



Contents lists available at ScienceDirect

## Journal of Computer and System Sciences

journal homepage: [www.elsevier.com/locate/jcss](http://www.elsevier.com/locate/jcss)On the complexity of the storyplan problem <sup>☆</sup>Carla Binucci <sup>a</sup>, Emilio Di Giacomo <sup>a</sup>, William J. Lenhart <sup>b</sup>, Giuseppe Liotta <sup>a</sup>,  
Fabrizio Montecchiani <sup>a</sup>, Martin Nöllenburg <sup>c,\*</sup>, Antonios Symvonis <sup>d</sup><sup>a</sup> Department of Engineering, University of Perugia, Italy<sup>b</sup> Department of Computer Science, Williams College, USA<sup>c</sup> Algorithms and Complexity Group, TU Wien, Vienna, Austria<sup>d</sup> School of Applied Mathematical & Physical Sciences, NTUA, Greece

## ARTICLE INFO

## Article history:

Received 2 February 2023

Received in revised form 7 July 2023

Accepted 4 August 2023

Available online 11 August 2023

## Keywords:

Dynamic graph drawing

NP-hardness

Parameterized complexity

## ABSTRACT

We study the problem of representing a graph as a storyplan, a recently introduced model for dynamic graph visualization. It is based on a sequence of frames, each showing a subset of vertices and a planar drawing of their induced subgraphs, where vertices appear and disappear over time. Namely, in the StoryPlan problem, we are given a graph and we want to decide whether there exists a total vertex appearance order for which a storyplan exists. We prove that the problem is NP-complete, and complement this hardness with two parameterized algorithms, one in the vertex cover number and one in the feedback edge set number of the input graph. We prove that partial 3-trees always admit a storyplan, which can be computed in linear time. Finally, we show that the problem remains NP-complete if the vertex appearance order is given and we have to choose how to draw the frames.

© 2023 The Author(s). Published by Elsevier Inc. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Let  $G = (V, E)$  be a graph with  $n$  vertices. We write  $[n]$  as shorthand for the set  $\{1, 2, \dots, n\}$ . A *storyplan*  $\mathcal{S} = \langle \tau, \{D_i\}_{i \in [n]} \rangle$  of  $G$  is a pair defined as follows. The first element is a bijection  $\tau : V \rightarrow [n]$  that represents a total order of the vertices of  $G$ . For a vertex  $v \in V$ , let  $i_v = \tau(v)$  and let  $j_v = \max_{u \in N[v]} \tau(u)$ , where  $N[v]$  is the set containing  $v$  and its neighbors. The *lifespan* of  $v$  is the interval  $[i_v, j_v]$ . We say that  $v$  *appears* at step  $i_v$ , is *visible* at step  $i$  for each  $i \in [i_v, j_v]$ , and *disappears* at step  $j_v + 1$ . Note that a vertex disappears as soon as all its neighbors have appeared. The second element of  $\mathcal{S}$  is a sequence of drawings  $\{D_i\}_{i \in [n]}$ , such that: (i) each drawing  $D_i$  contains all vertices visible at step  $i$ , (ii) each drawing  $D_i$  is planar, (iii) the point representing a vertex  $v$  is the same over all drawings that contain  $v$  (i.e., it does not change during the lifespan of  $v$ ), and (iv) the curve representing an edge  $e$  is the same over all drawings that contain  $e$ . We introduce the STORYPLAN problem.

<sup>☆</sup> This work started at the 2022 Bertinoro Workshop on Graph Drawing. A preliminary version of this paper appeared in the proceedings of GD 2022 [1]. Research partially supported by MIUR grant 20174LF3T8 "AHeAD: efficient Algorithms for HAarnessing networked Data", the University of Perugia Progetto RICBA21LG "Algoritmi, modelli e sistemi per la rappresentazione visuale di reti", Progetto RICBA22CB "Modelli, algoritmi e sistemi per la visualizzazione e l'analisi di grafi e reti", and the Vienna Science and Technology Fund (WWTF) [10.47379/ICT19035]. The authors acknowledge TU Wien Bibliothek for financial support through its Open Access Funding Programme.

\* Corresponding author.

E-mail addresses: [carla.binucci@unipg.it](mailto:carla.binucci@unipg.it) (C. Binucci), [emilio.digiacoimo@unipg.it](mailto:emilio.digiacoimo@unipg.it) (E. Di Giacomo), [wlenhart@williams.edu](mailto:wlenhart@williams.edu) (W.J. Lenhart), [giuseppe.liotta@unipg.it](mailto:giuseppe.liotta@unipg.it) (G. Liotta), [fabrizio.montecchiani@unipg.it](mailto:fabrizio.montecchiani@unipg.it) (F. Montecchiani), [noellenburg@ac.tuwien.ac.at](mailto:noellenburg@ac.tuwien.ac.at) (M. Nöllenburg), [symvonis@math.ntua.gr](mailto:symvonis@math.ntua.gr) (A. Symvonis).

**STORYPLAN****Input:** Graph  $G = (V, E)$ **Question:** Does  $G$  admit a storyplan?

In what follows, each drawing  $D_i$  of a storyplan  $\mathcal{S}$  is called a *frame* of  $\mathcal{S}$ . Also, we denote by  $|D_i|$  the number of vertices of  $D_i$ , while the *width* of  $\mathcal{S}$  is  $w(\mathcal{S}) = \max_{i \in [n]} |D_i| - 1$  (we subtract one to align the definition with other width parameters). If  $G$  admits a storyplan, then the *framewidth* of  $G$ , denoted by  $\text{fw}(G)$ , is the minimum width over all its storyplans; otherwise, the framewidth of  $G$  is conventionally set to  $+\infty$ . We will observe that the framewidth of  $G$  upper bounds its pathwidth [2], since each frame can be interpreted as a bag of a path decomposition with the addition of conditions (ii)–(iv).

**Motivation and related work.** Testing for the existence of a storyplan of a graph generalizes planarity and it is of theoretical interest as it combines classical width parameters of graphs with topological properties. From a more practical perspective, computing a storyplan (if any) of a graph  $G$  is a natural way to gradually visualize  $G$  in a story-like or small-multiples fashion, such that each single drawing is planar and the reader's mental map is preserved throughout the sequence of drawings (see, e.g., [3] for a similar approach). More in general, the problem of visualizing graphs that change over time has motivated a notable amount of literature in graph drawing and network visualization (see, e.g., [4–9]). While numerous dynamic graph visualization models have been proposed, two models are of particular interest for our research. The first one is the *graph stories* model introduced by Borrazzo et al. [6]. A *graph story* is formed by a graph  $G$ , a total order of its vertices  $\tau$ , and a positive integer  $W$ . The problem is to find a sequence of drawings  $\{D_i\}_{i \in [n]}$  in which each  $D_i$  contains all vertices  $v$  such that  $i - W < \tau(v) \leq i$ , and the position of a vertex is the same over all drawings it belongs to. Borrazzo et al. prove that any story of a path or a tree can be drawn on a  $2W \times 2W$  and on an  $(8W + 1) \times (8W + 1)$  grid, respectively, so that all the drawings of the story are straight-line and planar. Note that having a fixed window of size  $W$  implies that at most  $O(W \cdot n)$  edges of  $G$  can be represented, in particular, any edge whose endpoints are at distance larger than  $W$  in  $\tau$  does not appear in any drawing. Having both a fixed order and a fixed lifespan are the key differences with our setting. In particular, unconstrained lifespans allow us to find stories in which all edges are drawn in at least one step, while planarity still guarantees that even large frames are readable. Besides such differences in the models, our focus is on the complexity of the decision problem, rather than on area bounds for specific graph families. More recently, the graph stories model has also been investigated by Di Battista et al. [10] who studied the problem of computing graph stories by using a set of  $W + k$  points, for small values of  $k$ . They prove that the problem is NP-complete in general and it belongs to FPT when parameterized by  $W + k$ ; they also give some positive and negative results on the existence of graph stories for various values of  $k$  and various families of graphs.

The second model is *stream planarity* introduced by Da Lozzo and Rutter [7]. Given a graph  $G$ , a total order  $\tau$  of the edges of  $G$ , and a positive integer  $W$ , stream planarity asks for a sequence of drawings  $\{D_i\}_{i \in [m]}$  in which each  $D_i$  contains all edges  $e$  such that  $i - W < \tau(e) \leq i$ , and the subdrawing of the vertices and edges shared by  $D_i$  and  $D_{i-1}$  is the same in both drawings. Da Lozzo and Rutter prove that there exists a constant value for  $W$  for which the stream planarity problem is NP-complete. They also study a variant where a backbone graph is given whose edges must stay in the drawing at each time step; for this variant they prove that the problem is NP-complete for all  $W \geq 2$  and can be solved in polynomial time when  $W = 1$  or when the backbone graph is biconnected. The difference of stream planarity with our problem, besides the fact that edges are streamed rather than vertices, is again having a fixed order and a fixed lifespan.

**Contribution.** The main results in this paper can be summarized as follows.

- We show that STORYPLAN is NP-complete (Section 3.1). As we reduce from ONE-IN-THREE 3SAT and we blow up the instance by a linear factor, it follows that there is no algorithm that solves STORYPLAN in  $2^{o(n)}$  time unless ETH fails. On the other hand, such a lower bound can be complemented with a simple algorithm running in  $2^{O(n \log n)}$  time.
- Motivated by the above hardness, we study the parameterized complexity of STORYPLAN and describe two fixed-parameter tractable algorithms. We first show that STORYPLAN belongs to FPT when parameterized by the vertex cover number via the existence of a kernel, whose size is however super-polynomial (Section 3.2). We then prove that STORYPLAN parameterized by the feedback edge set number (i.e., the minimum number of edges whose removal makes the graph acyclic) admits a kernel of linear size (Section 3.3).
- In parameterized analysis, a central parameter to consider is treewidth. In this direction, finding a parameterized algorithm for STORYPLAN appears to be an elusive task. However, we show that for partial 3-trees (the graphs with treewidth at most three), a storyplan always exists and can be computed in linear time (Section 3.4).
- Finally, we initiate the study of the complexity of a variant of STORYPLAN in which the total order of the vertices is fixed in advance (but the vertex lifespan remains unconstrained). We prove NP-completeness for this problem via a reduction from SUNFLOWER SEFE [11] (Section 4).

Section 2 contains preliminaries and basic results, in particular, it shows some connections between framewidth and pathwidth. Section 3 contains the main results of this paper, namely the NP-hardness of STORYPLAN along with two param-

eterized algorithms and a technique for partial 3-trees. Section 4 concludes the contribution with the NP-hardness of the fixed-order setting. Section 5 discusses some open problems stemming from our research.

## 2. Preliminaries and basic results

A drawing  $\Gamma$  of a graph  $G = (V, E)$  is a mapping of the vertices of  $V$  to points in the plane  $\mathbb{R}^2$ , and of the edges of  $E$  to Jordan arcs connecting their corresponding endpoints but not passing through any other vertex. The drawing  $\Gamma$  is *planar* if no edge is crossed. A graph is *planar* if it admits a planar drawing. A planar drawing of a planar graph  $G$  subdivides the plane into topologically connected regions, called *faces*. The infinite region is the *outer face*. A *planar embedding*  $\mathcal{E}$  of  $G$  is an equivalence class of planar drawings that define the same set of faces and the same outer face. For any  $V' \subseteq V$ , we denote by  $G[V']$  the subgraph of  $G$  induced by the vertices of  $V'$  and by  $\Gamma[V']$  the subdrawing of  $\Gamma$  representing  $G[V']$ . It may be worth remarking that, although some illustrations use straight-line curves for the edges, our algorithms produce topological drawings that may not be straight-line.

**Connection with pathwidth.** The next properties show some simple connections between storyplans and path decompositions [2]. A *path decomposition* of a graph  $G = (V, E)$  is a sequence  $\{X_i\}_{i \in [h]}$  of subsets of  $V$ , called *bags*, for some integer  $h$ , such that: (1) for each edge  $e = uv \in E$ , there exists a bag  $X_i$  containing both  $u$  and  $v$ , and (2) for every three indices  $i \leq j \leq k$ ,  $X_i \cap X_k \subseteq X_j$ . The *width* of  $\{X_i\}_{i \in [h]}$  is equal to  $\max_{i \in [h]} |X_i| - 1$ , and the *pathwidth* of  $G$  is the minimum width over all its path decompositions.

In order to prove that the pathwidth is upper bounded by the framewidth, we begin with the following lemma.

**Lemma 1.** *Let  $G = (V, E)$  be a graph. If  $G$  admits a storyplan  $\mathcal{S} = \langle \tau, \{D_i\}_{i \in [n]} \rangle$  of width  $\omega = w(\mathcal{S})$ , then it admits a path decomposition of width at most  $\omega$ .*

**Proof.** It is easily seen that  $\omega$  equals the *vertex separation number* of the total order  $\tau$ , that is, for each index  $i \in [n]$  there are at most  $\omega$  vertices  $v$  with  $\tau(v) \leq i$  having a neighbor  $u$  such that  $\tau(u) > i$ . Suppose for a contradiction that this is not the case, namely, there exists an index  $i \in [n]$  such that there are at least  $\omega + 1$  vertices  $v$  with  $\tau(v) \leq i$  having a neighbor  $u$  such that  $\tau(u) > i$ . Let  $u_1$  be the one of such neighbors that appears first. All the at least  $\omega + 1$  vertices are still visible at step  $\tau(u_1)$  and therefore the size of the frame at step  $\tau(u_1)$  is at least  $\omega + 2$ , thus contradicting the fact that  $\mathcal{S}$  has with  $\omega$ . On the other hand, a total order  $\tau$  with vertex separation number  $\omega$  implies the existence of a path decomposition of width at most  $\omega$  [12].  $\square$

**Theorem 1.** *Let  $G = (V, E)$  be a graph, then  $\text{pw}(G) \leq \text{fw}(G)$ . Also, if  $G$  is planar then it always admits a storyplan, and in particular  $\text{pw}(G) = \text{fw}(G)$ .*

**Proof.** If  $G$  does not admit a storyplan, for any total order of its vertices, then  $\text{pw}(G) < \text{fw}(G) = +\infty$ . Otherwise,  $\text{pw}(G) \leq \text{fw}(G)$  by Lemma 1.

If  $G$  is planar, we show that  $\text{fw}(G) \leq \text{pw}(G)$ . Namely, let  $\{X_i\}_{i \in [h]}$  be a path decomposition of  $G$ . Without loss of generality, we can assume that this path decomposition is *nice* (or normalized), i.e.,  $X_1$  contains exactly one vertex, and, for each bag  $X_i$  with  $1 < i \leq h$ , there is a vertex  $v$  such that either  $X_i = X_{i-1} \cup \{v\}$  or  $X_i = X_{i-1} \setminus \{v\}$ . Then we have that at each step at most one vertex is introduced in the decomposition, and hence we can associate each vertex  $v$  with an index  $l_v$  such that  $l_v = 1$  if  $v$  is the first introduced vertex, and  $l_v = l_u + 1$ , where  $l_u$  is the last vertex introduced before  $v$ , otherwise. We define a total order  $\tau$  of  $V$  by setting  $\tau(v) = l_v$ . Let  $\Gamma$  be a planar drawing of  $G$ . Consider the set of  $n$  drawings such that  $D_i$  corresponds to the subdrawing of  $\Gamma$  induced by the vertices that are visible at step  $i$ , for each  $i \in [n]$ . One immediately verifies that  $\langle \tau, \{D_i\}_{i \in [n]} \rangle$  is a storyplan of  $G$  of width  $\text{pw}(G)$ .  $\square$

Since computing the pathwidth is NP-hard for planar graphs of bounded degree [13], the next corollary immediately follows from Theorem 1.

**Corollary 1.** *Computing the framewidth of a graph is NP-hard already for planar graphs of bounded degree.*

Analogously, computing the pathwidth of a graph is FPT in the pathwidth [14], hence computing the framewidth of a planar graph is also FPT in the framewidth.

**Complete bipartite graphs.** It is not difficult to verify that if a graph admits a storyplan, then it does not contain  $K_5$  as a subgraph. However, complete bipartite graphs always admit a storyplan and such storyplans have important properties. In particular, Lemma 4 plays a central role in most of our proofs. In order to prove it, we first prove a few auxiliary lemmas.

**Lemma 2.** *Let  $K_{a,b} = (A \cup B, E)$  be a complete bipartite graph with  $a = |A|$  and  $b = |B|$ . Let  $\mathcal{S} = \langle \tau, \{D_i\}_{i \in [a+b]} \rangle$  be a storyplan of  $K_{a,b}$ . There exists  $i \in [a + b]$  such that all vertices of  $A$  or  $B$  are visible.*

**Proof.** Let  $i$  be such that  $D_i$  contains the largest number  $t$  of vertices of  $A$  over all frames of  $\mathcal{S}$ . If  $t = a$ , we are done. If  $t < a$ , there exist two vertices  $u, v$  of  $A$  such that  $j_u < i_v$ . Note that all vertices in  $B$  are adjacent to  $u$ , and hence they all appear at some step smaller than or equal to  $j_u$ . On the other hand, since all vertices in  $B$  are adjacent to  $v$  as well, they cannot disappear before  $i_v + 1$ . It follows that all vertices of  $B$  are visible at step  $j_u$ .  $\square$

**Lemma 3.** Let  $K_{a,b} = (A \cup B, E)$  be a complete bipartite graph with  $a = |A|$  and  $b = |B|$ . Let  $\mathcal{S} = \langle \tau, \{D_i\}_{i \in [a+b]} \rangle$  be a storyplan of  $K_{a,b}$ . If a step  $i \in [a+b]$  contains at least three vertices of  $A$  (resp.  $B$ ), then it contains at most two vertices of  $B$  (resp.  $A$ ).

**Proof.** The statement immediately follows from the fact any frame of  $\mathcal{S}$  is planar and hence cannot contain  $K_{3,3}$  as a subgraph.  $\square$

We are now ready to prove Lemma 4.

**Lemma 4.** Let  $K_{a,b} = (A \cup B, E)$  be a complete bipartite graph with  $a = |A|$ ,  $b = |B|$ , and  $3 \leq b \leq a$ . Let  $\mathcal{S} = \langle \tau, \{D_i\}_{i \in [a+b]} \rangle$  be a storyplan of  $K_{a,b}$ . Exactly one of  $A$  and  $B$  is such that all its vertices are visible at some  $i \in [a+b]$ .

**Proof.** Consider the interval  $I = [s, t] \subseteq [a+b]$  of maximal length such that the vertices of  $A$  or  $B$ , say  $A$ , are all visible. By Lemma 2,  $I$  is not empty. Let  $j$  be one of the steps that contain the largest number  $h$  of vertices of  $B$ . Observe that, since  $I$  is maximal, one vertex of  $A$  appears at  $s$ . Therefore, any vertex of  $B$  visible at a step smaller than  $s$  is visible also at step  $s$ . Similarly, by the maximality of  $I$ , one vertex of  $A$  disappears at step  $t+1$ , therefore any vertex of  $B$  visible at a step greater than  $t$  is visible also at step  $t$ . Consequently, we can assume that  $j \in I$ , and we can conclude  $h \leq 2$  by Lemma 3.  $\square$

In view of Lemma 4, we have the following key definition.

**Definition 1.** For a complete bipartite graph  $K_{a,b}$  with  $3 \leq b \leq a$  and a storyplan  $\mathcal{S}$  of  $K_{a,b}$ , we call *fixed* the partite set of  $K_{a,b}$  whose vertices are all visible at some step of  $\mathcal{S}$ , and *flexible* the other partite set.

### 3. Complexity of STORYPLAN

In this section we prove that: STORYPLAN is NP-complete and cannot be solved in  $2^{o(n)}$  time unless *Exponential Time Hypothesis* (ETH) fails, but there is an algorithm running in  $2^{O(n \log n)}$  time (Section 3.1); STORYPLAN is in FPT parameterized by vertex cover number or feedback edge set number (Sections 3.2 and 3.3); graphs of treewidth at most 3 always admit a storyplan, which can be computed in linear time (Section 3.4). The ETH, combined with the Sparsification Lemma, states that 3SAT formulas on  $N$  variables and  $M$  clauses cannot be solved in  $2^{o(N+M)}$  time, see [15,16].

#### 3.1. Hardness

We reduce from ONE-IN-THREE 3SAT, a variant of 3SAT which asks whether there is a satisfying assignment in which exactly one literal in each clause is true. Let  $\varphi$  be a 3SAT formula over  $N$  variables  $\{x_i\}_{i \in [N]}$  and  $M$  clauses  $\{C_i\}_{i \in [M]}$ . We construct an instance of STORYPLAN, i.e., a graph  $G = (V, E)$ , as follows; refer to Fig. 1 for an illustration.

**Variable gadget.** Each variable  $x_i$  is represented in  $G$  by a copy  $K(x_i)$  of  $K_{3,3}$ . Let  $A_i$  and  $B_i$  be the two partite sets of  $K(x_i)$ , which we call the *v-sides* of  $K(x_i)$ . A true (false) assignment of  $x_i$  will correspond to set  $A_i$  being flexible (fixed) in a putative storyplan of  $G$  (see Definition 1).

**Clause gadget.** Consider a copy of  $K_{2,2,2} = (U_1 \cup U_2 \cup U_3, F)$ . An *extended*  $K_{2,2,2}$  is the graph obtained from any such a copy by adding three vertices  $s_1, s_2, s_3$ , such that these three vertices are pairwise adjacent, and each  $s_j$  is adjacent to both vertices in  $U_j$ , for  $j \in \{1, 2, 3\}$ . In the following,  $s_1, s_2, s_3$  are the *special vertices* of the extended  $K_{2,2,2}$ , while the other vertices are the *simple vertices*. A clause  $C_i$  is represented in  $G$  by an extended  $K_{2,2,2}$ , denoted by  $K(C_i)$ . In particular, we call each of the three sets of vertices  $U_j \cup \{s_j\}$  a *c-side* of  $K(C_i)$ . The idea is that  $K(C_i)$  admits a storyplan if and only if exactly one c-side is flexible (each c-side will be part of a  $K_{3,3}$ , see the wire gadget below).

**Wire gadget.** Let  $x_i$  be a variable having a literal  $l_{ij}$  in a clause  $C_j$ . Any such variable-clause incidence is represented in  $G$  by a set of three vertices, which we call the *w-side*  $W(l_{ij})$ . All vertices of  $W(l_{ij})$  are connected to all vertices of one of the three c-sides of  $K(C_j)$ , which we call  $U$ , such that the graph induced by  $W(l_{ij}) \cup U$  in  $G$  contains a copy of  $K_{3,3}$ . Also, each vertex of  $W(l_{ij})$  is connected to all vertices of the v-side  $A_i$  ( $B_i$ ) if the literal is positive (negative), such that the graph induced by  $W(l_{ij}) \cup A$  ( $W(l_{ij}) \cup B$ ) in  $G$  is a copy of  $K_{3,3}$ . Also, note that each c-side of  $K(C_j)$  is adjacent to exactly one w-side.

**Lemma 5.** If graph  $G$  admits a storyplan then  $\varphi$  admits a satisfying assignment with exactly one true literal in each clause.

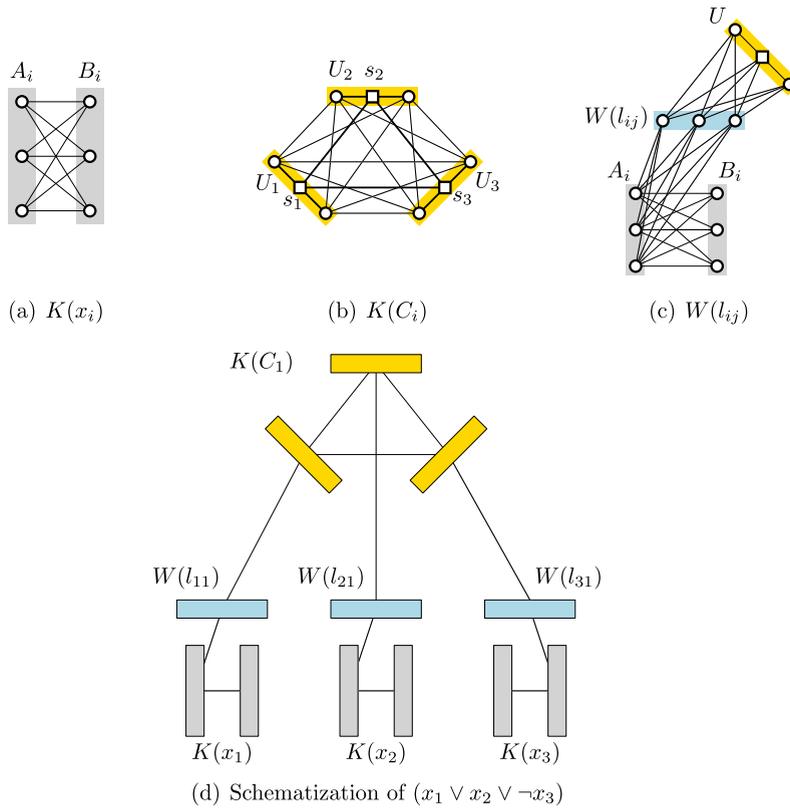


Fig. 1. Illustration for the reduction of Theorem 2.

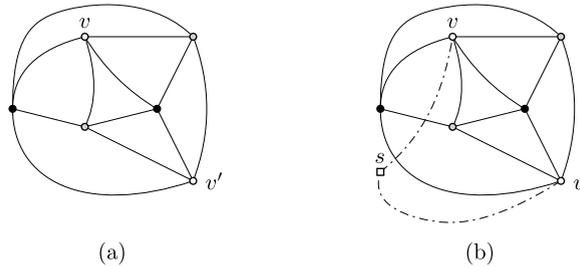


Fig. 2. Illustration for the reduction of Lemma 5.

**Proof.** Let  $S = \langle \tau, \{D_i\}_{i \in [n]} \rangle$  be a storyplan of  $G$ . For each variable gadget  $K(x_i)$ , we assign the value *true* to  $x_i$  if and only if the  $v$ -side  $A_i$  is flexible in  $S$ . Consider any literal  $l_{ij}$  and the wire gadget  $W_{ij}$ . If  $l_{ij}$  is a positive (negative) literal, then  $A_i$  ( $B_i$ ) and  $W_{ij}$  form a  $K_{3,3}$ , hence by Lemma 4 the  $w$ -side  $W_{ij}$  is fixed (flexible). Analogously, if we consider the clause gadget  $K(C_j)$ , the  $c$ -side connected with  $W_{ij}$  is flexible (fixed). If  $A_i$  is fixed, and hence the  $v$ -side  $B_i$  is instead flexible, we assign the value *false* to  $x_i$ . In this case, for any positive (negative) literal  $l_{ij}$ , the  $w$ -side  $W_{ij}$  is flexible (fixed), while the corresponding  $c$ -side of  $K(C_j)$  is fixed (flexible). In other words, the value of  $x_i$  propagates consistently throughout all its literals. It remains to prove that, for any clause  $C_j$  of  $\varphi$ , precisely one  $c$ -side of  $K(C_j)$  is flexible, while the other two are fixed.

We first argue that not all  $c$ -sides can be fixed in  $S$ . Assume, for a contradiction, that they are. Let  $D_h$  be the frame of  $S$  in which the last simple vertex  $v$  of  $K(C_j)$  appears. Observe that all other simple vertices are also visible at step  $h$ . Namely, let  $u \neq v$  be a simple vertex of  $K(C_j)$ . Either  $u$  is adjacent to  $v$  or in the same  $c$ -side as  $v$ . Hence, since  $i_u < h$  by assumption and since  $u$  cannot disappear until  $h + 1$ , it follows that  $u$  is visible at step  $h$ . It follows that  $D_h$  contains a drawing of  $K_{2,2,2}$ , which is a maximal planar graph and hence has a unique planar embedding up to the choice of the outer face. In particular, vertex  $v$  and the other vertex on the same  $c$ -side as  $v$ , which we call  $v'$ , do not belong to the same face; see Fig. 2(a) for an illustration. Consider now the special vertex  $s$  adjacent to both  $v$  and  $v'$ . Vertex  $s$  cannot be visible at step  $h$ , else  $D_h$  would not be planar, as it would contain  $K_{2,2,2}$  plus a subdivided edge between  $v$  and  $v'$ ; see Fig. 2(b). On the other hand, it cannot have disappeared already, since  $s$  and  $v$  are adjacent. Hence,  $s$  has not appeared yet. Similarly, no

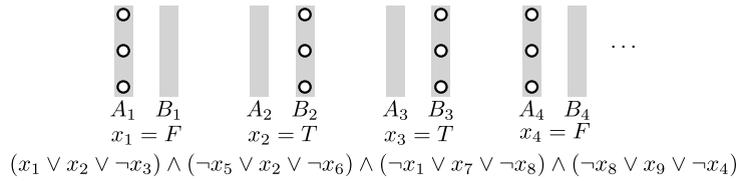


Fig. 3. Proof of Lemma 6: drawing the vertices of the fixed v-sides.

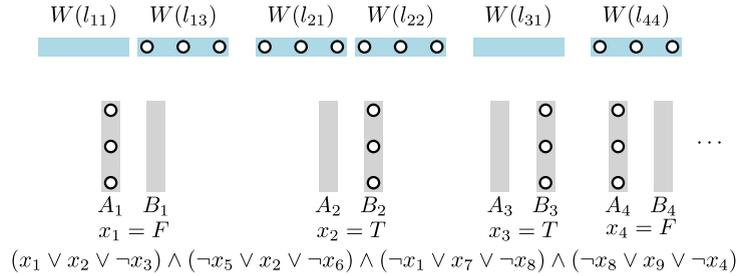


Fig. 4. Proof of Lemma 6: drawing the vertices of the fixed w-sides.

other special vertex is visible at step  $h$ , and since the special vertices are pairwise adjacent, it follows that none of them has appeared yet. Consequently, no vertex of  $K_{2,2,2}$  can disappear until the first special vertex has appeared, which implies the existence of a frame that contains a planar drawing of  $K_{2,2,2}$  plus a subdivided edge between two vertices in the same c-side, which contradicts the fact that the considered frame is planar.

We now rule out the case in which more than one c-side is flexible, which corresponds to a clause with more than one true literal. If a c-side is flexible, there can be no frame containing both its simple vertices. This follows from the fact that the special vertex is adjacent to both simple vertices, and hence there would be a frame containing all three vertices together, which is not possible since the c-side is flexible. On the other hand, when the last simple vertex appears, its four neighbors in  $K_{2,2,2}$  are all visible, and hence at least two c-sides must be fixed.

Altogether, we have proved that at least two c-sides are fixed and that at least one c-side is flexible. Therefore, in each clause gadget, exactly one c-side is flexible, which corresponds to having exactly one true literal, as desired.  $\square$

**Lemma 6.** *If the formula  $\varphi$  admits a satisfying assignment with exactly one true literal in each clause, then graph  $G$  admits a storyplan.*

**Proof.** Suppose that  $\varphi$  admits a satisfying assignment with exactly one true literal per clause. We show how to compute a storyplan  $\mathcal{S} = (\tau, \{D_i\}_{i \in [n]})$  of  $G$ . In what follows, when the order of a group of vertices is not specified, it means that any relative order is valid.

Consider a single variable gadget  $K(x_i)$ . If  $x_i$  is true in the satisfying assignment, then we let appear the three vertices of the v-side  $B_i$  of  $K(x_i)$ , that is,  $B_i$  is the fixed side of  $K(x_i)$ . If  $x_i$  is false, we do the opposite, namely we let appear the three vertices of the v-side  $A_i$  of  $K(x_i)$ . This procedure is repeated for all variables in any order. For ease of presentation, we can imagine that all the drawn v-sides are horizontally aligned, as shown in Fig. 3. Thus, for the variable gadgets, it remains to draw their flexible v-sides.

Consider now a wire gadget  $W(l_{ij})$ . If  $x_i$  is true and  $l_{ij}$  is positive, then  $W(l_{ij})$  must be fixed because it forms a  $K_{3,3}$  with the v-side  $A_i$  of  $K(x_i)$ , which is flexible. Therefore, we let appear the three vertices of  $W(l_{ij})$ . Similarly, if  $x_i$  is false and  $l_{ij}$  is negative, then  $W(l_{ij})$  must be fixed, and we let appear the three vertices of  $W(l_{ij})$ . Again, this procedure is repeated for all wires in any order. For ease of presentation, we can imagine that all the drawn w-sides are arranged along a horizontal line slightly above the variable gadgets, as shown in Fig. 4. Thus, also for the wire gadgets, it remains to draw the flexible w-sides.

At this point, observe that each flexible v-side can be drawn independently, by letting appear and disappear its three vertices one by one. When a vertex appears, its edges towards the fixed w-sides of its wire gadgets and the fixed v-side of its variable gadget form a star and thus they can be easily drawn without crossings; see Fig. 5(a).

We now draw the clauses one by one. We begin by showing how to draw a single clause gadget  $K(C_j)$ , ignoring the connections with the linked wire gadgets. Refer to Fig. 5 for an illustration. We first let appear the three special vertices of  $K(C_j)$  (Fig. 5(b)). Now we let appear the two simple vertices of a false literal, which we denote by  $l_1$  for convenience (Fig. 5(c)). Note that the whole c-side representing  $l_1$  is visible. Right after, the special vertex connected to the two simple vertices can disappear. Next, we let appear the two simple vertices of the remaining false literal, called  $l_2$ , hence again this c-side is entirely visible (Fig. 5(d)). Again this can be done without crossings, and the corresponding special vertex can disappear right after. Now we have a 4-cycle (depicted with a light blue background in Fig. 5(d)) formed by the edges connecting the simple vertices drawn so far, and the special vertex of the missing literal lying in one of the two faces made

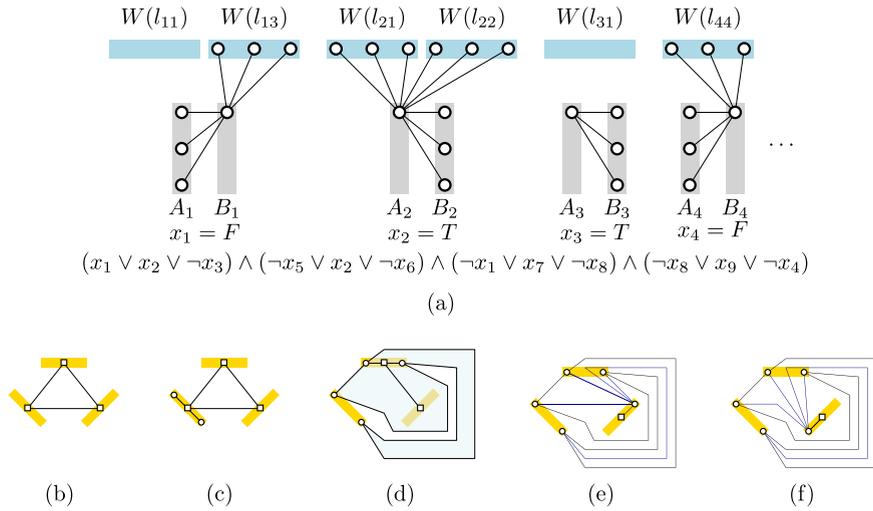


Fig. 5. Proof of Lemma 6: (a) drawing the vertices of the flexible  $v$ -sides; (b)–(g) drawing a clause gadget. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

by the cycle. We let one of the two simple vertices of the true literal appear (Fig. 5(e)), we draw its edges planarly, and afterwards we let it disappear and replace it with the last simple vertex (Fig. 5(f)). Once the last simple vertex has appeared, all vertices of the clause can disappear. Hence, the last  $c$ -side hasn't appeared together but instead acted as a flexible  $c$ -side.

Consider a wire gadget  $W(l_{ij})$ . Suppose first that  $l_{ij}$  is false. Then we have seen that its  $c$ -side in  $K(C_j)$ , say  $U$ , is fixed and hence  $W(l_{ij})$  must be flexible. Note that both if  $l_{ij} = l_1$  (Fig. 6(a)) or  $l_{ij} = l_2$  (Fig. 6(b)), in the frame in which the three vertices of  $U$  are visible there is a face whose boundary contains the three vertices of the corresponding  $c$ -side. We can assume that this face encloses the drawings of all the  $w$ -sides and  $v$ -sides that are currently visible. Hence the vertices of  $W(l_{ij})$  can be drawn one by one in this face. If  $l_{ij}$  is true, then its  $c$ -side  $U$  is flexible and hence  $W(l_{ij})$  must be fixed. Again in the two frames in which the vertices of  $U$  appear (see Figs. 5(e) and 5(f)), the drawing of the clause has a face whose boundary contains the vertex of  $U$  visible in that frame. We can assume that this face encloses the drawings of all the  $w$ -sides and  $v$ -sides that are currently visible, and therefore the vertices of  $U$  can be connected planarly both to its neighbors in  $K(C_j)$  and to its neighbors in  $W(l_{ij})$  (see Fig. 7). By repeating this procedure for each clause we complete the storyplan.  $\square$

We can conclude the following.

**Theorem 2.** *The STORYPLAN problem is NP-hard and it has no  $2^{o(n)}$  time algorithm unless ETH fails.*

**Proof.** Constructing the graph  $G$  from the formula  $\varphi$  clearly takes polynomial time, and the correctness of the reduction follows from Lemmas 5 and 6. This proves that STORYPLAN is NP-hard.

For the second part of the statement, we recall that the ETH, combined with the Sparsification Lemma, states that 3SAT formulas on  $N$  variables and  $M$  clauses cannot be solved in  $2^{o(N+M)}$  time, see [15,16]. On the other hand, there exists a polynomial-time reduction from 3SAT to ONE-IN-THREE 3SAT that transforms 3SAT instances with  $N + M$  variables and clauses into equivalent ONE-IN-THREE 3SAT instances with  $O(N + M)$  variables and clauses [17]. Finally, starting from an instance of ONE-IN-THREE 3SAT, our reduction constructs a graph with  $n$  vertices and  $m$  edges such that  $m = O(n)$  and  $n + m = O(N + M)$ . Therefore, an algorithm solving STORYPLAN in  $2^{o(n)}$  time would contradict ETH.  $\square$

The above lower bound for the running time of an algorithm solving STORYPLAN can be easily complemented with a nearly tight upper bound. The proof of the next theorem also shows that STORYPLAN belongs to NP. Namely, it gives a nondeterministic scheme to generate a set of candidate solutions, and then it shows how to check, in polynomial time, if a candidate solution is valid. In order to obtain a lower time complexity, the process of generating a set of candidate solutions and the process of checking if a solution is valid are not separated, but rather intertwined.

**Theorem 3.** *The STORYPLAN problem is in NP. Also, given an  $n$ -vertex graph  $G$ , there is an algorithm that solves STORYPLAN on  $G$  in  $2^{O(n \log n)}$  time.*

**Proof.** The key observation to claim that the problem belongs to NP lies in the fact that, since we consider topological drawings in which edges are Jordan arcs, once a total order of the vertices is fixed, the existence of a storyplan is equivalent to the existence of a sequence of planar embeddings that are “consistent”. Namely, for any two such planar embeddings,

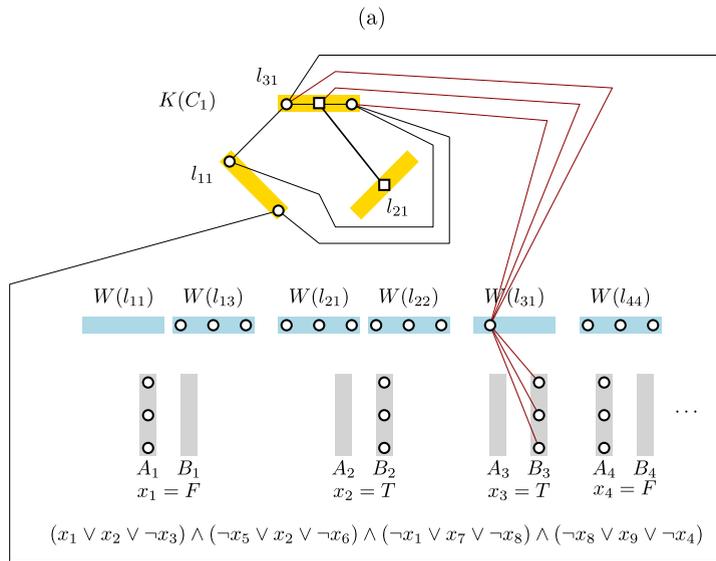
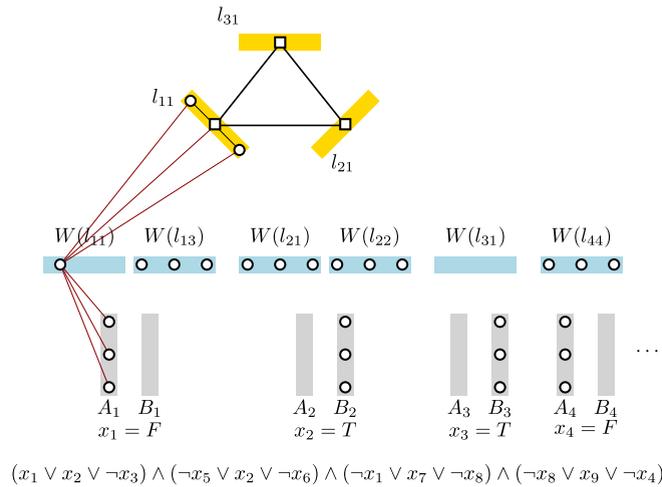


Fig. 6. Proof of Lemma 6: drawing the false literals.

if they share a common subgraph, then their restrictions to this subgraph must coincide. With this observation, we can guess a candidate solution and verify, in polynomial time, if it is valid. However, we intertwine the generation process of the solutions and their validity check in order to obtain a more efficient time complexity.

More formally, we proceed as follows. We first guess a linear order of the vertices of  $G$ . Note that there are  $n! \in 2^{O(n \log n)}$  such orders. For each  $i \in [n]$ , we identify the set of vertices  $V_i$  visible at step  $i$  and the corresponding set of edges  $E_i$ . By planarity, it must be  $|E_i| \leq 3|V_i| - 6$ , else we can immediately reject the instance. As already observed, we do not guess a drawing  $D_i$ , but instead a planar embedding  $\mathcal{E}_i$  for graph  $G_i = (V_i, E_i)$ .

If  $G_i$  is connected, the planar embedding  $\mathcal{E}_i$  can be described by the circular order of the edges around each vertex, called *rotation system*, and by the choice of the outer face. Indeed, with this information, one can reconstruct the boundary of each face of  $G_i$  (in linear time). If  $G$  is not connected, we also need the pairwise relative positions of distinct connected components (if more than one). Let  $C_1, \dots, C_q$  be the connected components of  $G_i$  and let  $\mathcal{E}_i(C_j)$  be the restriction of  $\mathcal{E}_i$  to  $C_j$  for each  $j \in [q]$ . The *position system* is a rooted tree whose vertices are the faces of the embeddings  $\mathcal{E}_i(C_j)$  for each  $j \in [q]$ , and a node  $\nu$  has a parent  $\mu$  in this tree if and only if the face corresponding to  $\nu$  contains the face corresponding to  $\mu$  in its interior and  $\mu$  is the outer face of its connected component.

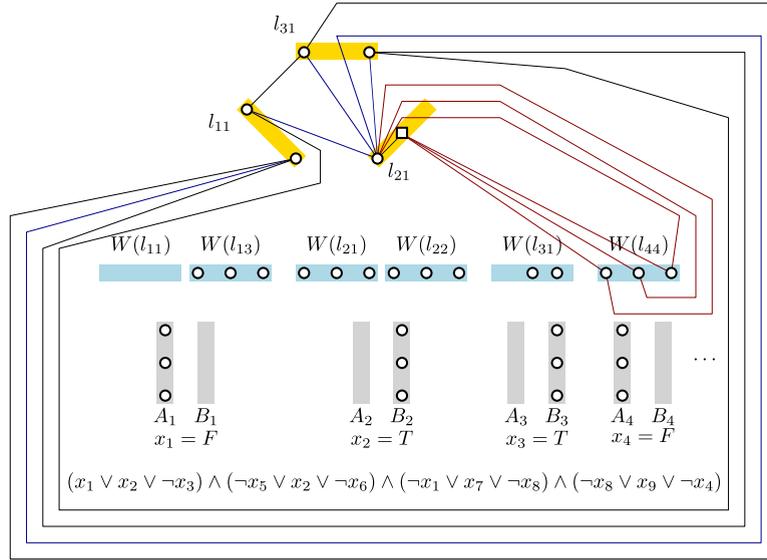


Fig. 7. Proof of Lemma 6: drawing the true literal.

The number of possible rotation systems of  $G_i$ , denoted by  $n_{rot}$ , is upper bounded by the number of possible permutations of edges around each vertex. Thus we have

$$n_{rot} \leq \prod_{v \in V_i} \deg(v)! < \prod_{v \in V_i} \deg(v)^{\deg(v)} \in n^{O(\sum_{v \in V_i} \deg(v))} \subseteq n^{O(n)}.$$

Each rotation system of  $G_i$  fixes the boundary of each face of each connected component of  $G_i$ . Hence we can verify (in linear time) whether the rotation system yields a planar combinatorial (up to the choice of the outer face) embedding of each connected component of  $G_i$ . If this is the case, there are  $O(n)$  possible outer faces for each connected component, we have  $O(n^2)$  possible choices over all connected components. Once both the rotation system and the outer face of each connected component have been fixed, the number  $n_{pos}$  of possible position systems is  $n_{pos} \subseteq n^{O(n)}$ , since it suffices to guess the parent of each node of the tree. Therefore, for each step  $i > 1$ , there are  $O(n^2) \cdot n_{rot} \cdot n_{pos} \subseteq n^{O(n)} \subseteq 2^{O(n \log n)}$  distinct embeddings that we can guess.

Once we have generated all planar embeddings for each  $i \in [n]$ , we aim at finding a sequence (if any) of planar embeddings, one for each step  $i$ , in which common subgraphs have the same embedding throughout the sequence. Namely, for each  $i > 1$  and for each planar embedding  $\mathcal{E}_i$ , we verify whether there is at least one planar embedding  $\mathcal{E}_{i-1}$  such that the restrictions of  $\mathcal{E}_i$  and  $\mathcal{E}_{i-1}$  to the common graph  $G_{i-1} \cap G_i$  are the same. If this is the case, we keep  $\mathcal{E}_i$ , else we discard it. The algorithm halts if the set of planar embeddings becomes empty, else it outputs at least one valid sequence. Overall, the procedure takes  $2^{O(n \log n)} \cdot 2^{O(n \log n)} = 2^{O(n \log n)}$  time.  $\square$

### 3.2. Parameterization by vertex cover number

A vertex cover of a graph  $G = (V, E)$  is a set  $C \subseteq V$  such that every edge of  $E$  is incident to a vertex in  $C$ , and the vertex cover number of  $G$  is the minimum size of a vertex cover of  $G$ . We prove the following by means of kernelization.

**Theorem 4.** Let  $G = (V, E)$  be a graph with  $n$  vertices and vertex cover number  $\kappa = \kappa(G)$ . Deciding whether  $G$  admits a storyplan, and computing one if any, can be done in  $O(2^{2^{O(\kappa)}} + n^2)$  time.

**Algorithm Description.** Without loss of generality, we assume that the input graph  $G$  does not contain isolated vertices, as they do not affect the existence of a storyplan. Let  $C$  be a vertex cover of size  $\kappa = \kappa(G)$  of graph  $G$ . For  $U \subseteq C$ , a vertex  $v \in V \setminus C$  is of type  $U$  if  $N(v) = U$ , where  $N(v)$  denotes the set of neighbors of  $v$  in  $G$ . This defines an equivalence relation on  $V \setminus C$  and in particular partitions  $V \setminus C$  into at most  $\sum_{i=1}^{\kappa} \binom{\kappa}{i} = 2^{\kappa} - 1 < 2^{\kappa}$  distinct types. Denote by  $V_U$  the set of vertices of type  $U$ . We define three reduction rules.

- R.1:** If there exists a type  $U$  such that  $|U| = 1$ , then pick an arbitrary vertex  $x \in V_U$  and remove it from  $G$ .
- R.2:** If there exists a type  $U$  such that  $|U| = 2$  and  $|V_U| > 1$ , then pick an arbitrary vertex  $x \in V_U$  and remove it from  $G$ .
- R.3:** If there exists a type  $U$  such that  $|U| \geq 3$  and  $|V_U| > 3$ , then pick an arbitrary vertex  $x \in V_U$  and remove it from  $G$ .

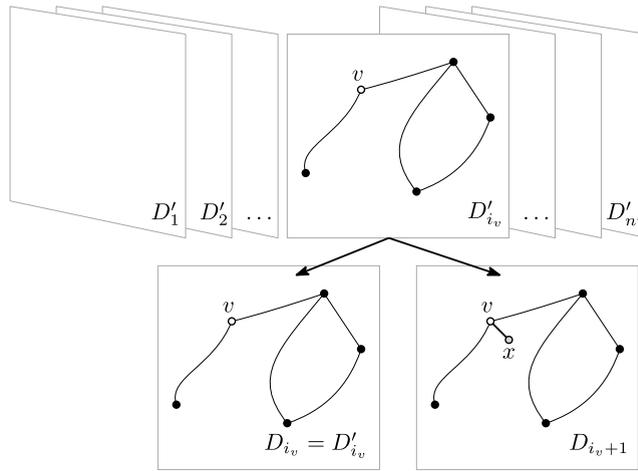


Fig. 8. Illustration for Case A of the proof of Lemma 7.

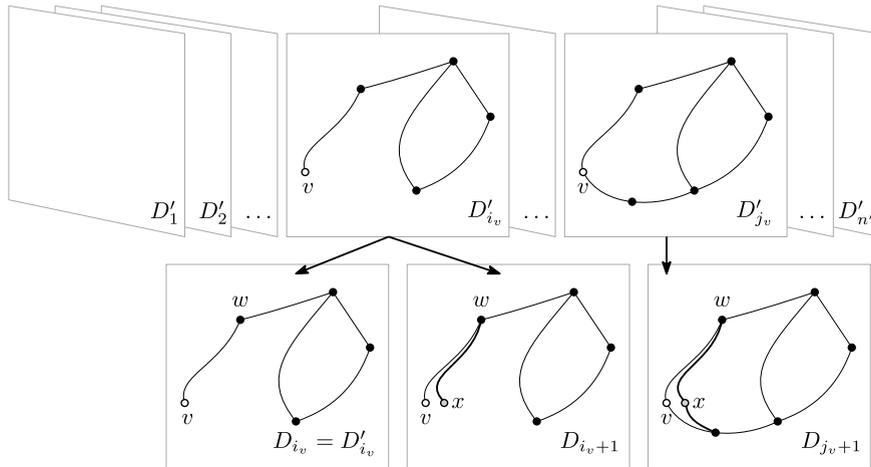


Fig. 9. Illustration for Case B of the proof of Lemma 7.

**Lemma 7.** Let  $G'$  be the graph obtained from  $G$  by applying one of the reduction rules **R.1–R.3**. Then  $G$  admits a storyplan if and only if  $G'$  does.

**Proof.** One direction is trivial, since it is easily seen that admitting a storyplan is a hereditary property. Suppose now that  $G'$  admits a storyplan  $\mathcal{S}' = \langle \tau', \{D'_i\}_{i \in [n']}\rangle$ , where  $n' = n - 1$ . We distinguish three cases based on the reduction rule applied to  $G$ . In each case, we denote by  $x$  the vertex removed from  $G$  to obtain  $G'$ .

**Case A (R.1).** See Fig. 8 for an illustration. Let  $v$  be the neighbor of  $x$  in  $G$ , whose lifespan according to  $\tau'$  is  $[i_v, j_v]$ . We compute  $\tau$  from  $\tau'$  by inserting  $x$  right after  $v$ ; consequently, the lifespan of  $x$  in  $\tau$  is  $[i_v + 1, i_v + 1]$ . Similarly, compute  $\{D_i\}_{i \in [n]}$  from  $\{D'_i\}_{i \in [n']}$  as follows. For each  $i \leq i_v$ , we set  $D_i = D'_i$ . For  $i = i_v + 1$ , we draw  $x$  in  $D'_{i_v}$  sufficiently close to  $v$  such that  $xv$  can be drawn as a straight-line segment that does not intersect any other edge. We then set  $D_i$  to be equal to the resulting drawing. Finally, for each  $i > i_v + 1$ , we set  $D_i = D'_{i-1}$ .

**Case B (R.2).** See Fig. 9 for an illustration. By assumption,  $G'$  contains at least one vertex  $v \neq x$  of type  $U$ , whose lifespan according to  $\tau'$  is  $[i_v, j_v]$ . We compute  $\tau$  from  $\tau'$  by inserting  $x$  right after  $v$ ; consequently, the lifespan of  $v$  in  $\tau$  is  $[i_v + 1, j_v]$ . For each  $i \leq i_v$ , we set  $D_i = D'_i$ . For  $i = i_v + 1$ , we extend  $D'_{i_v}$  by drawing  $x$  sufficiently close to  $v$  and by drawing, for each neighbor  $w$  of  $x$ , the edge  $xw$  such that it follows the curve representing the edge  $vw$ . Since  $vw$  does not cross any other edge and  $v$  has degree two, the same holds for  $xw$ . We then set  $D_i$  to be equal to the resulting drawing. Similarly, for each  $i \leq [i_v + 2, j_v + 1]$ , we extend (if needed) the frame  $D'_{i-1}$  by drawing any edge  $xw$  such that it follows the corresponding curve  $vw$ . Finally, for each  $i > j_v + 1$ , we set  $D_i = D'_{i-1}$ .

**Case C (R.3).** Observe that, by assumption, the graph induced by the vertices of  $U \cup V_U$  contains a complete bipartite graph  $K_{a,b}$  with  $a \geq 3$  and  $b \geq 3$ , whose partite sets are  $U$  and  $V_U$ . We distinguish two subcases based on whether  $U$  is the fixed set or the flexible set of  $K_{a,b}$ .

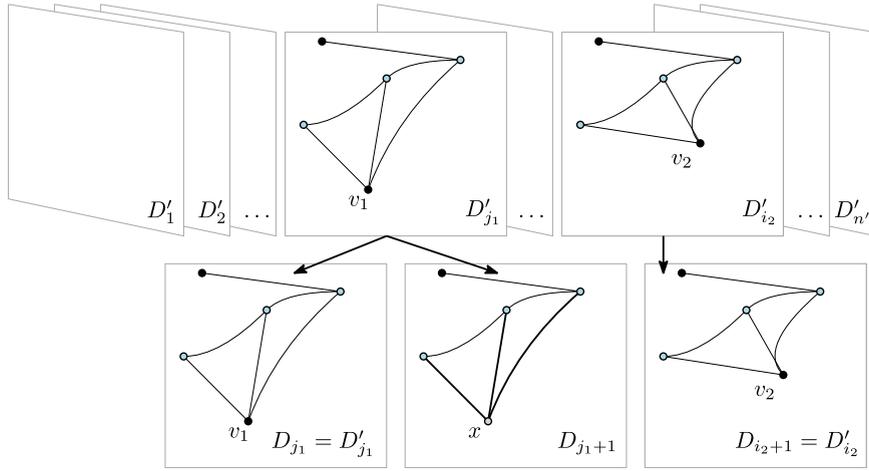


Fig. 10. Illustration for Case C of the proof of Lemma 7, when  $U$  is the fixed set.

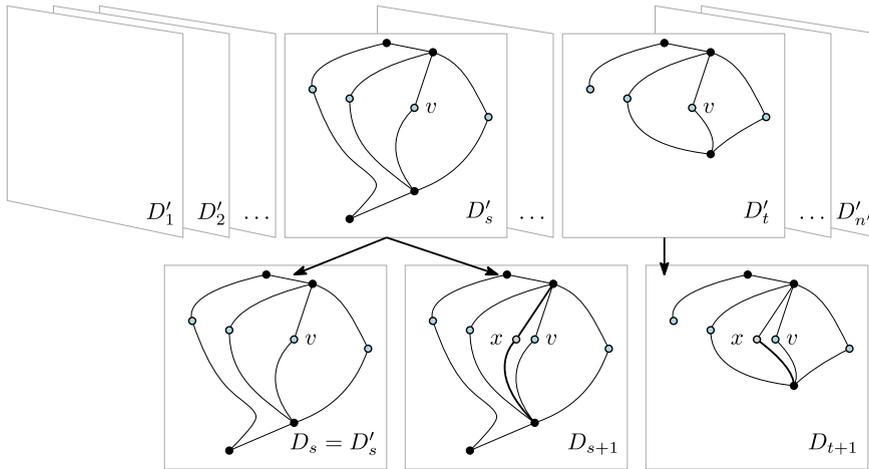


Fig. 11. Illustration for Case C of the proof of Lemma 7, when  $V_U$  is the fixed set.

Suppose first that  $U$  is the fixed set; see Fig. 10 for an illustration. Let  $I \in [n']$  be the interval in which all vertices of  $U$  are visible. By assumption,  $G'$  contains at least three vertices  $v_j \neq x$  of  $V_U$ , with  $j = 1, 2, 3$ . Observe that the lifespan of each vertex  $v_j$  intersects  $I$ . Therefore there are at least two vertices, say  $v_1$  and  $v_2$ , whose lifespans do not intersect, otherwise there would be a frame containing  $K_{3,3}$ . Let  $[i_1, j_1]$  and  $[i_2, j_2]$  be the lifespan of  $v_1$  and  $v_2$ , respectively, and suppose (without loss of generality) that  $j_1 < i_2$ . Observe that both  $j_1$  and  $i_2$  are in  $I$ . We compute  $\tau$  from  $\tau'$  by inserting  $x$  right after  $v_1$  disappears, such that its lifespan in  $\tau$  is  $[j_1 + 1, j_1 + 1]$ . For each  $i \leq j_1$ , we set  $D_i = D'_i$ . For  $i = j_1 + 1$ , we take  $D'_{j_1}$  and replace the drawing of  $v_1$  with the drawing of  $x$ . Namely, we place  $x$  on the same point of  $v_1$  and we draw each curve  $xw$  by following the curve  $v_1w$ . Since  $v_1w$  does not cross any other edge, the same holds for  $xw$ . The resulting drawing is  $D_i$ . Finally, for each  $i > j_1 + 1$ , we set  $D_i = D'_{i-1}$ .

Suppose now that  $V_U$  is the fixed set; see Fig. 11 for an illustration. Let  $[s, t] \in [n']$  be the maximal interval in which all vertices of  $V_U$  in  $G'$  are visible. Let  $v \neq x$  be the vertex of  $V_U$  such that  $\tau'(v) = s$ . We compute  $\tau$  from  $\tau'$  by inserting  $x$  right next  $v$  such that its lifespan in  $\tau$  is  $[s + 1, t + 1]$ . For each  $i \leq s$ , we set  $D_i = D'_i$ . For  $i = s + 1$ , we extend  $D'_i$  by drawing  $x$  sufficiently close to  $v$  and by drawing, for each neighbor  $w$  of  $x$ , the edge  $xw$  such that it follows the curve representing the edge  $vw$ . Note that in each frame  $D'_i$  with  $i \in [s, t]$  there are at most two vertices of  $U$ , and that there are no edges between vertices of  $V_U$ . Hence, similarly as in Case B, vertex  $v$  has degree at most two and the curve  $xw$  does not cross any other edge. We then set  $D_i$  to be equal to the resulting drawing. Similarly, for each  $i \leq [s + 2, t + 1]$ , we extend (if needed) the frame  $D'_{i-1}$  by drawing any edge  $xw$  such that it follows the corresponding curve  $vw$ . Finally, for each  $i > t + 1$ , we set  $D_i = D'_{i-1}$ .  $\square$

**Proof of Theorem 4.** By [18], we can determine the vertex cover number  $\kappa = \kappa(G)$  of  $G$  and compute a vertex cover  $C$  of size  $\kappa$  in time  $O(2^\kappa + \kappa \cdot n)$ . We construct a kernel  $G^*$  from  $G$  of size  $O(2^\kappa)$  as follows. We first classify each vertex

of  $G$  based on its type. We then apply reduction rules **R1**, **R2**, and **R3** exhaustively. Thus, constructing  $G^*$  can be done in  $O(2^\kappa + \kappa \cdot n)$  time, since  $O(2^\kappa)$  is the number of types and  $\kappa \cdot n$  is the maximum number of edges of  $G$ . Also,  $G^*$  contains  $n^* \leq 3 \cdot 2^\kappa + \kappa < 2^{\kappa+2}$  vertices.

From Lemma 7 we conclude that  $G$  admits a storyplan if and only if  $G^*$  does. To establish whether  $G^*$  admits a storyplan we proceed as follows: (1) We guess a total order  $\tau^*$  of  $G^*$ ; (2) For  $i = 1$ , we guess all planar embeddings of the graph induced by the vertices visible at step  $i$ ; (3) For each  $i > 1$ , we consider the embeddings computed at the previous step  $i - 1$ , we remove from them the vertices (if any) that disappear at step  $i$ , we remove possible duplicates, and we try to exhaustively extend each of the resulting planar embeddings by inserting the vertex that appears at step  $i$ . The algorithm halts if the set of planar embeddings becomes empty. It is readily seen that  $G^*$  admits a storyplan if and only if the algorithm terminates at step  $n^*$  with at least one planar embedding. Concerning the time complexity, step (1) takes  $O(2^{\kappa+2}!)$  time. Since  $G^*$  contains  $O(2^{O(\kappa)})$  vertices and edges, the number of possible planar embeddings are  $O((2^{O(\kappa)})^{O(2^{O(\kappa)})}) = O(2^{2^{O(\log \kappa)} \cdot 2^{O(\kappa)}}) = O(2^{2^{O(\kappa)}})$ . Hence step (2) takes  $O(2^{2^{O(\kappa)}})$  time and step (3) takes  $O(2^{2^{O(\kappa)}}) \cdot O(2^{2^{O(\kappa)}}) = O(2^{2^{O(\kappa)}})$  time. Starting from a storyplan of  $G^*$ , we can reinsert the missing  $O(n)$  vertices each in  $O(n)$  time, as detailed in Lemma 7.

### 3.3. Parameterization by feedback edge set

A *feedback edge set* of a graph  $G = (V, E)$  is a set  $F \subseteq E$  whose removal results in an acyclic graph, and the *feedback edge set number* of  $G$  is the minimum size of a feedback edge set of  $G$ . We prove the following.

**Theorem 5.** *Let  $G$  be a graph with  $n$  vertices and feedback edge set of size  $\psi = \psi(G)$ . Deciding whether  $G$  admits a storyplan, and computing one if any, can be done in  $O(2^{O(\psi \log \psi)} + n^2)$  time.*

**Algorithm Description.** A  $k$ -chain of  $G$  is a path with  $k + 2$  vertices and such that its  $k$  inner vertices all have degree two. We define two reduction rules.

**RA:** *If there exists a vertex of degree one, then remove it from  $G$ .*

**RB:** *If there exists a  $k$ -chain with  $k \geq 3$ , then remove its inner vertices from  $G$ .*

We first show that **RA** and **RB** are safe rules.

**Lemma 8.** *Let  $G'$  be the graph obtained from  $G$  by applying one of the reduction rules **RA** and **RB**. Then  $G$  admits a storyplan if and only if  $G'$  does.*

**Proof.** One direction is again trivial, since admitting a storyplan is a hereditary property. Suppose now that  $G'$  admits a storyplan  $S' = \langle \tau', \{D'_i\}_{i \in [n']}\rangle$ , where  $n' < n$ . We distinguish two cases based on the reduction rule applied to  $G$ .

**Case 1 (RA).** The argument is analogous as for rule **R1**. Let  $u$  be the removed vertex and let  $v$  be its neighbor in  $G$ , whose lifespan according to  $\tau'$  is  $[i_v, j_v]$ . We compute  $\tau$  from  $\tau'$  by inserting  $u$  right after  $v$ , consequently, the lifespan of  $v$  in  $\tau$  is  $[i_v + 1, i_v + 1]$ . Similarly, compute  $\{D_i\}_{i \in [n]}$  from  $\{D'_i\}_{i \in [n']}$  as follows. For each  $i \leq i_v$ , we set  $D_i = D'_i$ . For  $i = i_v + 1$ , we draw  $u$  in  $D'_v$  sufficiently close to  $v$  such that  $uv$  can be drawn as a straight-line segment that does not intersect any other edge. We then set  $D_i$  to be equal to the resulting drawing. Finally, for each  $i > i_v + 1$ , we set  $D_i = D'_{i-1}$ .

**Case 2 (RB).** Let  $(s, u_1, \dots, u_k, t)$  be the  $k$ -chain whose inner vertices,  $u_1, \dots, u_k$ , have been removed. Let  $[i_s, j_s]$  and  $[i_t, j_t]$  be the lifespans of  $s$  and  $t$  according to  $\tau'$ , respectively. Without loss of generality, we assume  $i_s < i_t$ . We compute  $\tau$  from  $\tau'$  as follows. We insert  $u_1$  right after  $s$  and  $u_k$  right after  $t$ . Now the order has length  $n' + 2$ . For each  $i \in \{2, \dots, k - 1\}$  (recall that  $k \geq 3$ ), we insert  $u_i$  at step  $[n' + i + 1]$ . Consequently, the lifespans of  $u_1$  and  $u_k$  in  $\tau$  are  $[i_s + 1, n' + 3]$  and  $[i_t + 1, n' + k]$ , respectively, the lifespan of  $u_i$  is  $[n' + i + 1, n' + i + 2]$ , for each  $i \in \{2, \dots, k - 2\}$ , and the lifespan of  $u_{k-1}$  is  $[n' + k, n' + k]$ . For each  $i \leq i_s$ , we set  $D_i = D'_i$ . For  $i = i_s + 1$ , we draw  $u_1$  in  $D'_{i_s}$  sufficiently close to  $s$  such that  $su_1$  can be drawn as a straight-line segment that does not intersect any other edge. Analogously, for  $i = i_t + 1$ , we draw  $u_k$  in  $D'_{i_t}$  sufficiently close to  $t$  such that  $u_k t$  can be drawn as a straight-line segment that does not intersect any other edge. Recall that  $k \geq 3$  and hence edge  $u_1 u_k$  does not exist, which implies that all frames up to  $D_{n'+2}$  are planar. For each  $i \in [n' + 3, n' + k]$  we have that vertex  $u_j$ , with  $j = i - n' - 1$  appears, and the only visible vertices are  $u_{j-1}$ ,  $u_j$ , and  $u_k$ . In particular, the only edges to be drawn are  $u_{j-1} u_j$  and  $u_j u_k$  if  $j = k - 1$  (i.e., if  $i = n' + k$ ), hence they can be realized as crossing-free straight-line segments for any position of  $u_j$  (avoiding the three vertices being collinear).  $\square$

**Lemma 9.** *STORYPLAN parameterized by feedback edge set number admits a kernel of linear size.*

**Proof.** Let  $G$  be a graph with  $n$  vertices and let  $F$  be a feedback edge set of  $G$  of size  $\psi = \psi(G)$ . Let  $G^*$  be the graph obtained after applying exhaustively the reduction rules **RA** and **RB**. Graph  $G^*$  is a kernel of  $G$  by Lemma 8. Let  $F^* \subseteq F$  be the edges of  $F$  that belong to  $G^*$ . Observe that removing the edges of  $F^*$  from  $G^*$  yields a forest  $H^*$  with at most  $2\psi$

leaves in total. Indeed, any leaf of  $H^*$  is an endpoint of some edge in  $F^*$ , as otherwise it would not be part of  $G^*$  by **R.A**. Since  $H^*$  is a forest, its number of vertices with degree larger than two is at most the number of leaves minus two, i.e.,  $2\psi - 2$ . This also implies that the number of  $k$ -chains is at most  $4\psi - 3$ . By **R.B**,  $G^*$  (and hence  $H^*$ ) contains a  $k$ -chain only if  $k < 3$ . Consequently,  $H^*$  contains at most  $2(4\psi - 3)$  vertices of degree two and  $12\psi - 8$  vertices in total.  $\square$

To conclude the proof of Theorem 5, observe that computing a linear kernel  $G^*$  of  $G$ , i.e., applying exhaustively the reduction rules **R.A** and **R.B**, can be done in  $O(n + \psi)$  time. Afterwards, following the lines of the proof of Theorem 4, we can brute-force a solution for  $G^*$  (if any) in  $2^{O(\psi \log \psi)}$  time, and reinsert the missing  $O(n)$  vertices each in  $O(n)$  time (as detailed in Lemma 8).

### 3.4. Partial 3-trees

We initiate the study of STORYPLAN for graphs of bounded treewidth. A  $k$ -tree has a recursive definition: A complete graph with  $k + 1$  vertices is a  $k$ -tree; for any  $k$ -tree  $H$ , the graph obtained from  $H$  by adding a new vertex  $v$  connected to a clique  $C$  of  $H$  of size  $k$  is a  $k$ -tree;  $C$  is the *parent clique* of  $v$ . A *partial  $k$ -tree* is a subgraph of a  $k$ -tree and partial  $k$ -trees are exactly the graphs of treewidth at most  $k$ . Since 2-trees are planar, they admit a storyplan by Theorem 1. We prove that the same holds for partial 3-trees (which may be not planar).

**Theorem 6.** *Every partial 3-tree  $G$  with  $n$  vertices admits a storyplan, which can be computed in  $O(n)$  time.*

**Proof.** We shall assume that  $G$  is a (non-partial) 3-tree. Indeed, if  $G$  is a partial 3-tree, a supergraph of  $G$  that is a 3-tree always exists by definition (and can be computed in  $O(n)$  time). We view the initial  $K_4$  as constructed by adding a vertex connected to a 3-cycle, which we call the *initial 3-clique*.

We now construct a specific tree decomposition  $\mathcal{T}$  of  $G$  that will be used to compute its storyplan; refer to Fig. 12. For a definition of tree decomposition see [19]. The subgraph  $C_\mu$  induced by the vertices of each bag  $\mu$  of  $\mathcal{T}$  is the *subgraph associated with  $\mu$*  and it is a 4-clique for each bag  $\mu$  of  $\mathcal{T}$ , except for the root  $\rho$  of  $\mathcal{T}$  for which  $C_\rho$  is the initial 3-clique. A 3-clique of  $C_\mu$  is called *active* if it can appear in some subgraph  $C_\nu$  associated with a child  $\nu$  of  $\mu$ , and it is called *non-active* otherwise. The subgraph  $C_\mu$  contains four 3-cliques, by construction, three of them are active and one is non-active. In particular, the unique 3-clique in  $C_\rho$  is active. Each bag  $\mu$  of  $\mathcal{T}$  has one child  $\nu$  for each vertex  $v$  whose parent clique is an active 3-clique of  $\mu$ . The 4-clique  $C_\nu$  consists of the parent clique  $C$  of  $\nu$ , vertex  $\nu$ , and the edges connecting  $\nu$  to  $C$ ;  $C$  becomes non-active in  $\nu$ , while the other three 3-cliques are active. For each bag  $\mu$  distinct from  $\rho$ , we denote by  $v_\mu$  the vertex shared by the three active 3-cliques of  $C_\mu$ . We say that  $v_\mu$  is *associated with  $\mu$* . One easily verifies that  $\mathcal{T}$  is a tree decomposition. Also,  $\mathcal{T}$  has  $n - 2$  bags: the root and a bag for each vertex of  $G$  that is not in the initial 3-clique.

We now associate to each bag  $\mu$  a subgraph  $G_\mu$  of  $G$ . For the root  $\rho$ , the subgraph  $G_\rho$  is the initial 3-clique. For a bag  $\mu$  with parent  $\lambda$ , the subgraph  $G_\mu$  is obtained from  $G_\lambda$  by connecting  $v_\mu$  to the vertices of its parent clique.

**Property 1.** *Every graph  $G_\mu$  is an embedded planar 3-tree such that the active 3-cliques of  $C_\mu$  are internal faces of  $G_\mu$ .*

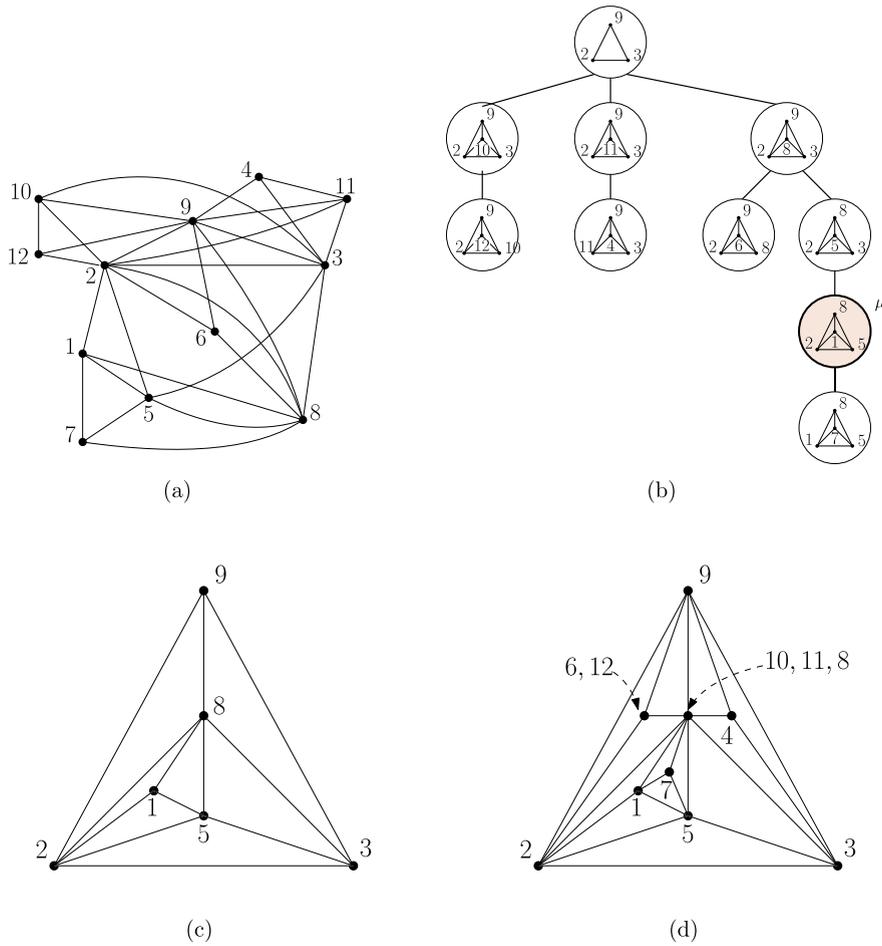
The proof of Property 1 is by induction on the length of the path from the root  $\rho$  to  $\mu$  in  $\mathcal{T}$ . The graph  $G_\rho$  consists of a 3-clique, which is the unique active 3-clique and which is both an internal and an external face. The graph  $G_\mu$  is obtained by adding the vertex  $v_\mu$  to  $G_\lambda$  and connecting it to its parent clique  $C$ , which is active in  $C_\lambda$ . By induction,  $C$  is an internal face of  $G_\lambda$  and therefore, by placing  $v_\mu$  inside this face, we obtain an embedded planar 3-tree such that the active faces of  $C_\mu$  are the three faces created by the addition of  $v_\mu$  inside  $C$ .

Let  $\mu \neq \rho$  be a bag of  $\mathcal{T}$ ; the next property follows from the definition of  $\mathcal{T}$ .

**Property 2.** *The neighbors of  $v_\mu$  distinct from those of its parent clique are all vertices associated with bags of the subtree of  $\mathcal{T}$  rooted at  $\mu$ .*

Let  $\rho = \mu_1, \mu_2, \dots, \mu_{n-2}$  be an order of the bags of  $\mathcal{T}$  according to a preorder visit of  $\mathcal{T}$ . To create a storyplan of  $G$ , we define an ordering  $\tau : v_1, v_2, \dots, v_n$  of the vertices of  $G$  such that  $v_1, v_2$ , and  $v_3$  are the vertices of the initial 3-clique, and each  $v_i$  with  $i > 3$  is the vertex associated with  $\mu_{i-2}$ .

Let  $G_i$  be the graph induced by the vertices that are visible at step  $i$ , for  $i \geq 3$ ; by Property 2 the graph  $G_i$  is a subgraph of  $G_{\mu_i}$  which, by Property 1 is an embedded planar 3-tree such that the three active 3-cliques of  $C_{\mu_i}$  are faces of  $G_{\mu_i}$ . To simplify the description we prove that there exists a storyplan  $\mathcal{S} = \langle \tau, \{D_i\}_{i \in [n]} \rangle$ , where each  $D_i$  is a drawing of  $G_{\mu_i}$ . This implies that there exists a storyplan where each  $D_i$  is a drawing of  $G_i$ . Let  $\mu_j$  be the parent of  $\mu_i$  in  $\mathcal{T}$ ; since the order  $\tau$  corresponds to a preorder of the bags of  $\mathcal{T}$ , we have  $j < i$ . Moreover, all bags  $\mu_k$  with  $j < k < i$ , if any, belong to the subtrees of  $\mu_j$  visited before  $\mu_i$  and for each such subtree  $\mathcal{T}'$  no other bag of  $\mathcal{T}'$  exists before  $\mu_j$  or after  $\mu_i$ . By Property 2 all the vertices associated with the bags  $\mu_k$  that belong to  $G_{\mu_{i-1}}$  do not have any neighbor after  $v_{\mu_i}$  and therefore they can be removed. The removal of these vertices transforms  $G_{\mu_{i-1}}$  into  $G_{\mu_j}$  (all the vertices associated with the bags  $\mu_k$  for  $j < k < i$  had been added to  $G_{\mu_j}$  that had never been changed). By Property 1 the active 3-cliques of  $G_{\mu_j}$  are faces of

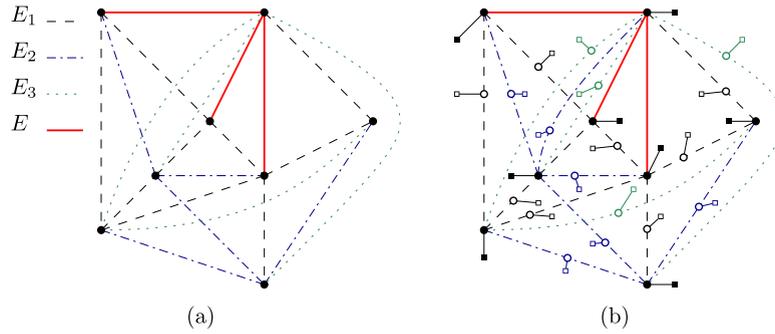


**Fig. 12.** (a) A 3-tree  $G$ . (b) The decomposition tree  $\mathcal{T}$  of  $G$ . (c) The subgraph  $G_\mu$  of  $G$  associated with bag  $\mu$ , highlighted in (b). (d) The graph  $G^*$ ; each vertex  $v$  of  $G^*$  is labeled with the vertices of  $G$  whose representative vertex is  $v$ .

$G_{\mu_j}$ . It follows that there exists a storyplan  $\mathcal{S} = \langle \tau, \{D_i\}_{i \in [n]} \rangle$  whose frames  $D_i$  are as follows.  $D_1$  is a planar drawing of a 3-clique; given  $D_{i-1}$  of  $G_{\mu_{i-1}}$ , a drawing  $D_i$  of  $G_{\mu_i}$  can be computed by removing all vertices associated with the bags  $\mu_k$  for  $j < k < i$ , and adding  $v_{\mu_i}$  inside a face of  $D_j$ .

It remains to prove that the above storyplan can be computed in  $O(n)$  time. If  $G$  is a partial 3-tree it can be augmented to a (non-partial) 3-tree in  $O(n)$  time [20]. The reverse of the order in which the vertices are added to construct  $G$  is called a *perfect elimination ordering* and can be computed in  $O(n)$  time [21]. Starting from this order we can compute  $\mathcal{T}$  in  $O(n)$  time. The order  $\tau$  can be computed in  $O(n)$  time by performing a preorder visit of  $\mathcal{T}$ . During this visit we also compute some information that are useful to compute efficiently all the drawings  $\{D_i\}_{i \in [n]}$ . First, we compute an embedded planar 3-tree  $G^*$  that is a supergraph of each  $G_i$  (refer to Fig. 12(d)). The idea to construct  $G^*$  is to keep only one representative, for each set of vertices whose parent cliques have the same set of representatives. We construct  $G^*$  starting with the initial 3-clique (which is associated with the root); each vertex of this 3-clique is its own representative. When we visit a bag  $\mu$ , we consider the three representatives of the vertices of the parent clique  $C$  of  $v_\mu$ ; these three vertices can be obtained in  $O(1)$  time by storing for each vertex its representative. If a vertex  $v$  connected to the representative vertices of  $C$  already exists, we do not modify  $G^*$  and store  $v$  as the representative vertex of  $v_\mu$ ; if no vertex has already been connected to the representative vertices of  $C$ , we add a new vertex  $v$  to  $G^*$  connecting it to the representative vertices of  $C$  and, also in this case, we store  $v$  as the representative vertex of  $v_\mu$ . Since both cases can be handled in  $O(1)$  time, the overall construction of  $G^*$  can be executed in  $O(n)$  time. Also,  $G^*$  has  $O(n)$  vertices and edges. Second, we store for each  $i$  (with  $i = 1, 2, \dots, n$ ) the list  $\ell_i$  of vertices that have to be removed when constructing  $D_i$  from  $D_{i-1}$ . These vertices are all the vertices associated with the bags of  $\mathcal{T}$  that are visited before  $\mu_i$  and after its parent  $\mu_j$  (this set of vertices may be empty) and they can be stored, in the reverse order of addition, while backtracking from  $\mu_{i-1}$  during the preorder visit.

Once we have  $G^*$  we can use any existing  $O(n)$ -time algorithm to compute a planar straight-line grid drawing  $D^*$  of  $G^*$  (for example the shift algorithm [22]). By the construction of  $G^*$ , each drawing  $\{D_i\}_{i \in [n]}$  is a sub-drawing of  $D^*$  and therefore we can compute each  $D_i$  by removing from  $D^*$  the vertices and edges that do not belong to  $G_i$ . In particular at



**Fig. 13.** Illustration for Theorem 7. (a) An instance  $G_1, G_2, G_3$  of SUNFLOWER SEFE; (b) The instance  $G$  constructed from  $G_1, G_2, G_3$ ; the subdivision vertices are circles with a white fill while the spectators are squares.

each step  $i$  we can use the list  $\ell_i$  to find the vertices to be removed from  $D_{i-1}$ ; these vertices are removed in reverse order of addition and when a vertex is removed we also remove the three edges that connect it to its parent clique. Since each vertex and each edge is added once and removed once, the set  $\{D_i\}_{i \in [n]}$  can be computed in  $O(n)$  time.  $\square$

#### 4. Complexity with fixed order

In this section we study the STORYPLANFIXEDORDER variant of STORYPLAN, defined below. This variant is closer to the setting studied in [6] and it models the case in which the order  $\tau$  of appearance of the vertices is given as part of the input.

**STORYPLANFIXEDORDER**

**Input:** Graph  $G = (V, E)$  with  $n$  vertices, total order  $\tau : V \rightarrow [n]$

**Question:** Does  $G$  admit a storyplan  $\mathcal{S} = \langle \tau, \{D_i\}_{i \in [n]} \rangle$ ?

We prove that STORYPLANFIXEDORDER is NP-complete by reducing from SUNFLOWER SEFE. Let  $G_1, \dots, G_k$  be  $k$  graphs on the same set  $V$  of vertices. A simultaneous embedding with fixed edges (SEFE) of  $G_1, \dots, G_k$  consists of  $k$  planar drawings  $\Gamma_1, \dots, \Gamma_k$  of  $G_1, \dots, G_k$ , respectively, such that each vertex is mapped to the same point in every drawing and each shared edge is represented by the same simple curve in all drawings sharing it. The SEFE problem asks whether  $k$  input graphs on the same set of vertices admit a SEFE, and it is NP-complete even when the pairwise intersection between any two input graphs is the same over all pairs of graphs [23,11]. This variant is called SUNFLOWER SEFE, and the result in [23,11] proves NP-completeness already when  $k = 3$ .

**Construction.** Refer to Fig. 13 for an example. Let  $G_1, G_2, G_3$  be an instance of SUNFLOWER SEFE. Let  $V$  be the common vertex set of the three graphs, let  $E$  be the common edge set, and let  $E_i$  be the exclusive edge set of  $G_i$  for  $i = 1, 2, 3$ . We construct an instance  $\langle G, \tau \rangle$  of STORYPLANFIXEDORDER as follows. Graph  $G$  contains all vertices in  $V$  and all edges in  $E$ . Also, for each edge  $e = uv$  in  $E_i$ , it contains a vertex  $w_e^i$ , called a subdivision vertex of  $E_i$ , and the edges  $uw_e^i$  and  $vw_e^i$  (i.e., it contains the edge  $e$  subdivided once). Moreover, for each vertex  $z$ , either a vertex in  $V$  or a subdivision vertex of an edge,  $G$  contains an additional vertex  $s_z$ , called the spectator of  $z$ , and the edge  $zs_z$ . To obtain the total order  $\tau$  we group the vertices of  $G$  in a set of blocks  $B_1, \dots, B_8$ , and we order the blocks by increasing index, while vertices within the same block can be ordered arbitrarily. We denote by  $\tau_i^-$  and  $\tau_i^+$  the position in  $\tau$  of the first and of the last vertex of  $B_i$ , respectively, for each  $i = 1, \dots, 8$ . Block  $B_1$  contains all vertices in  $V$ ; for  $i \in \{2, 4, 6\}$ , block  $B_i$  contains all subdivision vertices of  $E_{\frac{i}{2}}$ , while block  $B_{i+1}$  contains all spectators of the vertices in  $B_i$ ; finally  $B_8$  contains all spectators of the vertices in  $B_1$ .

**Theorem 7.** The STORYPLANFIXEDORDER problem is NP-complete.

**Proof.** Suppose first that  $G_1, G_2, G_3$  admit a SEFE  $\Gamma_1, \Gamma_2, \Gamma_3$ . Let  $\Gamma$  be the drawing obtained as follows: (i) let  $\Gamma'$  be the union of  $\Gamma_1, \Gamma_2, \Gamma_3$ ; (ii) let  $\Gamma''$  be the drawing obtained from  $\Gamma'$  by subdividing all exclusive edges in  $E_i$  for each  $i = 1, 2, 3$  (the subdivision vertex can be placed at any point along the curve representing the edge); (iii) finally, let  $\Gamma$  be the drawing obtained from  $\Gamma''$  by drawing for each vertex  $v$  its spectator  $s_v$  sufficiently close such that the edge  $vs_v$  can be realized as a straight-line segment that does not intersect any other edge. Then a storyplan  $\langle \tau, \{D_i\}_{i \in [n]} \rangle$  can be obtained by setting each drawing  $D_i$  equal to the subdrawing of  $\Gamma$  induced by the vertices visible at step  $i$ . By construction, the position of a vertex is the same over all frames that contain it. It remains to prove that each frame is planar.

- Each frame  $D_i$  with  $i \in [\tau_1^-, \tau_1^+]$  is a subdrawing of  $\Gamma[V]$ , which is planar because it contains only the edges that are part of the common intersection of  $G_1, G_2, G_3$ .
- Each frame  $D_i$  with  $i \in [\tau_2^-, \tau_3^+]$  is planar because it is a subdrawing of  $\Gamma_1$  in which each edge has been subdivided and the spectators of such subdivision vertices have been drawn without introducing crossings.
- Observe that in  $\tau_4^-$ , all spectators of the subdivision vertices of  $E_1$  disappear and hence the same holds for the subdivision vertices of  $E_1$ . It follows that each frame  $D_i$  with  $i \in [\tau_4^-, \tau_5^+]$  is planar because it is a subdrawing of  $\Gamma_2$  in which each edge has been subdivided and the spectators of such subdivision vertices have been drawn without introducing crossings. The same argument (with respect to  $E_3$ ) can be used to prove the planarity of each frame  $D_i$  with  $i \in [\tau_6^-, \tau_7^+]$ .
- In  $\tau_8^-$  all subdivision vertices of  $E_3$  disappear and again each frame  $D_i$  with  $i \in [\tau_8^-, \tau_8^+]$  is a subdrawing of  $\Gamma[V]$  where some spectators of the vertices in  $V$  have been drawn without introducing crossings.

Suppose now that  $\langle G, \tau \rangle$  admits a storyplan  $\langle \tau, \{D_i\}_{i \in [n]} \rangle$ . Then a SEFE  $\Gamma_1, \Gamma_2, \Gamma_3$  of  $\langle G_1, G_2, G_3 \rangle$  can be obtained as follows. Let  $\Gamma_1^* = \bigcup_{i=\tau_1^-}^{\tau_1^+} D_i$ , let  $\Gamma_2^* = \bigcup_{i=\tau_4^-}^{\tau_5^+} D_i$ , and let  $\Gamma_3^* = \bigcup_{i=\tau_6^-}^{\tau_7^+} D_i$ . Observe that, for each  $i = 1, 2, 3$ ,  $\Gamma_i^*$  is a planar drawing of  $G_i$  in which the exclusive edges of  $E_i$  are subdivided. Moreover, by construction it holds  $\Gamma_1^*[V] = \Gamma_2^*[V] = \Gamma_3^*[V]$ . It follows that  $\Gamma_i$  can be obtained by viewing each subdivision vertex as an inner point of the corresponding edge.

The membership to NP follows the lines of Theorem 3, simply ignoring the initial guess of a total order, which is instead given as part of the input.  $\square$

## 5. Discussion and open problems

Our work can stimulate further research based on several possible directions.

- It would be interesting to study further parameterizations of STORYPLAN. Is STORYPLAN parameterized by treewidth (pathwidth) in XP? In addition, we note that if the total order is fixed, then an FPT algorithm in the size of the largest frame (or the length of the longest lifespan) readily follows from the proof of Theorem 3.
- Conditions (iii) and (iv) of the definition of a storyplan can be replaced by the existence of a sequence of planar embeddings in which common subgraphs keep the same embedding. This is not true if we study more geometric versions of the problem, in which for instance edges are straight-line segments and/or vertices are restricted on an integer grid of fixed size (as in [6]).
- Condition (ii) of storyplan can be relaxed so to only allow specific crossing patterns [24,25], e.g., right-angle crossings or few crossings per edge.

## CRedit authorship contribution statement

**Carla Binucci:** Conceptualization, Formal analysis, Investigation, Writing – original draft, Writing – review & editing.  
**Emilio Di Giacomo:** Conceptualization, Formal analysis, Investigation, Writing – original draft, Writing – review & editing.  
**William J. Lenhart:** Conceptualization, Formal analysis, Investigation, Writing – original draft, Writing – review & editing.  
**Giuseppe Liotta:** Conceptualization, Formal analysis, Investigation, Writing – original draft, Writing – review & editing.  
**Fabrizio Montecchiani:** Conceptualization, Formal analysis, Investigation, Writing – original draft, Writing – review & editing.  
**Martin Nöllenburg:** Conceptualization, Formal analysis, Investigation, Writing – original draft, Writing – review & editing.  
**Antonios Symvonis:** Conceptualization, Formal analysis, Investigation, Writing – original draft, Writing – review & editing.

## Declaration of competing interest

Giuseppe Liotta, Carla Binucci, Emilio Di Giacomo, Fabrizio Montecchiani reports financial support was provided by Ministero dell'Istruzione, dell'Università e della Ricerca (MIUR). Martin Noellenburg reports financial support was provided by Vienna Science and Technology Fund. Work started at the 2022 Bertinoro Workshop on Graph Drawing.

## Data availability

No data was used for the research described in the article.

## References

- [1] C. Binucci, E. Di Giacomo, W.J. Lenhart, G. Liotta, F. Montecchiani, M. Nöllenburg, A. Symvonis, On the complexity of the storyplan problem, in: GD, in: Lecture Notes in Computer Science, vol. 13764, Springer, 2022, pp. 304–318.
- [2] N. Robertson, P.D. Seymour, Graph minors. I. Excluding a forest, J. Comb. Theory, Ser. B 35 (1) (1983) 39–61, [https://doi.org/10.1016/0095-8956\(83\)90079-5](https://doi.org/10.1016/0095-8956(83)90079-5).
- [3] E. Di Giacomo, W. Didimo, G. Liotta, F. Montecchiani, I.G. Tollis, Techniques for edge stratification of complex graph drawings, J. Vis. Lang. Comput. 25 (4) (2014) 533–543, <https://doi.org/10.1016/j.jvlc.2014.05.001>.

- [4] F. Beck, M. Burch, S. Diehl, D. Weiskopf, The state of the art in visualizing dynamic graphs, in: R. Borgo, R. Maciejewski, I. Viola (Eds.), EuroVis 2014, Eurographics Association, 2014.
- [5] C. Binucci, U. Brandes, G. Di Battista, W. Didimo, M. Gaertler, P. Palladino, M. Patrignani, A. Symvonis, K.A. Zweig, Drawing trees in a streaming model, *Inf. Process. Lett.* 112 (11) (2012) 418–422, <https://doi.org/10.1016/j.ipl.2012.02.011>.
- [6] M. Borrazzo, G. Da Lozzo, G. Di Battista, F. Frati, M. Patrignani, Graph stories in small area, *J. Graph Algorithms Appl.* 24 (3) (2020) 269–292, <https://doi.org/10.7155/jgaa.00530>.
- [7] G. Da Lozzo, I. Rutter, Planarity of streamed graphs, *Theor. Comput. Sci.* 799 (2019) 1–21, <https://doi.org/10.1016/j.tcs.2019.09.029>.
- [8] M. Skambath, T. Tantau, Offline drawing of dynamic trees: algorithmics and document integration, in: Y. Hu, M. Nöllenburg (Eds.), *Graph Drawing and Network Visualization, GD'16*, in: LNCS, vol. 9801, Springer, 2016, pp. 572–586.
- [9] C. Vehlou, F. Beck, D. Weiskopf, Visualizing dynamic hierarchies in graph sequences, *IEEE Trans. Vis. Comput. Graph.* 22 (10) (2016) 2343–2357, <https://doi.org/10.1109/TVCG.2015.2507595>.
- [10] G. Di Battista, W. Didimo, L. Grilli, F. Grosso, G. Ortali, M. Patrignani, A. Tappini, Small point-sets supporting graph stories, in: P. Angelini, R. von Hanxleden (Eds.), *Graph Drawing and Network Visualization, GD 2022*, Springer International Publishing, 2022, in press.
- [11] M. Schaefer, Toward a theory of planarity: Hanani-Tutte and planarity variants, *J. Graph Algorithms Appl.* 17 (4) (2013) 367–440, <https://doi.org/10.7155/jgaa.00298>.
- [12] N.G. Kinnersley, The vertex separation number of a graph equals its path-width, *Inf. Process. Lett.* 42 (6) (1992) 345–350, [https://doi.org/10.1016/0020-0190\(92\)90234-M](https://doi.org/10.1016/0020-0190(92)90234-M).
- [13] B. Monien, I.H. Sudborough, Min Cut is NP-complete for edge weighted trees, *Theor. Comput. Sci.* 58 (1988) 209–229, [https://doi.org/10.1016/0304-3975\(88\)90028-X](https://doi.org/10.1016/0304-3975(88)90028-X).
- [14] H.L. Bodlaender, T. Kloks, Efficient and constructive algorithms for the pathwidth and treewidth of graphs, *J. Algorithms* 21 (2) (1996) 358–402, <https://doi.org/10.1006/jagm.1996.0049>.
- [15] R. Impagliazzo, R. Paturi, On the complexity of k-SAT, *J. Comput. Syst. Sci.* 62 (2) (2001) 367–375, <https://doi.org/10.1006/jcss.2000.1727>.
- [16] R. Impagliazzo, R. Paturi, F. Zane, Which problems have strongly exponential complexity?, *J. Comput. Syst. Sci.* 63 (4) (2001) 512–530, <https://doi.org/10.1006/jcss.2001.1774>.
- [17] T.J. Schaefer, The complexity of satisfiability problems, in: R.J. Lipton, W.A. Burkhard, W.J. Savitch, E.P. Friedman, A.V. Aho (Eds.), *STOC 1978*, ACM, 1978, pp. 216–226.
- [18] J. Chen, I.A. Kanj, G. Xia, Improved upper bounds for vertex cover, *Theor. Comput. Sci.* 411 (40–42) (2010) 3736–3756, <https://doi.org/10.1016/j.tcs.2010.06.026>.
- [19] N. Robertson, P. Seymour, Graph minors. II. Algorithmic aspects of tree-width, *J. Algorithms* 7 (3) (1986) 309–322, [https://doi.org/10.1016/0196-6774\(86\)90023-4](https://doi.org/10.1016/0196-6774(86)90023-4).
- [20] J. Matoušek, R. Thomas, Algorithms finding tree-decompositions of graphs, *J. Algorithms* 12 (1) (1991) 1–22, [https://doi.org/10.1016/0196-6774\(91\)90020-Y](https://doi.org/10.1016/0196-6774(91)90020-Y).
- [21] D.J. Rose, R.E. Tarjan, G.S. Lueker, Algorithmic aspects of vertex elimination on graphs, *SIAM J. Comput.* 5 (2) (1976) 266–283, <https://doi.org/10.1137/0205021>.
- [22] M. Chrobak, T. Payne, A linear-time algorithm for drawing a planar graph on a grid, *Inf. Process. Lett.* 54 (4) (1995) 241–246, [https://doi.org/10.1016/0020-0190\(95\)00020-D](https://doi.org/10.1016/0020-0190(95)00020-D).
- [23] P. Angelini, G. Da Lozzo, D. Neuwirth, Advancements on SEFE and partitioned book embedding problems, *Theor. Comput. Sci.* 575 (2015) 71–89, <https://doi.org/10.1016/j.tcs.2014.11.016>.
- [24] W. Didimo, G. Liotta, F. Montecchiani, A survey on graph drawing beyond planarity, *ACM Comput. Surv.* 52 (1) (2019) 4:1–4:37, <https://doi.org/10.1145/3301281>.
- [25] S. Hong, T. Tokuyama (Eds.), *Beyond Planar Graphs, Communications of NII Shonan Meetings*, Springer, 2020.