paper is dedicated to demonstrating the centrality of Ordered Level Planarity by providing reductions to all the problems listed in Sections 1.2 and 1.4. All these reductions heavily rely on either a small value of Δ or λ and they produce very constrained instances of the targeted problems. Thereby, we are able to solve multiple open questions that were posed by the graph drawing community. We expect that Theorem 1.1 may serve as a suitable basis for more reductions in the future.

In Section 2 we study Geodesic Planarity and obtain:

Theorem 1.2. *Geodesic Planarity is $\mathcal{NP}$-hard for any set of directions S with $|S| \geq 4$ even for perfect matchings in general position.*

Observe the aforementioned discrepancy between Theorem 1.2 and the claim by Katz et al. [21] that Manhattan Geodesic Planarity for perfect matchings is in $\mathcal{P}$.

Bi-Monotonicity is closely related to a special case of Manhattan Geodesic Planarity. With a simple corollary we settle the complexity of Bi-Monotonicity and, thus, answer the open question by Fulek et al. [12].

Theorem 1.3. *Bi-Monotonicity is $\mathcal{NP}$-hard even for perfect matchings.*

Theorem 1.4. *Ordered Level Planarity reduces to Bi-Monotonicity in linear time. The reduction can be carried out such that the input graph is identical to the output graph, that is, only the coordinates are modified.*

In Section 3 we establish Ordered Level Planarity as a special case of all the variations of Level Planarity described in Section 1.2.

Theorem 1.5. *Ordered Level Planarity reduces in linear time to Constrained Level Planarity and T-Level Planarity, and in quadratic time to Clustered Level Planarity.*

The reduction to Constrained Level Planarity is immediate, which also yields:

Theorem 1.6. *Constrained Level Planarity is $\mathcal{NP}$-hard even for acyclic level graphs with maximum degree $\Delta = 2$ and level-width $\lambda = 2$ and prescribed* total *orderings.*

Angelini, Da Lozzo, Di Battista, Frati, and Roselli [2] propose the complexity of Clustered Level Planarity for clustered level graphs with a flat cluster hierarchy as an open question. Our reduction to Clustered Level Planarity provides the following answer.

Theorem 1.7. *Clustered Level Planarity is $\mathcal{NP}$-hard even for acyclic clustered level graphs with maximum degree $\Delta = 2$, level-width $\lambda = 2$ and a flat cluster hierarchy that partitions the vertices into two non-trivial clusters.*

In general, we can consider two different versions of all of the above problems: we may prescribe a combinatorial embedding or allow an arbitrary embedding. Our results apply to both of these versions, as in most cases the instances are just systems of paths and, thus, the embedding is unique. The only exception is the linear time algorithm for proper instances of Ordered Level Planarity. In this case, however, yes-instances have a unique drawing and we only need to check if it respects the given embedding.

In order to be able to reduce from Ordered Level Planarity to Geodesic Planarity, our main reduction (to Ordered Level Planarity) is tailored to achieve a small maximum degree of $\Delta = 2$. As a consequence, the resulting graphs are not connected. At the cost of an increased maximum degree, it is possible to make our instances connected by inserting additional edges. We discuss these adaptations in Section 5.

Theorem 1.8. *The following problems are $\mathcal{NP}$-hard even for connected instances with maximum degree $\Delta = 4$:*

- *Ordered Level Planarity even for level-width $\lambda = 2$,*
- *Constrained Level Planarity even for level-width $\lambda = 2$ and prescribed* total *orderings,*
- *Clustered Level Planarity even for level-width $\lambda = 2$ and flat cluster hierarchies that partition the vertices into two non-trivial clusters, and*
- *Bi-Monotonicity.*

2 GEODESIC PLANARITY AND BI-MONOTONICITY

In this section we establish that deciding whether an instance $\mathcal{G} = (G, x, y, S)$ of Geodesic Planarity has a geodesic drawing is $\mathcal{NP}$-hard even if G is a perfect matching and even if the coordinates assigned via x and y are in *general position*, that is, no two vertices lie on a line with a direction from S. The $\mathcal{NP}$-hardness of Bi-Monotonicity for perfect matchings follows as a simple corollary. Our results are obtained via a reduction from Ordered Level Planarity.

Lemma 2.1. *Let $S \subset \mathbb{Q}^2$ with $|S| \geq 4$ be a finite set of directions which is symmetric with respect to the origin. Ordered Level Planarity with maximum degree $\Delta = 2$ and level-width $\lambda = 2$ reduces to Geodesic Planarity such that the resulting instances are in general position and consist of a perfect matching and direction set S. The reduction can be carried out using a linear number of arithmetic operations.*

Proof. We first prove our claim for the classical case that S contains exactly the four horizontal and vertical directions. Afterwards, we discuss the necessary adaptations for the general case. Our reduction is carried out in two steps. Let $\mathcal{G}_o = (G_o = (V, E), \gamma, \chi)$ be an Ordered Level Planarity instance with maximum degree $\Delta = 2$ and level-width $\lambda = 2$. In Step (i) we turn $\mathcal{G}_o$ into an equivalent Geodesic Planarity instance $\mathcal{G}'_g = (G_o, x', \gamma, S)$. In Step (ii) we transform $\mathcal{G}'_g$ into an equivalent Geodesic Planarity instance $\mathcal{G}_g = (G_g, x, y, S)$ where G_g is a perfect matching and the vertex positions assigned via x and y are in general position.

Step (i): In order to transform $\mathcal{G}_o$ into $\mathcal{G}'_g$, we apply a horizontal shearing transformation to the vertex positions specified by χ and γ. More precisely, for every $v \in V$ we define $x'(v) = \chi(v) + 2\gamma(v)$, see Figures 3(a) and 3(b). Clearly, every geodesic drawing of $\mathcal{G}'_g$ can be turned into an ordered level planar drawing of $\mathcal{G}_o$. On the other hand, consider an ordered level planar drawing Γ_o of $\mathcal{G}_o$. Without loss of generality, we can assume that in Γ_o all edges are realized as polygonal paths in

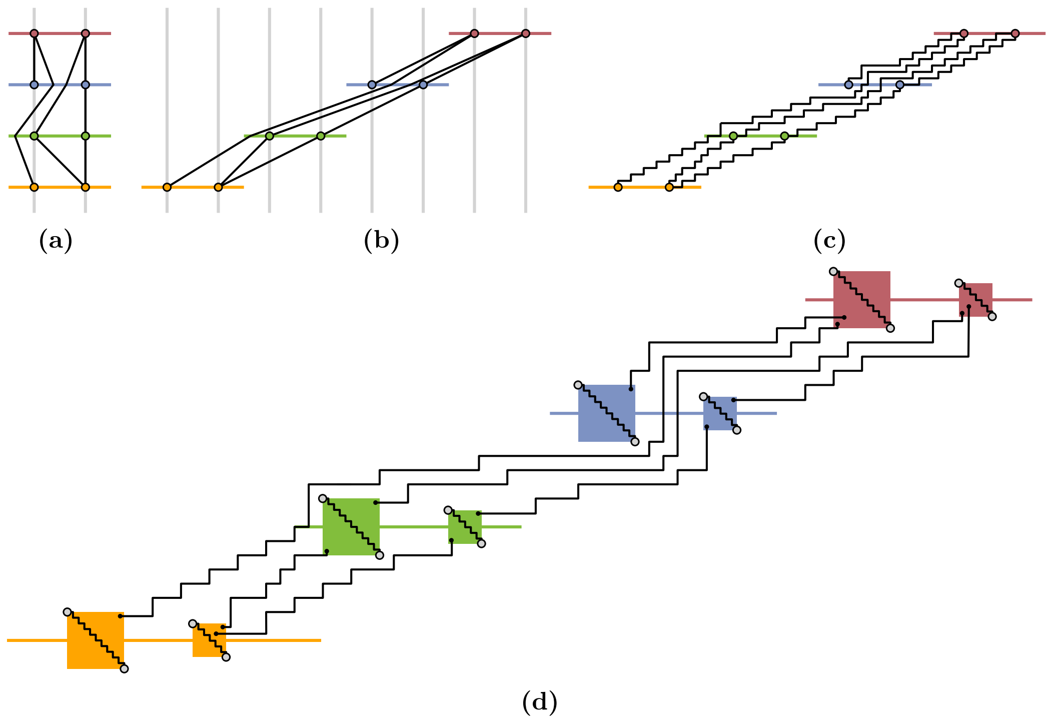


Fig. 3. **(a)**, **(b)** and **(c)**: Illustrations of Step (i). **(d)**: Illustration of Step (ii).

which bend points occur only on the horizontal lines L_i through the levels V_i where $0 \le i \le h$. Further, since $\chi(V) \subseteq \{0,1\}$ we may assume that all bend points have x-coordinates in the open interval $(-1/2, 3/2)$. We shear Γ_o by translating the bend points and vertices of level V_i by $2i$ units to the right for $0 \le i \le h$, see Figure 3(b). In the resulting drawing Γ_o', the vertex positions match those of $\mathcal{G}_g'$. Furthermore, all edge-segments have a positive slope. Thus, since the maximum degree is $\Delta = 2$ we can replace all edge-segments with L_1-geodesic rectilinear paths that closely trace the segments and we obtain a geodesic drawing Γ_g' of $\mathcal{G}_g'$, see Figure 3(c).

Step (ii): In order to turn $\mathcal{G}_g' = (G_o = (V,E), x', y, S)$ into the equivalent instance $\mathcal{G}_g = (G_g, x, y, S)$, we transform G_o into a perfect matching. To this end, we split each vertex $v \in V$ by replacing it with a small gadget that fits inside a square r_v centered on the position $p_v = (x'(v), y(v))$ of v, see Figure 3(d). We call r_v the *square* of v and use $p_v^{\mathrm{tr}}, p_v^{\mathrm{tl}}, p_v^{\mathrm{br}}$ and p_v^{bl} to denote the top-right, top-left, bottom-right and bottom-left corner of r_v, respectively. We use two different sizes to ensure general position. The size of the gadget square is $1/4 \times 1/4$ if $\chi(v) = 0$ and it is $1/8 \times 1/8$ if $\chi(v) = 1$. The gadget contains a degree-1 vertex for every edge incident to v. In the following we explain the gadget construction in detail. For an illustration, see Figure 4(a). Let $\{v, u\}$ be an edge incident to v. We create an edge $\{v_1, u\}$ where v_1 is a new vertex which is placed at $p_v^{\mathrm{tr}} - (1/48, 1/48)$ if u is located to the top-right of v and it is placed at $p_v^{\mathrm{bl}} + (1/48, 1/48)$ if u is located to the bottom-left of v. Similarly, if v is incident to a second edge $\{v, u'\}$, we create an edge $\{v_2, u'\}$ where v_2 is placed at $p_v^{\mathrm{tr}} - (1/24, 1/24)$ or $p_v^{\mathrm{bl}} + (1/24, 1/24)$ depending on the position of u'. We refer to v_1 and v_2 as the *gadget vertices* of v and its square r_v. Finally, we create a *blocking* edge $\{v_{\mathrm{tl}}, v_{\mathrm{br}}\}$ where v_{tl} is placed

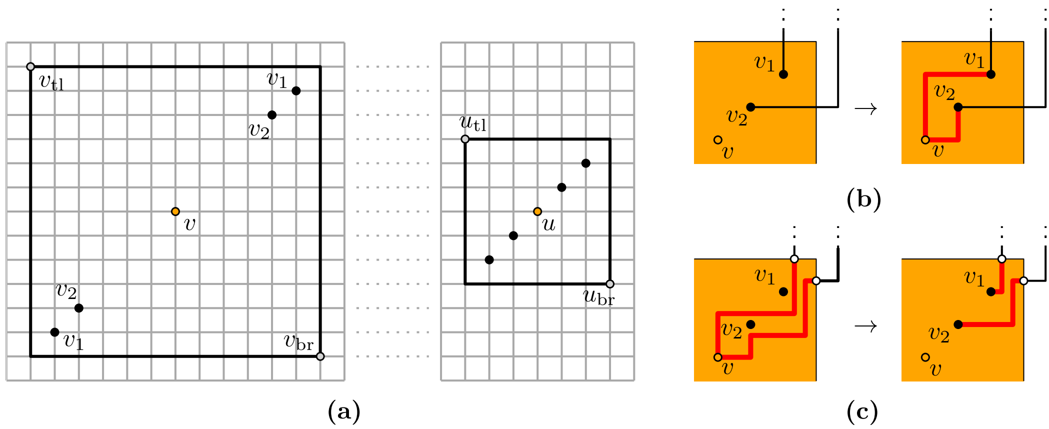


Fig. 4. **(a)** The two gadget squares of each level. Grid cells have size $1/48 \times 1/48$. **(b)** Turning a drawing of $\mathcal{G}_g$ into a drawing of $\mathcal{G}'_g$; **(c)** and vice versa.

at p_v^{tl} and v_{br} is placed at p_v^{br}. All the assigned coordinates are distinct in both components, and hence the points are in general position. The construction can be carried out in linear time.

Assume that $\mathcal{G}_g$ has a geodesic drawing Γ_g. By construction, for each blocking edge, one of its vertices is located to the top-*left* of the other. On the other hand, for each non-blocking edge, one of its vertices is located to the top-*right* of the other. As a result, a non-blocking edge $e = \{v, u\}$ cannot pass through any gadget square r_w where $w \notin \{v, u\}$, since otherwise e would have to cross the blocking edge of r_w. Accordingly, it is straightforward to obtain a geodesic drawing of Γ'_g: We remove the blocking edges, reinsert the vertices of V according to the mappings x' and y and connect them to the gadget vertices of their respective squares in a geodesic fashion. This can always be done without crossings. Figure 4(b) shows one possibility. If the edge from v_2 passes to the left of v_1, we may have to choose a reflected version. Finally, we remove the vertices v_1 and v_2, which now act as subdivision vertices.

On the other hand, let Γ'_g be a geodesic planar drawing of $\mathcal{G}'_g$. Without loss of generality, we can assume that each edge $\{u, v\}$ intersects only the squares of u and v. Furthermore, for each $v \in V$ we can assume that its incident edges intersect the boundary of r_v only to the top-right of $p_v^{\mathrm{tr}} - (1/48, 1/48)$ or to the bottom-left of $p_v^{\mathrm{bl}} + (1/48, 1/48)$, see Figure 4(c). Thus, we can simply replace the parts of the edges *inside* the gadget squares by connections to the gadget vertices v_1 and v_2 in a geodesic fashion, see Figure 4(c).

The general case. It remains to discuss the adaptations for the case that S is an arbitrary set of directions which is symmetric with respect to the origin. By applying a linear transformation we can assume without loss of generality that $(1, 0)$ and $(0, 1)$ are adjacent directions in S. Accordingly, all the remaining directions point into the top-left or the bottom-right quadrant. Further, by vertical scaling we can assume that no direction is parallel to $(1, -1)$. Observe that if we do not insist on a coordinate assignment in general position, the reduction for the restricted case discussed above is already sufficient.

In order to guarantee general position, we have to avoid *conflicting* vertices, i.e. distinct vertices whose positions lie on a common line with a direction from S. This requires some simple but somewhat technical modifications of our construction.

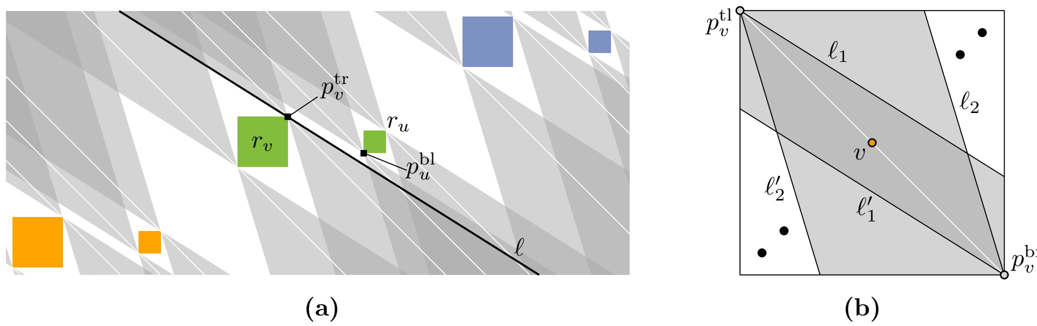


Fig. 5. Modifications **(a)** and **(b)** for the general case.

Let s_1 be the flattest slope of any direction in $S \setminus \{(1,0),(0,1)\}$, i.e. the slope with the smallest absolute value (note that all the slopes are negative). Further, let s_2 be the steepest slope of any direction in $S \setminus \{(1,0),(0,1)\}$, i.e. the slope with the largest absolute value.

Assume that c', d' are conflicting vertices such that c' belongs to the gadget square r_c of $c \in V$, and d' belongs to the gadget square r_d of $d \in V$. Consider Figure 5(a). Since no direction of S points to the top-right or bottom-left quadrant, $\gamma(c) = \gamma(d)$. It is possible that $c = d$.

In order to guarantee general position, we apply the following two modifications.

Modification (a). We first cover the case $c \neq d$, that is, we show how to avoid conflicts between two vertices c', d' which belong to *distinct* squares of the same level. To this end, we increase the horizontal distance between each pair of successive squares in the ordering in which the squares appear along the x-axis without changing said ordering. More precisely, instead of using the coordinates $(2i, i)$ and $(2i+1, i)$ for the centers of the two squares r_v and r_u of level i, we use the positions $(2ki, i)$ and $(2ki+k, i)$ where $k \geq 1$ is chosen large enough that p_u^{bl} is above the line ℓ with slope s_1 through p_v^{tr}, see Figure 5(a).

Modification (b). It remains to cover the case $c = d$, i.e. to avoid conflicts between vertices c', d' which belong to the same gadget square r_v. To this end, we modify the placement of the gadget vertices inside the gadget squares as follows. We change the offset to the gadget square corners from $\pm(1/48)$ and $\pm(1/24)$ to $\pm(z/48)$ and $\pm(z/24)$ where $0 < z < 1$ is chosen small enough such that the gadget vertices are placed above the line ℓ_1 with slope s_1 through p_v^{tl}, and above the line ℓ_2 with slope s_2 through p_v^{br}; or below the line ℓ_1' with slope s_1 through p_v^{br}, and below the line ℓ_2' with slope s_2 through p_v^{tl}; see the white regions in Figure 5(b). □

The bit size of the numbers involved in the calculations of our reduction is linearly bounded in the bit size of the directions of S. Together with Theorem 1.1 we obtain the proof of Theorem 1.2.

Theorem 1.2. *Geodesic Planarity is $\mathcal{NP}$-hard for any set of directions S with $|S| \geq 4$ even for perfect matchings in general position.*

The instances generated by Lemma 2.1 are in general position. In particular, this means that the mappings x and y are injective. We obtain an immediate reduction to Bi-Monotonicity. The correctness follows from the fact that every L_1-geodesic rectilinear path can be transformed into a bi-monotone curve and vice versa. Thus, we obtain Theorem 1.3.

Theorem 1.3. *Bi-Monotonicity is $\mathcal{NP}$-hard even for perfect matchings.*

By combining Lemma 2.1 and the remarks in the previous paragraph, we obtain a reduction from Ordered Level Planarity to Bi-Monotonicity. However, the intermediate reduction via Manhattan Geodesic Planarity requires the original Ordered Level Planarity instance to have a maximum out-degree of $\Delta^+ \leq 2$ and a maximum in-degree of $\Delta^- \leq 2$ (otherwise, our reduction would produce Manhattan Geodesic Planarity instances with vertices that have more than two neighbors in the same quadrant; these instances are never realizable, see Section 1.4). In Section 5, we require a reduction that accepts more general instances of Ordered Level Planarity. For this reason, we state the following direct (and, in fact, much simpler) reduction from Ordered Level Planarity to Bi-Monotonicity.

Theorem 1.4. *Ordered Level Planarity reduces to Bi-Monotonicity in linear time. The reduction can be carried out such that the input graph is identical to the output graph, that is, only the coordinates are modified.*

Proof. Let $\mathcal{G} = (G = (V, E), \gamma, \chi)$ be an ordered level graph with level-width λ and height h. We create an instance of Bi-Monotonicty as follows. The graph G remains unchanged. The new vertex-coordinates are obtained by applying the following linear function f to the assignment given by χ and γ. The function f is a linear deformation of the plane which scales the original coordinates and rotates them by 45°, see Figure 6.

$$f(x, y) := (f_1(x, y), f_2(x, y)) := \Big((\lambda + 1)y + x,\ (\lambda + 1)y - x\Big)$$

We define a coordinate assignment (x', y') with $(x'(v), y'(v)) := f(\chi(v), \gamma(v))$ for each vertex $v \in V$. The resulting Bi-Monotonicity instance is $\mathcal{G}' = (G, x', y')$ with $x'(v) = (\lambda + 1)\gamma(v) + \chi(v)$ and $y'(v) = (\lambda + 1)\gamma(v) - \chi(v)$.

Recall that L_i denotes the horizontal line with y-coordinate i, which passes through all the vertices of level V_i. We use $S_i \subset L_i$ to denote the *open* line segment between the points $(-1, i)$ and (λ, i). The correctness of our reduction relies on the following property:

Proposition 2.2. *Let $p_i \in f(S_i)$ and $p_{i+1} \in f(S_{i+1})$ for some $0 \leq i < \lambda$. Then $p_i < p_{i+1}$, componentwise.*

The correctness of Proposition 2.2 follows from the simple fact that for $(j, i) = f^{-1}(p_i)$ and $(j', i + 1) = f^{-1}(p_{i+1})$ we have:

$$\begin{aligned} p_i &= f(j, i) \\ &< \Big((\lambda + 1)i + \lambda,\ (\lambda + 1)i + 1\Big) \\ &= \Big((\lambda + 1)(i + 1) - 1,\ (\lambda + 1)(i + 1) - \lambda\Big) \\ &< f(j', i + 1) \\ &= p_{i+1} \end{aligned}$$

Let Γ be an ordered level planar drawing of $\mathcal{G}$. Without loss of generality, we can assume that in Γ all edges are realized as polygonal paths in which bend-points occur only on the horizontal segments S_i, see Figure 6(a). Applying f to all the bend-points yields a drawing $f(\Gamma)$ of $\mathcal{G}'$, see Figure 6(b). Since f is linear, $f(\Gamma)$ is plane. By Proposition 2.2, every edge in $f(\Gamma)$ is realized as a polygonal path whose segments have positive slopes. Therefore $f(\Gamma)$ is bi-monotone.

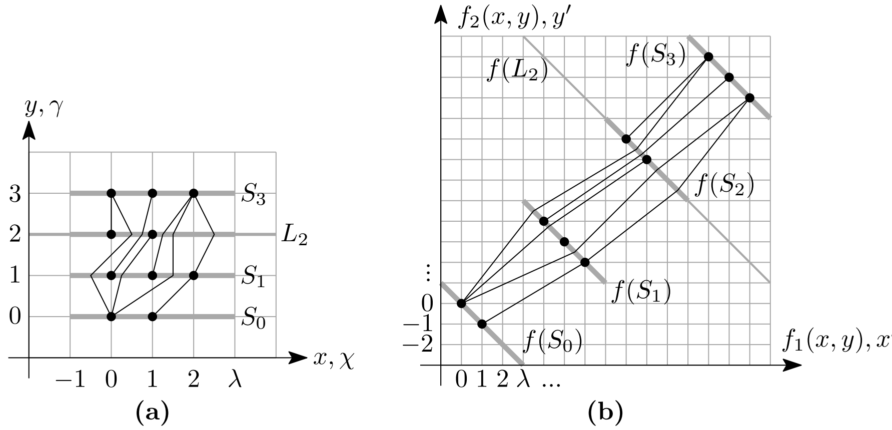


Fig. 6. **(a)** An ordered level planar drawing of $\mathcal{G}$; **(b)** and the corresponding bi-monotone drawing of $\mathcal{G}'$.

On the other hand, let Γ' be a planar bi-monotone drawing of $\mathcal{G}'$. The lines $f(L_i) \supset f(S_i)$ have a negative slope (of -1); and by Proposition 2.2, every edge is realized as a curve that is simultaneously increasing in the x- and y-directions. Therefore, every edge may intersect each line $f(L_i)$ at most once. More precisely, an edge (v_j, v_k) with $v_j \in V_j$, $v_k \in V_k$ and $j < k$ crosses each of the consecutive lines $f(L_{j+1}), \ldots, f(L_{k-1})$ exactly once. Further, all vertices of level V_i have been mapped to $f(S_i) \subset f(L_i)$. Thus, we can leave the intersection of each edge with each line $f(L_i)$ fixed and replace the intermediate pieces by line-segments. This does not introduce any crossings and turns all edges into x- and y-montone polygonal paths in which bend-points occur only on the lines $f(L_i)$, see Figure 6(b). Applying f^{-1} yields an ordered level planar drawing $f^{-1}(\Gamma')$ of $\mathcal{G}$, see Figure 6(a). □

3 VARIATIONS OF LEVEL PLANARITY

In this section we explore the connection between Ordered Level Planarity and other variants of Level Planarity. We prove the following theorem.

Theorem 1.5. *Ordered Level Planarity reduces in linear time to Constrained Level Planarity and T-Level Planarity, and in quadratic time to Clustered Level Planarity.*

The reduction to Constrained Level Planarity is immediate, which together with Theorem 1.1 also yields:

Theorem 1.6. *Constrained Level Planarity is $\mathcal{NP}$-hard even for acyclic level graphs with maximum degree $\Delta = 2$ and level-width $\lambda = 2$ and prescribed* total *orderings.*

For the other two reductions, we restrict our attention to ordered level graphs with level-width $\lambda = 2$. As we will see in Section 4, this restriction is no loss of generality (Lemma 4.2).

We first reduce to T-Level Planarity:

Lemma 3.1. *Ordered Level Planarity with maximum degree Δ and level-width $\lambda = 2$ reduces in linear time to T-Level Planarity with maximum degree $\Delta' = \max(\Delta, 2)$ and level-width $\lambda' = 4$.*

Proof. Let $\mathcal{G} = (G = (V, E), \gamma, \chi)$ be an ordered level graph with maximum degree Δ and level-width $\lambda = 2$. We augment each level V_i with $|V_i| = 1$ by adding an isolated dummy vertex v with $\gamma(v) = i$ and $\chi(v) = 1$ in order to avoid having to treat special cases. Thus, each level V_i has a vertex v_i^0 with $\chi(v_i^0) = 0$ and a vertex v_i^1 with $\chi(v_i^1) = 1$. The following steps are illustrated in Figure 7a. For each level V_i we create two new vertices v_i^l and v_i^r. We add edges (v_i^l, v_{i+1}^l) and (v_i^r, v_{i+1}^r) for $i = 0, \dots, h-1$, where h is the height of $\mathcal{G}$. Hence, we obtain a path p_l from v_0^l to v_h^l and a path p_r from v_0^r to v_h^r. The root r_i of each tree T_i has two children u_i^l and u_i^r. The two children of u_i^l are v_i^l and v_i^0. The two children of u_i^r are v_i^r and v_i^1. Let $\mathcal{G}'$ denote the resulting T-level graph. The construction of $\mathcal{G}'$ can be carried out in linear time.

Clearly, an ordered level planar drawing Γ of $\mathcal{G}$ can be augmented to a T-level planar drawing of $\mathcal{G}'$ by drawing p_l to the left of Γ and by drawing p_r to the right of Γ. On the other hand, let Γ' be a T-level-planar drawing of $\mathcal{G}'$. We can assume without loss of generality that all vertices are placed on vertical lines with x-coordinates -1, 0, 1 or 2. The paths p_l and p_r are vertex-disjoint and drawn without crossing. Thus, p_l is drawn either to the left or to the right of p_r. By reflecting horizontally at the line $x = 1/2$ we can assume without loss of generality that p_l is drawn to the left of p_r. Consequently, for each level V_i the vertex v_i^0 has to be drawn to the left of the vertex v_i^1 since v_i^l and v_i^0 are the children of u_i^l and since v_i^r and v_i^1 are the children of u_i^r. Therefore, the subdrawing of G or its mirror image is an ordered level planar drawing of $\mathcal{G}$. □

Together with Theorem 1.1 this shows the $\mathcal{NP}$-hardness of T-Level Planarity for instances with maximum degree $\Delta = 2$ and level-width $\lambda = 4$. However, a stronger statement was already given by Angelini et al. [2], who show $\mathcal{NP}$-hardness for instances with $\Delta = 2$ and $\lambda = 3$.

We proceed with a reduction to Clustered Level Planarity.

Lemma 3.2. *Ordered Level Planarity with maximum degree Δ and level-width $\lambda = 2$ reduces in quadratic time to Clustered Level Planarity with maximum degree $\Delta' = \max(\Delta, 2)$, level-width $\lambda' = 2$, and a clustering hierarchy that partitions the vertices into only two non-trivial clusters.*

Proof. Let $\mathcal{G} = (G = (V, E), \gamma, \chi)$ be an ordered level graph with maximum degree Δ and level-width $\lambda = 2$. As in the previous proof, we augment each level V_i with $|V_i| = 1$ by adding an isolated dummy vertex v with $\gamma(v) = i$ and $\chi(v) = 1$. Thus, each level V_i has a vertex v_i^0 with $\chi(v_i^0) = 0$ and a vertex v_i^1 with $\chi(v_i^1) = 1$. In addition to the trivial cluster that contains all vertices, we create two clusters $c_0 = \{v_0^0, \dots, v_h^0\}$ and $c_1 = \{v_0^1, \dots, v_h^1\}$, where h is the height of $\mathcal{G}$. Now we see the close correspondence between clustered level planar drawings and ordered level planar drawings: The two clusters pass through every level, their boundaries are not allowed to intersect, and they cannot be nested. Thus, by reflecting horizontally if necessary, we can assume without loss of generality that c_0 intersects each level to the left of c_1 as depicted in Figure 7c. Consequently, on each level V_i the vertex $v_i^0 \in c_0$ is placed to the left of $v_i^1 \in c_1$, just as in an ordered level planar drawing.

In order to make the reduction work, we have to subdivide each edge several times. Otherwise, an edge might be forced to cross a cluster boundary more than once: Consider an edge $e = (u, v)$ with $u, v \in c_0$ that has to pass the level of some vertex $b \in c_1$ with $\gamma(u) < \gamma(b) < \gamma(v)$ to the right of b, see Figure 7b. In this situation, e must cross the right boundary r_0 of c_0 at least twice, as r_0 has to be drawn to the right of $u, v \in c_0$, and to the left of $b \in c_1$. This example can be blown up to enforce arbitrarily many crossings between e and r_0.

In order to avoid this situation, we subdivide the edges of $\mathcal{G}$ as follows. Each edge from some level i to some level $j > i$ is transformed into a path of $2(j-i)+1$ edges whose inner vertices alternate between the clusters c_1 and c_0. More precisely, for each pair of consecutive levels V_i and V_{i+1} we add two new subdivision vertices on each edge $e = (u, v) \in E$ with $\gamma(u) \le i$ and $\gamma(v) \ge i + 1$. The lower one of the resulting subdivision vertices for e is added to c_1, the upper one is added to c_0. We

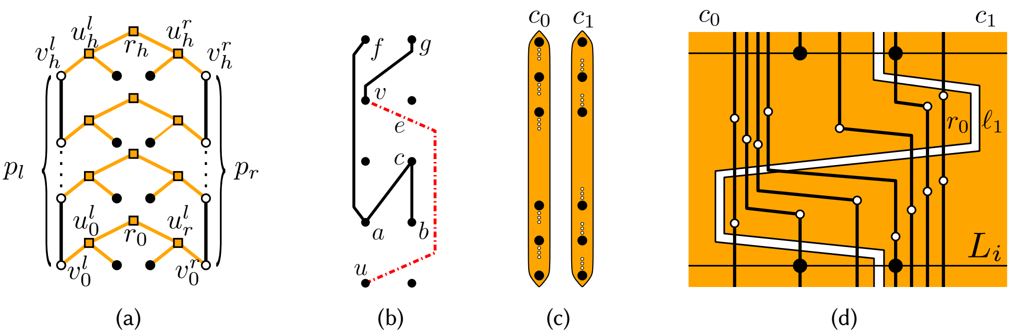


Fig. 7. **(a)** Reduction from Ordered Level Planarity to T-level Planarity. The square vertices illustrate each level's tree. **(b)** In an ordered level planar drawing, the edge $e = (u, v)$ has to pass the level of b to the right of b: Due to the edge (v, g), the edge (a, f) passes to the left of v. As a consequence, e cannot pass the level of b to the left of a. Further, due to (a, c) and (b, c), it can also not pass between a and b. **(c-d)** Reduction from Ordered Level Planarity to Clustered Level Planarity. Big black vertices are the vertices of the Ordered Level Planarity instance. The small vertices are subdivision vertices. **(c)** Schematic view of the entire clustered level graph. **(d)** The clustering boundaries can be drawn such that they cross each subdivision edge at most once.

place each of the subdivision vertices that was added to c_1 on a new separate level between the levels V_i and V_{i+1}. The relative order of these new levels is arbitrary. Above these new levels but below V_{i+1} we place all the subdivision vertices added to c_0, again each on a new separate level, see Figures 7c–7d.

Let $\mathcal{G}^s = (G^s, \gamma^s, \chi^s)$ denote the ordered level graph resulting from applying the subdivision to $\mathcal{G}$. The output of our reduction is the clustered level graph $\mathcal{G}^{\mathrm{cl}} = (G^s, \gamma^s, T)$ where T is the described hierarchy, with the clusters c_0 and c_1. Since edges may stretch over a linear number of levels, the size of G^s can be quadratic in the size of G and, therefore, the construction of $\mathcal{G}^{\mathrm{cl}}$ might require quadratic time.

Correctness. The subdivision does not affect the realizability of $\mathcal{G}$ as an ordered level planar drawing, since every subdivision vertex in $\mathcal{G}^s$ is the singleton vertex of some new level. Therefore, to prove correctness, it suffices to argue that $\mathcal{G}^{\mathrm{cl}}$ is realizable as a clustered level planar drawing if and only if $\mathcal{G}^s$ is realizable as an ordered level planar drawing.

For the easy direction, let Γ^{cl} be a clustered level planar drawing of $\mathcal{G}^{\mathrm{cl}}$. As discussed above, we may assume that c_0 is drawn to the left of c_1. Further, we may assume without loss of generality that all vertices are placed on vertical lines with x-coordinates 0 and 1, and moreover, all subdivision vertices, being singleton vertices of their levels, are placed on $x = 0$. Recall that each vertex v of the original graph is contained in c_0 if $\chi(v) = 0$; and it is contained in c_1 if $\chi(v) = 1$. Thus, by the above assumptions, $v \in V$ is placed on $x = 0$ if $\chi(v) = 0$; and it is placed on $x = 1$ if $\chi(v) = 1$. Therefore, the drawing Γ^{cl} (without the cluster boundaries) is an ordered level planar drawing of $\mathcal{G}^s$.

For the other direction, let Γ be an ordered level planar drawing of the ordered level graph $\mathcal{G}^s$. We create a clustered level planar drawing of $\mathcal{G}^{\mathrm{cl}}$ by adding the cluster boundaries of c_0 and c_1 to Γ. The left boundary ℓ_0 of c_0 is drawn as a vertical line segment to the left of Γ. Analogously, the right boundary r_1 of c_1 is drawn as a vertical line segment to the right of Γ.

It remains to draw the right boundary r_0 of c_0 and the left boundary ℓ_1 of c_1. We draw them from bottom to top. We keep them close together, and they will always cross the same edge in direct succession, see Figure 7d. Assume inductively that r_0 and ℓ_1 have already been drawn in the closed half-plane H_i below the line L_i through the vertices V_i of $\mathcal{G}$, and this subdrawing violates none of the conditions from the definition of a clustered level planar drawing. In particular, r_0 and ℓ_1 are realized as non-crossing y-monotone curves with all vertices of c_0 to the left of r_0, and with all vertices of c_1 to the right of ℓ_1. Moreover, no edge is intersected more than once by any of r_0 or ℓ_1. Further, let E_i be the set of edges of G^s that are intersected by L_i including the edges having their lower endpoint on L_i, but without the edges having their upper endpoint on L_i. We maintain the following two additional inductive assumptions: (a) L_i intersects the edges in E_i and the boundaries r_0 and ℓ_1 in the following left-to-right order (see Figure 7d): (1) all edges $E_\ell \subseteq E_i$ that intersect L_i to the left of v_i^1; (2) the boundary r_0; (3) the boundary ℓ_1; and (4) the remaining edges $E_r = E_i \setminus E_\ell$, i.e. the edges incident to v_i^1, or passing v_i^1 to its right. (b) No edge of E_ℓ has already been crossed by r_0 or ℓ_1 below L_i. Note that these conditions are easily met for $i = 0$.

We describe how the partial drawings of r_0 and ℓ_1 are extended upwards from L_i. For an illustration, see Figure 7d. Each edge in E_i is part of a path that has two subdivision vertices between L_i and L_{i+1}. The lower of these vertices belongs to c_1, and the upper one belongs to c_0. We draw r_0 and ℓ_1 in a very schematic and simple way. First we cross all edges in E_ℓ from right to left. By assumption (b), this is permitted. We then pass to the left of all the lower subdivision vertices, ensuring that they lie within the cluster boundaries of c_1. We then cross all edges between their two subdivision vertices from left to right, and pass to the right of all the subdivision vertices in c_0. Finally, we cross from right to left all edges which pass L_{i+1} to the right of v_{i+1}^1, and those whose upper endpoint is v_{i+1}^1. It is easy to check that the inductive assumptions hold again for L_{i+1}. Thus, we may iterate this procedure to obtain a clustered level planar drawing of $\mathcal{G}^{\mathrm{cl}}$. □

Together with Theorem 1.1 we obtain the following.

Theorem 1.7. *Clustered Level Planarity is $\mathcal{NP}$-hard even for acyclic clustered level graphs with maximum degree $\Delta = 2$, level-width $\lambda = 2$ and a flat cluster hierarchy that partitions the vertices into two non-trivial clusters.*

The previous $\mathcal{NP}$-hardness result by Angelini et al. [2] holds for instances with $\Delta = 2$ and $\lambda = 3$. Their cluster hierarchies have linear depths. The authors pose the complexity of Clustered Level Planarity for instances with flat cluster hierarchies as an open problem. Theorem 1.7 gives an answer to this question and improves the previous result by Angelini et al.

4 ORDERED LEVEL PLANARITY

In this section we study Ordered Level Planarity. For the $\mathcal{NP}$-hardness proof, we reduce from the 3-Satisfiability variant described in this paragraph. A *monotone* 3-Satisfiability formula is a Boolean 3-Satisfiability formula in which each clause is either *positive* or *negative*, that is, each clause contains either exclusively positive or exclusively negative literals, respectively. A *planar* 3SAT formula $\varphi = (\mathcal{U}, C)$ is a Boolean 3-Satisfiability formula with a set $\mathcal{U}$ of variables and a set C of clauses such that its *variable–clause graph* $G_\varphi = (\mathcal{U} \uplus C, E)$ is planar. The graph G_φ is bipartite, i.e. every edge in E is incident to a *clause* vertex from C and a *variable* vertex from $\mathcal{U}$. Furthermore, edge $\{c, u\} \in E$ if and only if a literal of variable $u \in \mathcal{U}$ occurs in $c \in C$. Planar Monotone 3-Satisfiability is a special case of 3-Satisfiability where we are given a planar and monotone 3-Satisfiability formula φ and a *monotone rectilinear representation* $\mathcal{R}$ of the variable-clause graph of φ. The representation $\mathcal{R}$ is a contact representation on an integer grid in which the variables are represented by horizontal line segments arranged on a common horizontal line ℓ. The clauses are represented by E-shapes turned by 90° such that all positive clauses are placed above ℓ

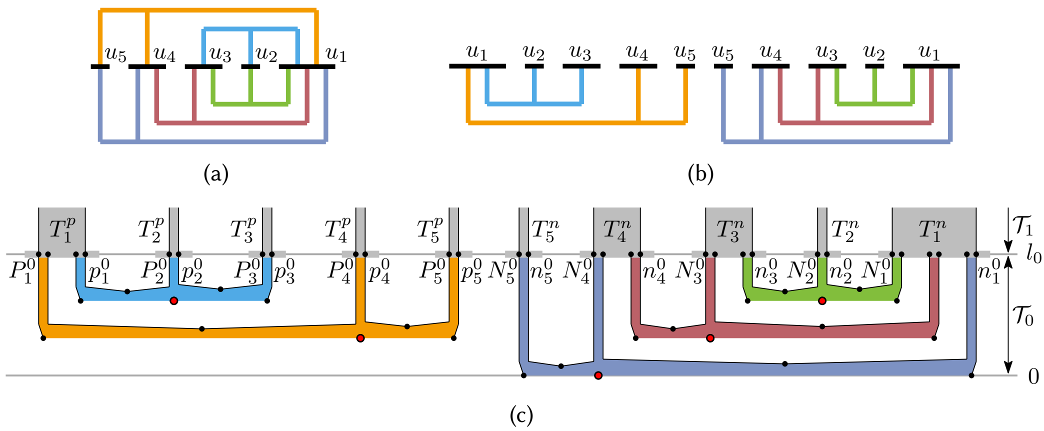


Fig. 8. **(a)** Representation $\mathcal{R}$ of φ with negative clauses $(\overline{u}_1 \vee \overline{u}_4 \vee \overline{u}_5)$, $(\overline{u}_1 \vee \overline{u}_3 \vee \overline{u}_4)$ and $(\overline{u}_1 \vee \overline{u}_2 \vee \overline{u}_3)$ and positive clauses $(u_1 \vee u_4 \vee u_5)$ and $(u_1 \vee u_2 \vee u_3)$ and **(b)** its modified version $\mathcal{R}'$ in Lemma 4.1. **(c)** Tier $\mathcal{T}_0$.

and all negative clauses are placed below ℓ, see Figure 8a. Planar Monotone 3-Satisfiability is $\mathcal{NP}$-complete [8]. We are now equipped to prove the core lemma of this section.

Lemma 4.1. *Planar Monotone 3-Satisfiability reduces in polynomial time to Ordered Level Planarity. The resulting instances have maximum degree $\Delta = 2$ and contain no source or sink with degree Δ on a level V_i with width $\lambda_i > 2$.*

Proof. We perform a polynomial-time reduction from Planar Monotone 3-Satisfiability. Let $\varphi = (\mathcal{U}, C)$ be a planar and monotone 3-Satisfiability formula with clause set $C = \{c_1, \ldots, c_{|C|}\}$. Let G_φ be the variable-clause graph of φ. Let $\mathcal{R}$ be a monotone rectilinear representation of G_φ. We construct an ordered level graph $\mathcal{G} = (G, \gamma, \chi)$ such that $\mathcal{G}$ has an ordered level planar drawing if and only if φ is satisfiable.

Overview. The ordered level graph $\mathcal{G}$ has $l_3 + 1$ levels which are partitioned into four *tiers* $\mathcal{T}_0 = \{0, \ldots, l_0\}, \mathcal{T}_1 = \{l_0{+}1, \ldots, l_1\}, \mathcal{T}_2 = \{l_1{+}1, \ldots, l_2\}$ and $\mathcal{T}_3 = \{l_2{+}1, \ldots, l_3\}$. Each clause $c_i \in C$ is associated with a *clause* edge $e_i = (c_i^s, c_i^t)$ starting with c_i^s in tier $\mathcal{T}_0$ and ending with c_i^t in tier $\mathcal{T}_2$. The clause edges have to be drawn in a system of tunnels that encodes the 3-Satisfiability formula φ. In $\mathcal{T}_0$ the layout of the tunnels corresponds directly to the rectilinear representation $\mathcal{R}$, see Figure 8c. For each E-shape there are three tunnels corresponding to the three literals of the associated clause. The bottom vertex c_i^s of each clause edge e_i is placed such that e_i has to be drawn inside one of the three tunnels of the E-shape corresponding to c_i. This corresponds to the fact that in a satisfying truth assignment every clause has at least one satisfied literal. In tier $\mathcal{T}_1$ we merge all the tunnels corresponding to the same literal. We create variable gadgets that ensure that for each variable u, the edges of clauses containing u can be drawn in the tunnel associated with either the negative or the positive literal of u but not in both. This corresponds to the fact that every variable is set to either true or false. Tiers $\mathcal{T}_2$ and $\mathcal{T}_3$ have a technical purpose.

We proceed by describing the different tiers in detail. Recall that in terms of realizability, Ordered Level Planarity is equivalent to the generalized version where γ and χ map to the reals. For the sake of convenience we will begin by designing $\mathcal{G}$ in this generalized setting. It is easy to transform $\mathcal{G}$ such that it satisfies the standard definition in a polynomial-time post processing step.

Tiers 0 and 2, clause gadgets. Each clause edge $e_i = (c_i^s, c_i^t)$ ends in tier $\mathcal{T}_2$. It is composed of $l_2 - l_1 = |C|$ levels each of which contains precisely one vertex. We assign $\gamma(c_i^t) = l_1 + i$. Recall that for levels with width 1, the assigned x-coordinates are irrelevant. Hence, we set $\chi(c_i^t) = 0$. Observe that the positions of the vertices c_i^t impose no constraints on the order in which the incident edges enter $\mathcal{T}_2$.

Tier $\mathcal{T}_0$ consists of a system of tunnels that resembles the monotone rectilinear representation $\mathcal{R}$ of $G_\varphi = (\mathcal{U} \uplus C, E)$, see Figure 8c. Intuitively it is constructed as follows: We take the top part of $\mathcal{R}$, rotate it by 180° and place it to the left of the bottom part such that the variables' line segments align, see Figure 8b. We call the resulting representation $\mathcal{R}'$. For each E-shape in $\mathcal{R}'$ we create a *clause gadget*, which is a subgraph composed of 11 vertices that are placed on a grid close to the E-shape, see Figure 9. The enlarged vertex at the bottom is the lower vertex c_i^s of the clause edge e_i of the clause c_i corresponding to the E-shape. Without loss of generality we assume the grid to be fine enough such that the resulting ordered level graph can be drawn as in Figure 8c without crossings. Further, we assume that the y-coordinates of every pair of horizontal segments belonging to distinct E-shapes differ by at least 3. This ensures that there are no sources or sinks with degree Δ on levels with width larger than 2.

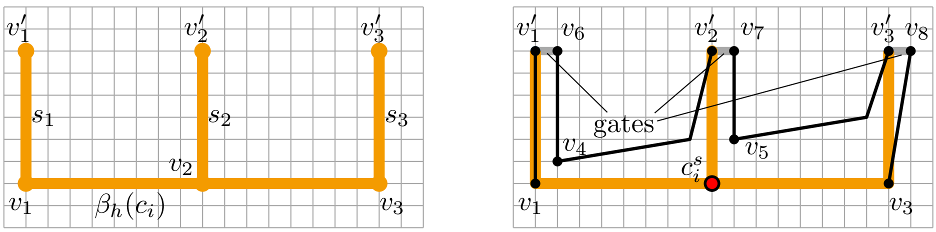
Technical Details. In the following two paragraphs, we describe the construction of the clause gadgets in detail.

For every $i = 1, \dots, |C|$ where c_i is negative, we create its 11-vertex clause gadget as follows, see Figure 9. Let s_1, s_2, s_3 be the three vertical line segments of the E-shape representing c_i in $\mathcal{R}'$ where s_1 is left-most and s_3 right-most. Let v_1, v_2, v_3 be the lower endpoints and v_1', v_2', v_3' be the upper endpoints of s_1, s_2, s_3, respectively. We place the tail c_i^s of the clause edge e_i of c_i at v_2. We create new vertices at $v_1, v_3, v_1', v_2', v_3', v_4 = v_1 + (1, 1), v_5 = v_2 + (1, 2)$ and at v_6, v_7, v_8 which are the lattice points one unit to the right of v_1', v_2', v_3', respectively. To simplify notation, we identify these new vertices with their locations on the grid. We add edges (v_1, v_1'), (v_3, v_8), (v_4, v_6), (v_4, v_2'), (v_5, v_7) and (v_5, v_3') to G.

As stated above, we can assume without loss of generality that the grid is fine enough such that the resulting ordered level graph can be drawn as in Figure 8c without crossing. It suffices to assume that the horizontal and vertical distance between any two segment endpoints of $\mathcal{R}'$ is at least 3 (unless the endpoints lie on a common horizontal or vertical line).

Gates and Tunnels. The clause gadget (without the clause edge) has a *unique* ordered level planar drawing in the sense that for every level V_i the left-to-right sequence of vertices and edges intersected by the horizontal line L_i through V_i is identical in every ordered level planar drawing. This is due to the fact that the order of the top-most vertices v_1', v_6, v_2', v_7, v_3' and v_8 is fixed and every edge of the gadget is incident to precisely one of these vertices. With the same reasoning, it follows that the subgraph G_0 induced by $\mathcal{T}_0$ (without the clause edges) has a unique ordered level planar drawing.

Consider the clause gadget of some clause c_i. We call the line segments $v_1'v_6$, $v_2'v_7$ and $v_3'v_8$ the *gates* of c_i. Note that the clause edge e_i has to intersect one of the gates of c_i. This corresponds to the fact the at least one literal of every clause has to be satisfied. In tier $\mathcal{T}_1$ we bundle all gates that belong to the same literal together by creating two long paths for each literal. These two paths form the *tunnel* of the corresponding literal. All clause edges intersecting a gate of some literal have to be drawn inside the literal's tunnel, see Figure 8c. More precisely, for $j = 1, \dots, |\mathcal{U}|$ we use N_j^0 (n_j^0) to refer to the left-most (right-most) vertex of a negative clause gadget placed on a line segment of $\mathcal{R}'$ representing $u_j \in \mathcal{U}$. The vertices N_j^0 and n_j^0 are the first vertices of the paths forming the *negative* tunnel T_j^n of the negative literal of variable u_j. Analogously, we use P_j^0 (p_j^0) to refer to the left-most (right-most) vertex of a positive clause gadget placed on a line segment of $\mathcal{R}'$ representing u_j. The vertices P_j^0 and p_j^0 are the first vertices of the paths forming the *positive*



(a) (b)

Fig. 9. **(a)** The E-shape and **(b)** the clause gadget of clause c_i.

tunnel T_j^p of the positive literal of variable u_j. If for some j the variable u_j is not contained both in negative and positive clauses, we artificially add two vertices N_j^0 and n_j^0 or P_j^0 and p_j^0 on the corresponding line segments in order to avoid having to treat special cases in the remainder of the construction.

Tiers 1 and 3, variable gadgets. Recall that every clause edge has to pass through a gate that is associated with some literal of the clause, and, thus, every edge is drawn in the tunnel of some literal. We need to ensure that for no variable it is possible to use both the tunnel associated with its positive literal, as well as the tunnel associated with its negative literal simultaneously. To this end, we create a *variable gadget* with vertices in tiers $\mathcal{T}_1$ and T_3 for each variable. The variable gadget of variable u_j is illustrated in Figure 10a. The variable gadgets are nested in the sense that they start in $\mathcal{T}_1$ in the order $u_1, u_2, \dots, u_{|\mathcal{U}|}$, from bottom to top and they end in the reverse order in $\mathcal{T}_3$, see Figure 11. We force each tunnel with index at least j to be drawn between the vertices u_j^a and u_j^b. This is done by subdividing the tunnel edges on this level, see Figure 10b. The *long edge* (u_j^s, u_j^t) has to be drawn to the left or right of u_j^c in $\mathcal{T}_3$. Accordingly, it is drawn to the left of u_j^a or to the right of u_j^b in $\mathcal{T}_1$. Thus, it is drawn either to the right (Figure 10b) of all the tunnels or to the left (Figure 10c) of all the tunnels. As a consequence, the *blocking edge* (u_j^s, u_j^p) is also drawn either to the right or the left of all the tunnels. Together with the edge (u_j^q, u_j^p) it prevents clause edges from being drawn either in the positive tunnel T_j^p or negative tunnel T_j^n of variable u_j which end at level $\gamma(u_j^q)$ because they cannot reach their endpoints in $\mathcal{T}_2$ without crossings. We say T_j^p or T_j^n are *blocked* respectively.

Technical Details. In the following two paragraphs, we describe the construction of the variable gadgets in detail.

Tier $\mathcal{T}_3$ has $l_3 - l_2 = 2 \cdot |\mathcal{U}|$ layers each of which contains precisely one vertex. We refer to the vertex in layer $(l_3 - 2j + 1)$ as u_j^t and to the vertex in layer $(l_3 - 2j)$ as u_j^c for $j = 1, \dots, |\mathcal{U}|$. Tier $\mathcal{T}_1$ has $l_1 - l_0 = 4 \cdot |\mathcal{U}|$ levels. In each of the levels $(l_0 + 4j - 3)$, $(l_0 + 4j - 1)$ and $(l_0 + 4j)$ where $j = 1, \dots, |\mathcal{U}|$ we create one vertex. These vertices are called u_j^s, u_j^q and u_j^p respectively. In level $(l_0 + 4j - 2)$ we create two vertices u_j^a and u_j^b in this order. We add the edges (u_j^s, u_j^t), (u_j^s, u_j^p), (u_j^a, u_j^c), (u_j^b, u_j^c) and (u_j^q, u_j^p).

Finally, for $j = 1, \dots, |\mathcal{U}|$ we do the following, see Figure 10b or Figure 10c. In level $(l_0 + 4j - 2)$ we create vertices $P_j^j, p_j^j, \dots, P_{|\mathcal{U}|}^j, p_{|\mathcal{U}|}^j, N_{|\mathcal{U}|}^j, n_{|\mathcal{U}|}^j, \dots, N_j^j, n_j^j$ and add them in this order between u_j^a and u_j^b. In level $(l_0 + 4j - 1)$ we create vertices P_j^{j+1} and p_j^{j+1} in this order before u_j^q and we create vertices N_j^{j+1} and n_j^{j+1} in this order after u_j^q. We create edges realizing the paths $t_j^P = (P_j^0, \dots, P_j^{j+1})$,

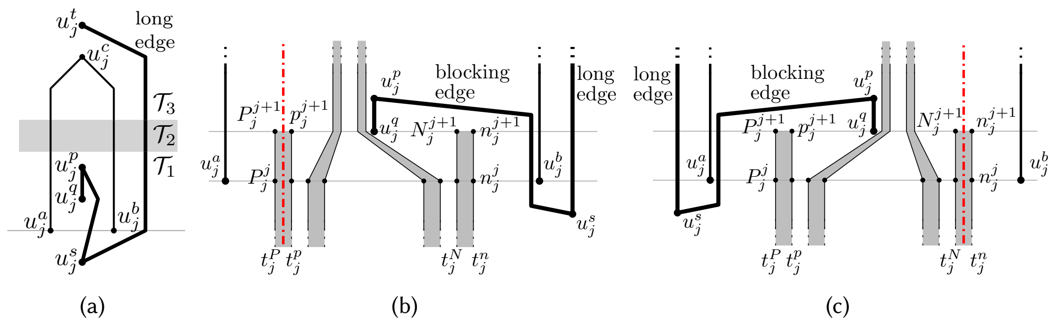


Fig. 10. **(a)** The variable gadget of u_j in **(b)** positive and **(c)** negative state. For the sake of visual clarity, these figures make use of the relaxed but equivalent version of Ordered Level Planarity which only requires that the vertices of each level appear according to the total ordering corresponding to χ, cf. Section 1.3. In particular, a vertex v of a level V_i with width $\lambda_i = 1$ may appear *anywhere* on the horizontal line L_i. The dash-dotted edges are clause edges.

$t^p_j = (p^0_j, \dots, p^{j+1}_j)$, $t^N_j = (N^0_j, \dots, N^{j+1}_j)$ and $t^n_j = (n^0_j, \dots, n^{j+1}_j)$. The pair of paths $T^p_j = (t^P_j, t^p_j)$ is the positive tunnel of variable u_j. The pair of paths $T^n_j = (t^N_j, t^n_j)$ is the negative tunnel of variable u_j. If an edge e is drawn in the region between the two paths of a tunnel T, we say it is drawn *in* T.

Runtime and Properties. The construction of the ordered level graph $\mathcal{G}$ can be carried out in polynomial time. Note that its maximum degree is $\Delta = 2$ and that no source or sink with degree Δ is located on a level V_i with width $\lambda_i > 2$.

Correctness. It remains to show that $\mathcal{G}$ has an ordered level planar drawing if and only if φ is satisfiable. Assume that $\mathcal{G}$ has an ordered level planar drawing Γ. We create a satisfying truth assignment for φ. If T^n_j is blocked we set u_j to true, otherwise we set u_j to false for $j \in 1, \dots, |\mathcal{U}|$. Recall that the subgraph G_0 induced by the vertices in tier $\mathcal{T}_0$ has a unique ordered level planar drawing. Consider a clause c_i and let f, g, j be the indices of the variables whose literals are contained in c_i. Clause edge $e_i = (e^s_i, e^t_i)$ has to pass level l_0 through one of the gates of c_i. More precisely, e_i has to be drawn between either N^0_f and n^0_f, N^0_g and n^0_g, or N^0_j and n^0_j if c_i is negative, or between either P^0_f and p^0_f, P^0_g and p^0_g, or P^0_j and p^0_j if c_i is positive, see Figure 8c. First, assume that c_i is negative and assume without loss of generality that e_i traverses l_0 between N^0_j and n^0_j. In this case e_i has to be drawn in T^n_j. Recall that this is only possible if T^n_j is not blocked, which is the case if u_j is false, see Figure 10c. Analogously, if c_i is positive and e_i traverses w.l.o.g. between p^P_j and p^p_j, then u_j is true, Figure 10b. Thus, we have established that one literal of each clause in C evaluates to true for our truth assignment and, hence, formula φ is satisfiable.

Now assume that φ is satisfiable and consider a satisfying truth assignment. We create an ordered level planar drawing Γ of $\mathcal{G}$. It is clear how to create the unique subdrawing of G_0. The variable gadgets are drawn in a nested fashion, see Figure 11. For $j = 1, \dots, |\mathcal{U}| - 1$ we draw edge (u^a_j, u^c_j) to the left of u^a_{j+1} and u^s_{j+1} and edge (u^b_j, u^c_j) to the right of u^b_{j+1} and u^s_{j+1}. In other words, the pair $((u^a_j, u^c_j), (u^b_j, u^c_j))$ is drawn between all such pairs with index smaller than j. Recall that the vertices u^a_j, u^b_j, u^s_j, u^p_j and u^q_j are located on higher levels than the according vertices of variables with index smaller than j and that u^t_j and u^c_j are located on lower levels than the according vertices of variables with index smaller than j.

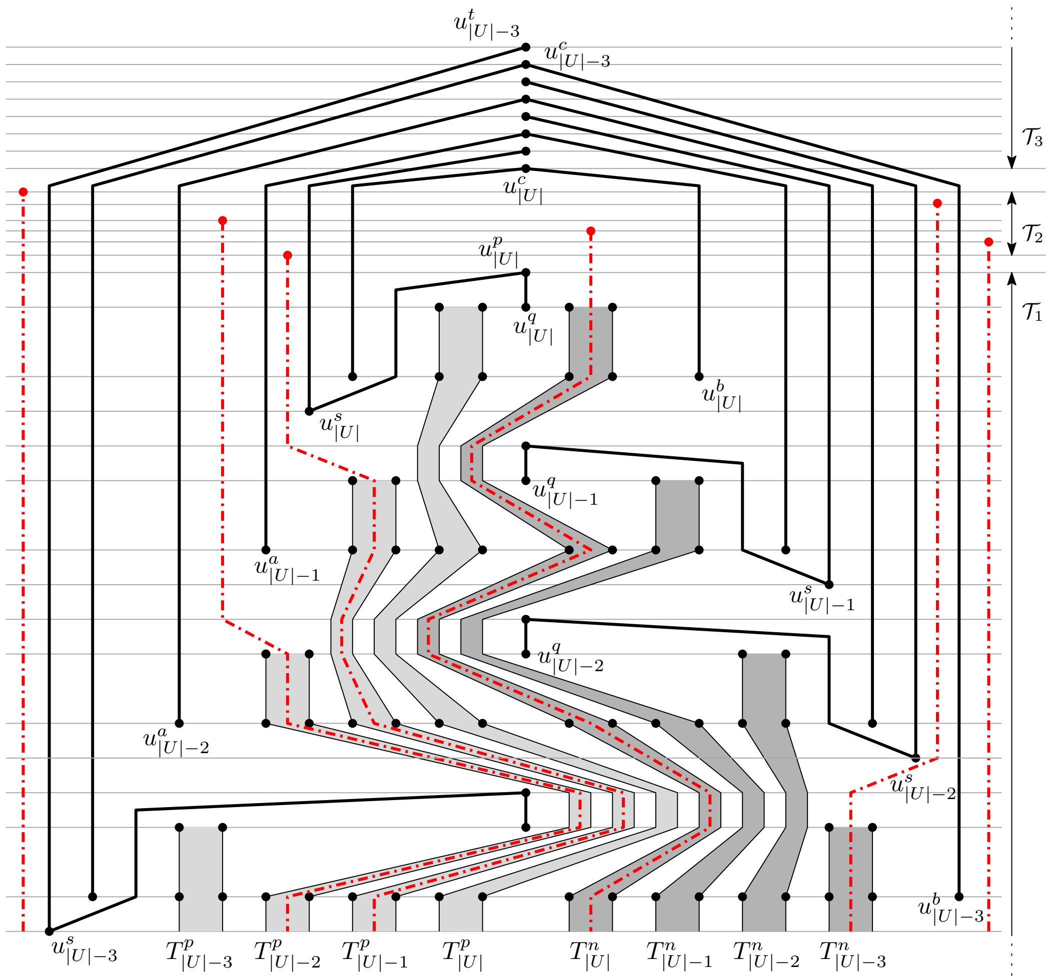
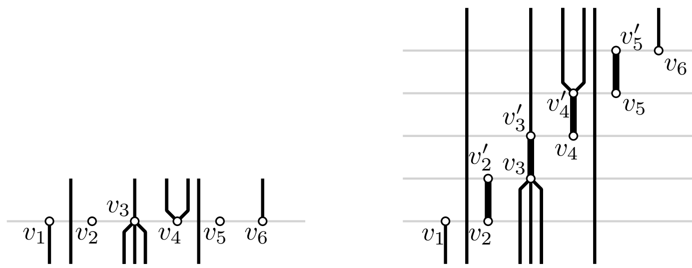


Fig. 11. The nesting structure of the variable gadgets. Only the gadgets of the variables with the four largest indices are shown. They are nested within the remaining variable gadgets. Tier $\mathcal{T}_0$ is located below *all* these gadgets. As in Figure 10, this figure uses the version of Ordered Level Planarity which uses relative x-coordinates on each level. The dash-dotted edges are clause edges.

For $j = 1, \ldots, |\mathcal{U}|$ if u_j is positive, we draw the long edge (u^s_j, u^t_j) to the right of u^b_j and u^c_j and, accordingly, we have to draw all tunnels left of u^s_j and u^q_j (except for T^n_j, which has to be drawn to the left of u^s_j and must end to the right of u^q_j), see Figure 10b. If u_j is negative we draw the long edge (u^s_j, u^t_j) to the left of u^b_j and u^c_j and, accordingly, we have to draw all tunnels right of u^s_j and u^q_j (except for T^p_j, which has to be drawn to the right of u^s_j and end to the left of u^q_j), see Figure 10c. We have to draw the blocking edge (u^s_j, u^p_j) to the right of n^{j+1}_j if u_j is positive and to the left of P^{j+1}_j if u_j is negative.

It remains to describe how to draw the clause edges. Let c_i be a clause. There is at least one true literal in c_i. Let k be the index of the corresponding variable. We describe the drawing of clause edge $e_i = (c^s_i, c^t_i)$ from bottom to top. We start by drawing e_i in the tunnel T^p_k (T^n_k) if c_i is



(a) (b)

Fig. 12. **(a)** A level V_i with width $\lambda_i > 2$. **(b)** In order to reduce the level-width, we replace V_i with $\lambda_i - 1$ levels. Thick edges are the stretch edges.

positive (negative). Immediately after level $\gamma(p_k^{k+1})$ we end up to the left (right) of all tunnels with index larger than k, see Figure 10b (Figure 10c). Note that since T_k^p (T_k^n) is not blocked, we can continue without having to cross blocking edge (u_k^s, u_k^p) or (u_k^q, u_k^p). We draw e_i to the left (right) of all vertices belonging to variable gadgets with index larger than k, see Figure 11. This introduces no crossings since above level $\gamma(p_k^{k+1})$ all tunnels with index larger than k are drawn to the right of $u_{k+1}^a, \dots, u_{|\mathcal{U}|}^a$ and the left of $u_{k+1}^b, \dots, u_{|\mathcal{U}|}^b$. Connecting to c_i^t in tier $\mathcal{T}_2$ is straight-forward since every level contains only one vertex. $\square$

We obtain $\mathcal{NP}$-hardness for instances with maximum degree $\Delta = 2$. In fact, we can restrict our attention to instances with level-width $\lambda = 2$. To this end, we split levels with width $\lambda_i > 2$ into $\lambda_i - 1$ levels containing exactly two vertices each.

Lemma 4.2. *An instance $\mathcal{G} = (G = (V, E), \gamma, \chi)$ of* Ordered Level Planarity *with maximum degree Δ and level-width $\lambda > 2$ can be transformed in linear time into an equivalent instance $\mathcal{G}' = (G' = (V', E'), \gamma', \chi')$ of* Ordered Level Planarity *with maximum degree $\Delta' \leq \Delta + 1$ and level-width $\lambda' = 2$. Further, if $\mathcal{G}$ contains no source or sink with degree Δ on a level V_i with width $\lambda_i > 2$, then $\Delta' \leq \Delta$.*

Proof. We replace each level V_i with width $|V_i| = \lambda_i > 2$ by $\lambda_i - 1$ levels with 2 vertices each, as illustrated in Figure 12. Accordingly, vertices on levels above V_i are shifted upwards by $\lambda_i - 2$ levels. Formally, let $V_i = \{v_1, \dots, v_{\lambda_i}\}$ with $\chi(v_1) < \cdots < \chi(v_{\lambda_i})$. We increase the level of vertex v_j by $j-2$ for $j = 3, \dots, \lambda_i$. For $j = 2, \dots, \lambda_i - 1$ we create a vertex v_j' one level above v_j with $\chi(v_j') = 0$ and we create a new *stretch* edge (v_j, v_j'). For $j = 2, \dots, \lambda_i$ we set $\chi(v_j) = 1$.

For all the vertices v that have been split in this way into v and v', the bottom vertex v inherits all the incoming edges and the top vertex v' inherits all the outgoing edges. Let $\mathcal{G}'$ denote the resulting instance, which can be constructed in linear time. It is easy to verify that the vertex degrees behave as desired.

An ordered level planar drawing of $\mathcal{G}$ can easily be converted to a drawing of $\mathcal{G}'$. For the conversion in the other direction, we successively contract each stretch edge (v_i, v_i') back into a single vertex, thereby merging two consecutive levels of $\mathcal{G}'$. Apart from the edge (v_i, v_i'), the vertex v_i has incident edges from below and the vertex v_i' has incident edges from above only. Therefore, such a contraction cannot cause any problems. The stretch edges ensure that the vertices of each level of $\mathcal{G}$ end up in the correct order. $\square$

Corollary 4.3. *Ordered Level Planarity is $\mathcal{NP}$-hard, even for acyclic ordered level graphs with maximum degree $\Delta = 2$ and level-width $\lambda = 2$.*

The reduction in Lemma 4.1 requires degree-2 vertices. With $\Delta = 1$, the problem becomes polynomial-time solvable. In fact, one can easily solve it as long as the maximum in-degree and the maximum out-degree are both bounded by 1.

Lemma 4.4. *Ordered Level Planarity restricted to instances with maximum in-degree $\Delta^- = 1$ and maximum out-degree $\Delta^+ = 1$ can be solved in linear time.*

Proof. Let $\mathcal{G} = (G = (V, E), \gamma, \chi)$ be an ordered level graph with maximum indegree $\Delta^- = 1$ and maximum outdegree $\Delta^+ = 1$. Such a graph $\mathcal{G}$ consists of a set P of y-monotone paths. Each path $p \in P$ has vertices on some sequence of levels, possibly skipping intermediate levels.

We define the following relation on P: We write $p \prec q$, meaning that p must be drawn to the left of q, if p and q have vertices v_p and v_q that lie adjacent on a common level, i.e. $\gamma(v_p) = \gamma(v_q)$ and $\chi(v_q) = \chi(v_p) + 1$. This relation has at most $|V|$ pairs, and by topological sorting, we can find in $O(|V|)$ time a linear ordering that is consistent with the relation $\prec$, unless this relation has a cycle. The former case implies the existence of an ordered level drawing while the latter case implies that the problem has no solution.

This follows from considerations about horizontal separability of y-monotone sets by translations, cf. [3, 9]. An easy proof can be given following Guibas and Yao [16, 17]: Consider an ordered level planar drawing of $\mathcal{G}$. We say that a vertex is *visible from the left* if the infinite horizontal ray emanating from that vertex to the left does not intersect the drawing. Among the paths whose lower endpoint is visible from the left, the one with the topmost lower endpoint must precede all other paths to which it is related in the $\prec$-relation. Removing this path and iterating the procedure leads to a linear order that extends $\prec$. On the other hand, if we have such a linear order $x \colon P \to \{1, \dots, |P|\}$, we can simply draw each path p straight at x-coordinate $x(p)$, subdivide all edges properly and, finally, shift the vertices on each level such that the vertices of V are placed according to χ while maintaining the order x. □

For $\lambda = 1$, Ordered Level Planarity is solvable in linear time since Level Planarity can be solved in linear time [20]. Proper instances have a unique drawing (if it exists). The existence can be checked with a simple linear-time sweep through every level. The problem is obviously contained in $\mathcal{NP}$. The results of this section establish Theorem 1.1.

Theorem 1.1. *Ordered Level Planarity is $\mathcal{NP}$-complete, even for acyclic ordered level graphs with maximum degree $\Delta = 2$ and level-width $\lambda = 2$. The problem can be solved in linear time if the given level graph is proper; or if the level-width is $\lambda = 1$; or if $\Delta^+ = \Delta^- = 1$, where Δ^+ and Δ^- are the maximum in-degree and out-degree respectively.*

5 CONNECTED INSTANCES

In order to be able to reduce from Ordered Level Planarity to Geodesic Planarity, our main reduction (to Ordered Level Planarity) is tailored to achieve a small maximum degree of $\Delta = 2$. As a consequence, the resulting graphs are not connected. At the cost of an increased maximum degree, it is possible to make our instances connected by inserting additional edges. In this section, we discuss the necessary adaptations in order to obtain the following theorem.

Theorem 1.8. *The following problems are $\mathcal{NP}$-hard even for connected instances with maximum degree $\Delta = 4$:*

- *Ordered Level Planarity even for level-width $\lambda = 2$,*

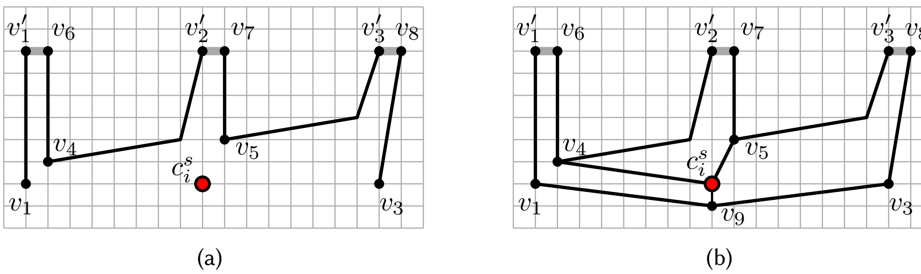


Fig. 13. **(a)** The original clause gadget and **(b)** the augmented version for the connected case. The clause edge starting at c_i^s is not shown.

- *Constrained Level Planarity even for level-width $\lambda = 2$ and prescribed* total *orderings,*
- *Clustered Level Planarity even for level-width $\lambda = 2$ and flat cluster hierarchies that partition the vertices into two non-trivial clusters, and*
- *Bi-Monotonicity.*

We begin by showing the $\mathcal{NP}$-hardness of Ordered Level Planarity for connected instances.

Lemma 5.1. *Planar Monotone 3-Satisfiability reduces in polynomial time to Ordered Level Planarity. The resulting instances are connected and have maximum degree $\Delta = 4$. The maximum in-degree Δ^- and maximum out-degree Δ^+ are both* 3.

Proof. We proceed exactly as in Lemma 4.1. We augment the resulting instances such that they become connected. During this augmentation step, we need to make sure that the degree constraints remain satisfied.

Recall that $\mathcal{U}$ is the set of variables and that tier $\mathcal{T}_3$ contains precisely $2|\mathcal{U}|$ vertices each of which is the only vertex of its level, see Figure 10a and Figure 11. We connect all these vertices with a directed path, that is, we insert the edges (u_j^c, u_j^t) for $j = 1, \ldots, |\mathcal{U}|$ and the edges (u_j^t, u_{j+1}^c) for $j = 2, \ldots, |\mathcal{U}|$. One can easily check that the degree constraints are satisfied: The degree of the vertices u_j^t is now 3 (except for u_1^t, which has degree 2). The degree of the vertices u_j^c is now $4 = \Delta$ (except for $u_{|\mathcal{U}|}^c$, which has degree 3). The largest out-degree of all these vertices is $1 < \Delta^+$, while the largest in-degree is $3 = \Delta^-$.

Recall that for each clause c_i we have created a clause gadget as depicted in Figure 13a. We replace this graph with the graph shown in Figure 13b. Precisely, we do the following: We add a new vertex v_9 one unit below c_i^s and we add the edges (v_9, c_i^s), (c_i^s, v_4), (c_i^s, v_5). Again, the degree bounds are easily verified: Vertex c_i^s now has degree $4 = \Delta$ (including the clause edge); vertices v_4, v_5 and v_9 have degree 3 and vertices v_1 and v_3 have degree 2. The overall maximum out-degree is $3 = \Delta^+$, while the maximum in-degree is 1.

Recall that the segments $v_1'v_6$, $v_2'v_7$ and $v_3'v_8$ of each clause gadget are called the gates of c_i. All gates (of all clauses) are located on the same level V_g, see Figure 8c. We now ensure that all vertices of V_g become connected to each other. The two vertices that bound each gate are already connected through the augmented clause gadgets. We connect two consecutive vertices u, v from different gates by adding for each such pair u, v a new vertex w one level below V_g with two edges (w, u) and (w, v).

The resulting instance has two connected components: one containing all the clause gadgets, clause edges and tunnels; the other containing all the variable gadgets. We can connect these components by adding a path P between the top-most vertex v_t and bottom-most vertex v_b

of the instance. Note that $v_t = u^t_{|\mathcal{U}|}$. The bottom-most vertex is vertex v_9 of the clause gadget corresponding to the (unique) E-shape with the lowest horizontal line segment. Simply choosing $P = (v_b, v_t)$ would result in an increased maximum out-degree of 4. Instead, we choose the (undirected) path $P = (v_b, v'_b, v'_t, v_t)$, where v'_b and v'_t are new vertices placed below v_b and above v_t respectively. This way, the out-degree of v_b remains 3.

The new connected instance is equivalent to the original one as the clause edge (c^s_i, c^t_i) can still reach each of the three gates of c_i by choosing the corresponding embedding. Aside from the edges incident to the vertices c^s_i, no new edge impairs the realizability of the instance in any way. □

We remark that it is possible to decrease the maximum in-degree guaranteed in Lemma 5.1 to $\Delta^- = 2$ by splitting the vertices u^c_j before the augmentation step.

Since the maximum out-degree and in-degree of the instances produced by Lemma 5.1 are strictly smaller than the maximum degree $\Delta = 4$, it follows that no source or sink has degree Δ. Thus, Lemma 4.2 implies the statement about Ordered Level Planarity and Constrained Level Planarity in Theorem 1.8. The statement about Bi-Monotonicity follows from Theorem 1.4. Finally, the statement about Clustered Level Planarity follows from the fact that the reduction given in Lemma 3.2 does not change the graph except for the subdivisions of the edges and the addition of isolated vertices. This concludes the proof of Theorem 1.8.

6 CONCLUSION

We introduced and studied the problem Ordered Level Planarity. Our main result is an $\mathcal{NP}$-hardness statement that cannot be strengthened. We demonstrated the relevance of our result by stating reductions to several other graph drawing problems. These reductions answer multiple questions posed by the graph drawing community and establish connections between problems that (to the best of our knowledge) have not been considered in the same context before. Recently, Da Lozzo, Di Battista, and Frati [7] used Theorem 1.1 to show the $\mathcal{NP}$-hardness of another generalization of Ordered Level Planarity. We expect that Theorem 1.1 will serve as a useful tool for further reductions.

In Section 5, we extended most of our reductions in order to produce problem instances which are connected. We did not provide such a modification for our reduction to Geodesic Planarity. Due to the increased vertex degrees in Ordered Level Planarity instances generated by Theorem 1.8, our reduction to Geodesic Planarity in Step (i) of Lemma 2.1 breaks down, as there is not enough space anymore to attach all the edges around each vertex. It does not seem straight-forward to modify our construction in order to obtain a reduction to Geodesic Planarity that produces connected instances. Thus, we leave it as an open question whether $\mathcal{NP}$-hardness still holds for connected instances of (Manhattan) Geodesic Planarity.

ACKNOWLEDGMENTS

We thank the authors of [21] for providing us with unpublished information regarding their plane sweep approach for Manhattan Geodesic Planarity.

REFERENCES

[1] Patrizio Angelini, Giordano Da Lozzo, Giuseppe Di Battista, Fabrizio Frati, Maurizio Patrignani, and Ignaz Rutter. 2016. Beyond Level Planarity. In *Graph Drawing and Network Visualization - 24th International Symposium, GD 2016, Athens, Greece, September 19-21, 2016, Revised Selected Papers (Lecture Notes in Computer Science)*, Yifan Hu and Martin Nöllenburg (Eds.), Vol. 9801. Springer, 482–495. DOI: 10.1007/978-3-319-50106-2_37

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

[3] Andrei Asinowski, Tillmann Miltzow, and Günter Rote. 2015. Quasi-parallel segments and characterization of unique bichromatic matchings. *Journal of Computational Geometry* 6 (2015), 185–219. DOI: 10.20382/jocg.v6i1a8 arXiv:1302.4400

[4] Christian Bachmaier, Franz-Josef Brandenburg, and Michael Forster. 2005. Radial Level Planarity Testing and Embedding in Linear Time. *J. Graph Algorithms Appl.* 9, 1 (2005), 53–97. http://jgaa.info/accepted/2005/BachmaierBrandenburgForster2005.9.1.pdf

[5] Christian Bachmaier and Wolfgang Brunner. 2008. Linear Time Planarity Testing and Embedding of Strongly Connected Cyclic Level Graphs. In *Algorithms - ESA 2008, 16th Annual European Symposium, Karlsruhe, Germany, September 15-17, 2008. Proceedings (Lecture Notes in Computer Science)*, Dan Halperin and Kurt Mehlhorn (Eds.), Vol. 5193. Springer, 136–147. DOI: 10.1007/978-3-540-87744-8_12

[6] Guido Brückner and Ignaz Rutter. 2017. Partial and Constrained Level Planarity. In *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2017, Barcelona, Spain, Hotel Porta Fira, January 16-19*, Philip N. Klein (Ed.). SIAM, 2000–2011. DOI: 10.1137/1.9781611974782.130

[7] Giordano Da Lozzo, Giuseppe Di Battista, and Fabrizio Frati. 2019. Extending Upward Planar Graph Drawings. In *15th Algorithms and Data Structures Symposium (WADS '19)*. arXiv:1902.06575 To appear.

[8] Mark de Berg and Amirali Khosravi. 2012. Optimal Binary Space Partitions for Segments in the Plane. *Int. J. Comput. Geometry Appl.* 22, 3 (2012), 187–206. http://www.worldscinet.com/doi/abs/10.1142/S0218195912500045

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

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

[11] Michael Forster and Christian Bachmaier. 2004. Clustered Level Planarity. In *SOFSEM 2004: Theory and Practice of Computer Science, 30th Conference on Current Trends in Theory and Practice of Computer Science, Merin, Czech Republic, January 24-30, 2004 (Lecture Notes in Computer Science)*, Peter van Emde Boas, Jaroslav Pokorný, Mária Bieliková, and Julius Stuller (Eds.), Vol. 2932. Springer, 218–228. DOI: 10.1007/978-3-540-24618-3_18

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

[13] Ashim Garg and Roberto Tamassia. 2001. On the Computational Complexity of Upward and Rectilinear Planarity Testing. *SIAM J. Comput.* 31, 2 (2001), 601–625. DOI: 10.1137/S0097539794277123

[14] Emilio Di Giacomo, Fabrizio Frati, Radoslav Fulek, Luca Grilli, and Marcus Krug. 2013. Orthogeodesic point-set embedding of trees. *Comput. Geom.* 46, 8 (2013), 929–944. DOI: 10.1016/j.comgeo.2013.04.003

[15] Emilio Di Giacomo, Luca Grilli, Marcus Krug, Giuseppe Liotta, and Ignaz Rutter. 2012. Hamiltonian orthogeodesic alternating paths. *J. Discrete Algorithms* 16 (2012), 34–52. DOI: 10.1016/j.jda.2012.04.012

[16] Leonidas J. Guibas and F. Francis. Yao. 1980. On Translating a Set of Rectangles. In *Proc. 12th Annual ACM Symposium Theory of Computing (STOC 1980)*. 154–160. DOI: 10.1145/800141.804663

[17] Leonidas J. Guibas and F. Francis. Yao. 1983. On Translating a Set of Rectangles. In *Computational Geometry*, F. P. Preparata (Ed.). Advances in Computing Research, Vol. 1. JAI Press, Greenwich, Conn., 61–77.

[18] Lenwood S. Heath and Sriram V. Pemmaraju. 1995. Recognizing Leveled-Planar Dags in Linear Time. In *Graph Drawing, Symposium on Graph Drawing, GD '95, Passau, Germany, September 20-22, 1995, Proceedings (Lecture Notes in Computer Science)*, Franz-Josef Brandenburg (Ed.), Vol. 1027. Springer, 300–311. DOI: 10.1007/BFb0021813

[19] Michael Jünger, Sebastian Leipert, and Petra Mutzel. 1997. Pitfalls of Using PQ-Trees in Automatic Graph Drawing. In *Graph Drawing, 5th International Symposium, GD '97, Rome, Italy, September 18-20, 1997, Proceedings (Lecture Notes in Computer Science)*, Giuseppe Di Battista (Ed.), Vol. 1353. Springer, 193–204. DOI: 10.1007/3-540-63938-1_62

[20] Michael Jünger, Sebastian Leipert, and Petra Mutzel. 1998. Level Planarity Testing in Linear Time. In *Graph Drawing, 6th International Symposium, GD'98, Montréal, Canada, August 1998, Proceedings (Lecture Notes in Computer Science)*, Sue Whitesides (Ed.), Vol. 1547. Springer, 224–237. DOI: 10.1007/3-540-37623-2_17

[21] Bastian Katz, Marcus Krug, Ignaz Rutter, and Alexander Wolff. 2009. Manhattan-Geodesic Embedding of Planar Graphs. In *Graph Drawing, 17th International Symposium, GD 2009, Chicago, IL, USA, September 22-25, 2009. Revised Papers (Lecture Notes in Computer Science)*, David Eppstein and Emden R. Gansner (Eds.), Vol. 5849. Springer, 207–218. DOI: 10.1007/978-3-642-11805-0_21

[22] Boris Klemz and Günter Rote. 2017. Ordered Level Planarity, Geodesic Planarity and Bi-Monotonicity. In *Graph Drawing and Network Visualization - 25th International Symposium, GD 2017, Boston, MA, USA, September 25-27, 2017, Revised Selected Papers (Lecture Notes in Computer Science)*, Fabrizio Frati and Kwan-Liu Ma (Eds.), Vol. 10692. Springer, 440–453. DOI: 10.1007/978-3-319-73915-1_34

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