The Parameterized Complexity Of Extending Stack Layouts

Thomas Depian
Algorithms and Complexity Group, TU Wien, Austria

Simon D. Fink
Algorithms and Complexity Group, TU Wien, Austria

Robert Ganian
Algorithms and Complexity Group, TU Wien, Austria

Martin Nöllenburg
Algorithms and Complexity Group, TU Wien, Austria

Abstract

An ℓ-page stack layout (also known as an ℓ-page book embedding) of a graph is a linear order of the vertex set together with a partition of the edge set into ℓ stacks (or pages), such that the endpoints of no two edges on the same stack alternate. We study the problem of extending a given partial ℓ-page stack layout into a complete one, which can be seen as a natural generalization of the classical NP-hard problem of computing a stack layout of an input graph from scratch. Given the inherent intractability of the problem, we focus on identifying tractable fragments through the refined lens of parameterized complexity analysis. Our results paint a detailed and surprisingly rich complexity-theoretic landscape of the problem which includes the identification of paraNP-hard, W[1]-hard and XP-tractable, as well as fixed-parameter tractable fragments of stack layout extension via a natural sequence of parameterizations.

2012 ACM Subject Classification Theory of computation → Fixed parameter tractability; Mathematics of computing → Graphs and surfaces

Keywords and phrases Stack Layout, Drawing Extension, Parameterized Complexity, Book Embedding

Digital Object Identifier 10.4230/LIPIcs.GD.2024.12

Related Version *Full Version*: `https://arxiv.org/abs/2409.02833` [17]

Funding All authors acknowledge support from the Vienna Science and Technology Fund (WWTF) [10.47379/ICT22029]. Robert Ganian and Thomas Depian furthermore acknowledge support from the Austrian Science Fund (FWF) [10.55776/Y1329].

1 Introduction

An ℓ-page stack layout (or ℓ-page book embedding) of a graph G consists, combinatorially speaking, of (i) a linear order $\prec$ of its vertex set $V(G)$ and (ii) a partition σ of its edge set $E(G)$ into $\ell \geq 1$ *(stack-)pages* such that for no two edges (with distinct endpoints) uv and wx with $u \prec v$ and $w \prec x$ that are assigned to the same page their endpoints alternate in $\prec$, i.e., we have $u \prec w \prec v \prec x$. When drawing a stack layout, the vertices are placed on a line called the *spine* in the order given by $\prec$ and the edges of each page are drawn as pairwise non-crossing arcs in a separate half-plane bounded by the spine, see Figure 1a. Stack layouts are a classic and well-studied topic in graph drawing and graph theory [6, 12, 30]. They have immediate applications in graph visualization [4, 25, 38] as well as in bioinformatics, VLSI design, and parallel computing [14, 27]; see also the overview by Dujmović and Wood [20].

The minimum number ℓ such that a given graph G admits an ℓ-page stack layout is known as the *stack number*, *page number*, or *book thickness* of G. While the graphs with stack number $\ell = 1$ are the outerplanar graphs, which can be recognized in linear time, the

32nd International Symposium on Graph Drawing and Network Visualization (GD 2024).
Editors: Stefan Felsner and Karsten Klein; Article No. 12; pp. 12:1–12:17
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

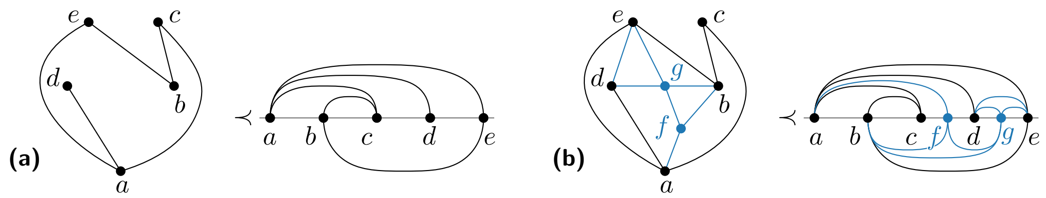

Figure 1 **(a)** A graph H and a two-page stack layout of it. In **(b)**, the graph H and its two-page stack layout are extended by the new vertices and edges marked in blue.

problem of computing the stack number is NP-complete in general. Indeed, the class of graphs with stack number $\ell \leq 2$ are precisely the subhamiltonian graphs (i.e., the subgraphs of planar Hamiltonian graphs) and recognizing them is NP-complete [6, 14, 39]. Computing the stack number is known to also remain NP-complete if the vertex order is provided as part of the input and $\ell = 4$ [36], and overcoming the intractability of these problems has been the target of several recent works in the field [10, 11, 24, 29]. Many other results on stack layouts are known – for instance, every planar graph has a 4-page stack layout and this bound is tight [5, 40]. For a comprehensive list of known upper and lower bounds for the stack number of different graph classes, we refer to the collection by Pupyrev [33].

In this paper, we take a new perspective on stack layouts, namely the perspective of drawing extensions. In drawing extension problems, the input consists of a graph G together with a partial drawing of G, i.e., a drawing of a subgraph H of G. The task is to insert the vertices and edges of G which are missing in H in such a way that a desired property of the drawing is maintained; see Figure 1b for an example. Such drawing extension problems occur, e.g., when visualizing dynamic graphs in a streaming setting, where additional vertices and edges arrive over time and need to be inserted into the existing partial drawing. Drawing extension problems have been investigated for many types of drawings in recent years – including planar drawings [1, 28, 31, 32], upward planar drawings [16], level planar drawings [13], 1-planar drawings [21, 22], and planar orthogonal drawings [2, 3, 9] – but until now, essentially nothing was known about the extension of stack layouts/book embeddings.

Since it is NP-complete to determine whether a graph admits an ℓ-page stack layout (even when ℓ is a small fixed integer), the extension problem for ℓ-page stack layouts is NP-complete as well – after all, setting H to be empty in the latter problem fully captures the former one. In fact, the extension setting can seemlessly also capture the previously studied NP-complete problem of computing an ℓ-stack layout with a prescribed vertex order [10, 11, 14, 36, 37]; indeed, this corresponds to the special case where $V(H) = V(G)$ and $E(H) = \emptyset$. Given the intractability of extending ℓ-page stack layouts in the classical complexity setting, we focus on identifying tractable fragments of the problem through the more refined lens of parameterized complexity analysis [15, 19], which considers both the input size of the graph and some additional parameter k of the instance[1].

Contributions. A natural parameter in any drawing extension problem is the size of the missing part of the graph, i.e., the missing number of vertices and/or edges. We start our investigation by showing that the Stack Layout Extension problem (SLE) for instances without any missing vertices, i.e., $V(G) = V(H)$, is fixed-parameter tractable when parameterized by the number of missing edges $|E(G) \setminus E(H)|$ (Section 3).

[1] We assume familiarity with the basic foundations of *parameterized complexity theory*, notably including the notions of *fixed-parameter tractability*, XP, W[1]-, and paraNP-hardness [15].

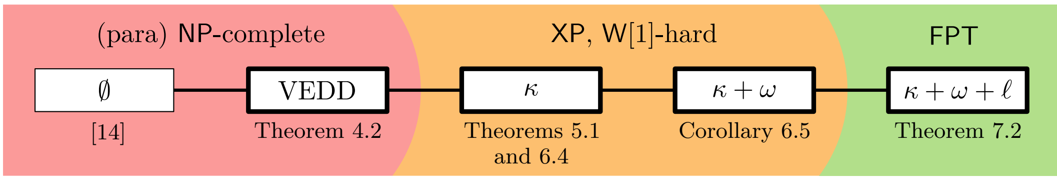

Figure 2 The complexity landscape of Stack Layout Extension. VEDD denotes the vertex+edge deletion distance, ω denotes the page width of the ℓ-page stack layout of H, and $\kappa = |V(G) \setminus V(H)| + |E(G) \setminus E(H)|$. Boxes outlined in bold represent new results that we show in the linked theorems and corollaries. The only result that is not depicted is Theorem 3.2.

The above result, however, only applies in the highly restrictive setting where no vertices are missing – generally, we would like to solve instances with missing vertices as well as edges. A parameterization that has been successfully used in this setting is the *vertex+edge deletion distance*, i.e., the number of vertex and edge deletion operations[2] required to obtain H from G. But while this parameter has yielded parameterized algorithms when extending, e.g., 1-planar drawings [21,22] and orthogonal planar drawings [9], we rule out any analogous result for SLE by establishing its NP-completeness even if H can be obtained from G by deleting only two vertices (Section 4). This means that more "restrictive" parameterizations are necessary to achieve tractability for the problem of extending ℓ-page stack layouts.

Since the missing vertices in our hardness reduction have a high degree, we then consider parameterizations by the combined number of missing vertices and edges $\kappa = |V(G) \setminus V(H)| + |E(G) \setminus E(H)|$. We show that SLE belongs to the class XP when parameterized by κ (Section 5) while being W[1]-hard (Section 6), which rules out the existence of a fixed-parameter tractable algorithm under standard complexity assumptions. The latter result holds even if we additionally bound the page width ω of the stack layout of H, which measures the maximum number of edges that are crossed on a single page by a line perpendicular to the spine [14]. On our quest towards a fixed-parameter tractable fragment of the problem, we thus need to include another restriction, namely the number ℓ of pages of the stack layout. So finally, when parameterizing SLE by the combined parameter $\kappa + \omega + \ell$, we show that it becomes fixed parameter tractable (Section 7). Our results are summarized in Figure 2.

Full proofs of statements marked by ★ can be found in the full version [17].

2 Preliminaries

We assume the reader to be familiar with standard graph terminology [18]. Throughout this paper, we assume standard graph representations, e.g., as double-linked adjacency list, that allow for efficient graph modifications. For two integers $p \leq q$ we denote with $[p, q]$ the set $\{p, p+1, \ldots, q\}$ and use $[p]_0$ and $[p]$ as abbreviations for $[0, p]$ and $[1, p]$, respectively. Let G be a graph that is, unless stated otherwise, simple and undirected, with vertex set $V(G)$ and edge set $E(G)$. For $X \subseteq V(G)$, we denote by $G[X]$ the subgraph of G induced on X.

Stack Layouts. For an integer $\ell \geq 1$, an *ℓ-page stack layout* of G is a tuple $\langle \prec_G, \sigma_G \rangle$ where $\prec_G$ is a linear order of $V(G)$ and $\sigma_G \colon E(G) \to [\ell]$ is a function that assigns each edge to a *page* $p \in [\ell]$ such that for each pair of edges u_1v_1 and u_2v_2 with $\sigma(e_1) = \sigma(e_2)$ it does not hold $u_1 \prec u_2 \prec v_1 \prec v_2$. For the remainder of the paper, we write $\prec$ and σ if the graph G is

[2] As usual, we assume that deleting a vertex automatically also deletes all of its incident edges.

clear from context. We call $\prec$ the *spine (order)* and σ the *page assignment*. Observe that we can interpret a stack layout as a drawing of G on different planar half-planes, one per page $p \in [\ell]$, each of which is bounded by the straight-line spine delimiting all half-planes. One fundamental property of a stack layout is its *page width* – denoted as $\omega(\langle \prec, \sigma \rangle)$ or simply ω if $\langle \prec, \sigma \rangle$ is clear from context – which is the maximum number of edges that are crossed on a single page by a line perpendicular to the spine [14]. The properties of stack layouts with small page width have been studied, e.g., by Stöhr [34, 35].

We say that two vertices u and v are *consecutive on the spine* if they occur consecutively in $\prec$. A vertex $u \in V(G)$ *sees* a vertex $v \in V(G)$ on a page $p \in [\ell]$ if there does not exist an edge $e = xy \in E(G)$ with $\sigma(e) = p$ and $x \prec u \prec y \prec v$ or $u \prec x \prec v \prec y$. Note that if u sees v, then v also sees u. For two vertices u and v which are consecutive in $\prec$, we refer to the segment on the spine between u and v as the *interval* between u and v, denoted as $[u, v]$.

Problem Statement. Let $H \subseteq G$ be a subgraph of a graph G. We say that $\langle \prec_G, \sigma_G \rangle$ is an *extension* of $\langle \prec_H, \sigma_H \rangle$ if $\sigma_H \subseteq \sigma_G$ and $\prec_H \subseteq \prec_G$. We now formalize our problem of interest:

Stack Layout Extension (SLE)
Given Integer $\ell \geq 1$, graph G, subgraph H of G, and ℓ-page stack layout $\langle \prec_H, \sigma_H \rangle$.
Question Does there exist an ℓ-page stack layout $\langle \prec_G, \sigma_G \rangle$ of G that extends $\langle \prec_H, \sigma_H \rangle$?

We remark that while SLE is defined as a decision problem for complexity-theoretic reasons, every algorithm presented in this article is constructive and can be trivially adapted to also output a layout $\langle \prec_G, \sigma_G \rangle$ as a witness (also called a *solution*) for positive instances. For an instance $\mathcal{I}$ of SLE, we use $|\mathcal{I}|$ as shorthand for $|V(G)| + |E(G)| + \ell$.

In line with the terminology previously used for drawing extension problems [21], we refer to the vertices and edges in $V(H) \cup E(H)$ as *old* and call all other vertices and edges of G *new*. Let V_{add} and E_{add} denote the sets of all new vertices and edges, respectively, and set $n_{\text{add}} := |V_{\text{add}}|$ and $m_{\text{add}} := |E_{\text{add}}|$. Furthermore, we denote with E^H_{add} the set of new edges incident to two old vertices, i.e., $E^H_{\text{add}} := \{e = uv \in E_{\text{add}} \mid u, v \in V(H)\}$. We consider the parameterized complexity of our extension problem by measuring how “incomplete” the provided partial solution is using the following natural parameters that have also been used in this setting before [7, 8, 21–23]: the *vertex+edge deletion distance*, which is $n_{\text{add}} + \left|E^H_{\text{add}}\right|$, and the total *number of missing vertices and edges*, i.e., $n_{\text{add}} + m_{\text{add}}$.

3 SLE With Only Missing Edges is FPT

We begin our investigation by first analyzing the special case where $V(G) = V(H)$, i.e., when only edges are missing from H. We recall that the problem remains NP-complete even in this setting, as it generalizes the problem of computing the stack number of a graph with a prescribed vertex order [10, 11, 14, 36, 37]. Furthermore, both of the aforementioned measures of the incompleteness of $\langle \prec_H, \sigma_H \rangle$ are the same and equal $m_{\text{add}} = \left|E^H_{\text{add}}\right|$. As a “warm-up” result, we show that in this setting SLE is fixed-parameter tractable parameterized by m_{add}.

Towards this, consider the set $S(e) \subseteq [\ell]$ of pages on which we could place a new edge e without introducing a crossing with edges from H; formally, $p \in S(e)$ if and only if $\langle \prec_H, \sigma_H \cup (e, p) \rangle$ is an ℓ-page stack layout of $H \cup \{e\}$. Intuitively, if $|S(e)|$ is large enough, then we are always able to find a “free” page to place e independent of the placement of the remaining new edges. Formally, one can easily show:

▶ **Lemma 3.1 (★).** *Let $\mathcal{I} = (\ell, G, H, \langle \prec, \sigma \rangle)$ be an instance of* SLE *with $V_{add} = \emptyset$ that contains an edge $e \in E_{add}$ with $|S(e)| \geq m_{add}$. The instance $\mathcal{I}' = (\ell, H, G', \langle \prec, \sigma \rangle)$ with $G' = G \setminus \{e\}$ is a positive instance if and only if $\mathcal{I}$ is a positive instance.*

With Lemma 3.1 in hand, we can establish the desired result:

▶ **Theorem 3.2 (★).** *Let $\mathcal{I} = (\ell, G, H, \langle \prec, \sigma \rangle)$ be an instance of* SLE *with $V_{add} = \emptyset$. We can find an ℓ-page stack layout of G that extends $\langle \prec, \sigma \rangle$ or report that none exists in $\mathcal{O}({m_{add}}^{m_{add}} \cdot |\mathcal{I}|)$ time.*

Proof sketch. We compute for a single edge $e \in E_{\text{add}}$ the set $S(e)$ in linear time by checking with which of the old edges e would cross. If $S(e) \geq m_{\text{add}}$, then following Lemma 3.1, we remove e from G. Overall, this takes $\mathcal{O}(m_{\text{add}} \cdot |\mathcal{I}|)$ time and results in a graph G' with $H \subseteq G' \subseteq G$. Furthermore, each remaining new edge $e' \in E(G') \setminus E(H)$ can be put in fewer than m_{add} different pages. Hence, we can brute-force over all the at most $\mathcal{O}({m_{\text{add}}}^{m_{\text{add}}})$ page assignments σ' that extend σ_H for all edges in $E(G') \setminus E(H)$, and for each such assignment we check in linear time whether no pair of edges $e', e'' \in E(G') \setminus E(H)$ cross each other. ◀

4 SLE With Two Missing Vertices is NP-complete

Adding only edges to a given linear layout is arguably quite restrictive. Therefore, we now lift this restriction and consider SLE in its full generality, i.e., also allow adding vertices. Somewhat surprisingly, as our first result in the general setting we show that SLE is NP-complete even if the task is to merely add two vertices, i.e., for $n_{\text{add}} = 2$ and $E^H_{\text{add}} = \emptyset$. This rules out not only fixed-parameter but also XP tractability when parameterizing by the vertex+edge deletion distance, and represents – to the best of our knowledge – the first example of a drawing extension problem with this behavior.

To establish the result, we devise a reduction from 3-Sat [26]. In our reduction, we insert two new vertices into a partial layout derived from the given formula, and use the page assignment of their incident edges to encode a truth assignment and validate that it satisfies all clauses. For this, we will need to restrict the positions of the new vertices to a certain range along the spine. In Section 4.1, we introduce the *fixation gadget* that ensures this. We also reuse this gadget in the reduction shown in Section 6. But first, we use it in this section to perform our reduction and prove NP-completeness in Section 4.2.

The graph H that we construct will have multi-edges to facilitate the presentation of the reduction. The procedure for removing multi-edges is detailed in the full version [17].

4.1 Restricting the Placement of New Vertices: The Fixation Gadget

The purpose of the so-called *fixation gadget* is to restrict the possible positions of new vertices to given intervals. As this gadget will also find applications outside this reduction, we describe in the following in detail its general construction for $F > 1$ new vertices $\mathcal{F} = \{f_1, \ldots, f_F\}$.

First, we introduce $3(F+1)$ new vertices $v_1, \ldots, v_{F+1}$, $b_1, \ldots, b_{F+1}$, and $a_1, \ldots, a_{F+1}$. We fix the spine order $\prec_H$ among these vertices to $b_1 \prec v_1 \prec a_1 \prec b_2 \prec v_2 \prec a_2 \prec \ldots \prec b_{F+1} \prec v_{F+1} \prec a_{F+1}$; see also Figure 3. Then, every new vertex f_i is made adjacent to v_i and v_{i+1} and we aim to allow these new edges to be placed only in a dedicated further page p_d. To achieve this, we first introduce for every $i \in [F+1]$ and every page $p \neq p_d$ an edge $e(b_i, a_i, p) = b_i a_i$ in H and set $\sigma(e(b_i, a_i, p)) = p$; see Figure 3. Furthermore, we also introduce the edges $b_i v_i$ and $v_i a_i$ and set $\sigma(b_i v_i) = \sigma(v_i a_i) = p_d$ for all $i \in [F+1]$. For every $i \in [F]$, we add the edge $v_i v_{i+1}$ and place it on the page p_d, i.e., we have $\sigma(v_i v_{i+1}) = p_d$ as in

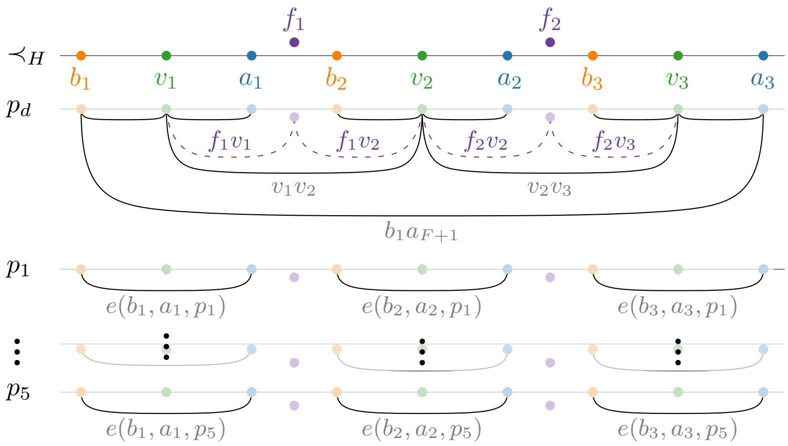

Figure 3 A fixation gadget for $F = 2$ with five other pages in the stack layout. We also highlight the intended position for f_1 and f_2 on the spine and the page assignment for their incident edges.

Figure 3. Finally, we also create the edge b_1a_{F+1} and set $\sigma(b_1a_{F+1}) = p_d$. To complete the construction of the fixation gadget, we add the new edges f_iv_i and f_iv_{i+1} for every $i \in [F]$ to G. Figure 3 shows an example of the fixation gadget for $F = 2$.

Next, we show that the fixation gadget forces f_i to lie between v_i and v_{i+1} on the spine and the edges f_iv_i and f_iv_{i+1} to be on the page p_d for every $i \in [F]$.

▶ **Lemma 4.1** (★). *Let $\mathcal{I} = (\ell, G, H, \langle\prec, \sigma\rangle)$ be an instance of* SLE *that contains a fixation gadget on F vertices $\{f_1, \ldots, f_F\}$. In any solution $\langle\prec_G, \sigma_G\rangle$ to $\mathcal{I}$ and for every $i \in [F]$, we have $v_i \prec f_i \prec v_{i+1}$ and $\sigma(f_iv_i) = \sigma(f_iv_{i+1}) = p_d$. Furthermore, the fixation gadget contributes $4F + 3$ vertices and $(\ell + 4)F + \ell + 2$ edges to the size of $\mathcal{I}$.*

Proof sketch. Towards establishing $v_i \prec f_i \prec v_{i+1}$, one can show that $f_i \prec v_i$ would prevent f_i from seeing v_{i+1} on any page: As $f_i \prec v_i$ implies $f_i \prec b_{i+1} \prec v_{i+1} \prec a_{i+1}$ and we have the edge $b_{i+1}a_{i+1}$ on any page except p_d, only visibility on page p_d would still be possible. However, the edges on the page p_d prevent visibility to v_{i+1} for any spine position left of v_i. By symmetric arguments, we can obtain that $v_{i+1} \prec f_i$ would prevent v_i from seeing f_i. Using again the fact that we have the edge b_ia_i on any page except p_d, in concert with the relation $v_i \prec f_i \prec v_{i+1}$ shown above and the edges v_ia_i and $b_{i+1}v_{i+1}$ on the page p_d, one can deduce that $\sigma(f_iv_i) = \sigma(f_iv_{i+1}) = p_d$ must hold. Finally, the bound on the size of the gadget can be obtained by a close analysis of the construction. ◀

Lemma 4.1 tells us that we can restrict the feasible positions for f_i to a pre-defined set of consecutive intervals by choosing suitable positions for v_i and v_{i+1} in the spine order $\prec_H$. As the fixation gadget requires an additional page p_d, we must ensure that the existence of the (otherwise mostly empty) page p_d does not violate the semantics of our reductions. In particular, we will (have to) ensure that our full constructions satisfy the following property.

▶ **Property 1.** *Let $\mathcal{I} = (\ell, G, H, \langle\prec, \sigma\rangle)$ be an instance of* SLE *that contains a fixation gadget on F vertices $\{f_1, \ldots, f_F\}$. In any solution $\langle\prec_G, \sigma_G\rangle$ to $\mathcal{I}$ and for every new edge $e \in E_{add}$ with $\sigma(e) = p_d$, we have $e \in \{f_iv_i, f_iv_{i+1} \mid i \in [F]\}$.*

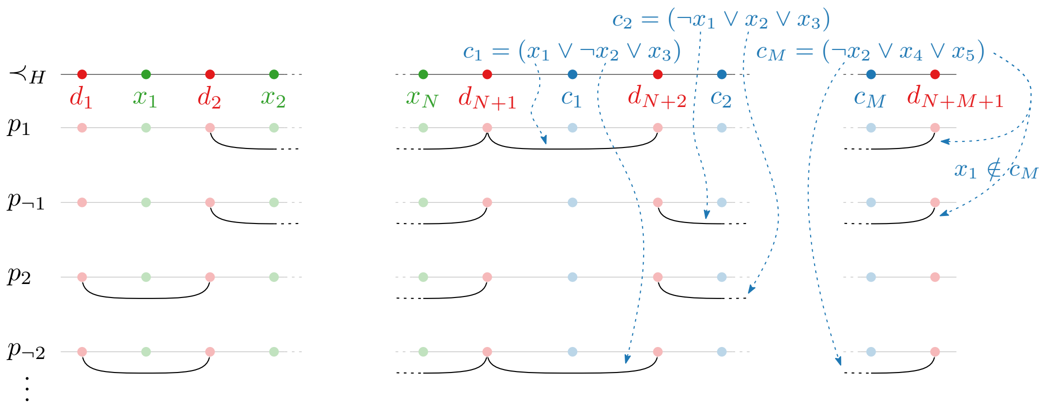

Figure 4 An overview of the created vertices and edges in our reduction. Green vertices represent variables, blue vertices clauses, and red vertices the dummy vertices d_q. Furthermore, we visualize some of the edges in H that are created for the variable-vertices (left) and clause-vertices (middle and right) to block visibility on the respective pages. If an edge is created due to the (non-)existence of a literal in the clauses c_1, c_2, or c_M it is indicated via a blue arc.

4.2 The Complete Reduction

Let $\varphi = (\mathcal{X}, \mathcal{C})$ be an instance of 3-SAT consisting of N variables $\mathcal{X} = \{x_1, \dots, x_N\}$ and M clauses $\mathcal{C} = \{c_1, \dots, c_M\}$, each consisting of three different and pairwise non-complementary literals. The reduction constructs an instance $\mathcal{I} = (\ell, G, H, \langle \prec, \sigma \rangle)$ of SLE which represents each variable x_i and each clause c_j of φ, respectively, by a corresponding vertex in H. The linear order $\prec_H$ has the form $x_1 \prec x_2 \prec \ldots \prec x_N \prec c_1 \prec \ldots \prec c_M$; see Figure 4. Furthermore, $\mathcal{I}$ contains two new vertices s and v. The vertex s is adjacent to all variable-vertices and the construction will ensure that the page assignment for its incident edges represents, i.e., ***s***elects, a truth assignment Γ for φ. The vertex v is adjacent to all clause-vertices, and its purpose is to ***v***erify that the truth assignment satisfies all clauses. For the following description of how this is achieved, we assume $s \prec v \prec x_1$ as we will use a fixation gadget to ensure that every solution $\langle \prec, \sigma \rangle$ of $\mathcal{I}$ has this property.

To each variable x_i, we associate two pages p_i and $p_{\neg i}$ corresponding to its possible truth states. We ensure that s can see each variable-vertex only on its associated pages using edges incident to dummy vertices d_q with $q \in [N + M + 1]$. These dummy vertices are distributed as in Figure 4. Hence, a page assignment for the edges incident to s induces a truth assignment. Similar edges also ensure that v can see a clause-vertex c_j only on the pages that are associated to the negation of the literals the clause c_j is composed of, see the blue arcs in Figure 4 for an illustration. We defer the full construction to the full version [17].

We now ensure that $s \prec v \prec x_1$ holds in every solution of $\mathcal{I}$ by using a fixation gadget on two vertices, i.e., for $F = 2$. In particular, we set $a_3 \prec d_1$, i.e., we place the fixation gadget at the beginning of the spine, and identify $s = f_1$ and $v = f_2$. The spine order $\prec_H$ is then the transitive closure of all the partial orders stated until now; see Figure 4. Finally, we add the edge $d_1 d_{N+M+1}$ and set $\sigma(d_1 d_{N+M+1}) = p_d$ to ensure that our construction has Property 1.

Regarding the correctness of our reduction, we make the following observation. If an induced truth assignment does not satisfy a clause c_j, then it must use the pages associated to the negated literals of c_j. Thus, the new edge vc_j will cross another edge no matter which page we use. However, if a clause c_j is satisfied, we can find a page for the edge vc_j that does

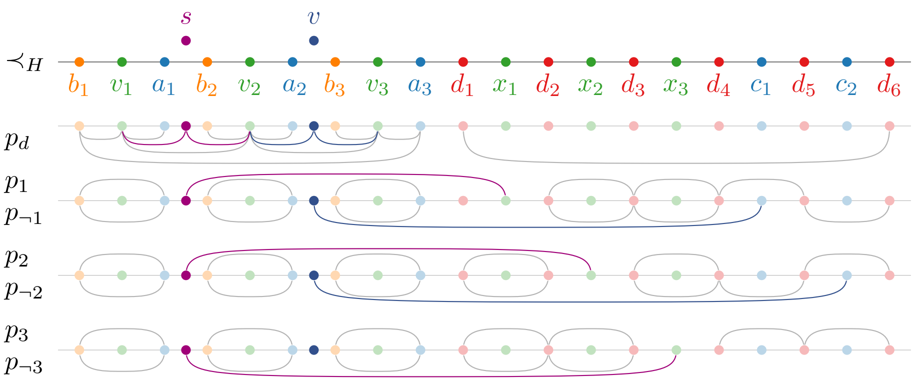

Figure 5 An example of our reduction for the formula φ consisting of the clauses $c_1 = (x_1 \vee \neg x_2 \vee x_3)$ and $c_2 = (\neg x_1 \vee x_2 \vee x_3)$. The extension indicated in saturated colors induces the truth assignment $\Gamma(x_1) = \Gamma(x_2) = 1$ and $\Gamma(x_3) = 0$, which satisfies φ.

not introduce a crossing: the page associated to the negation of the literal that satisfies c_j. Consequently, if φ is satisfiable, then there exists an extension $\langle \prec_G, \sigma_G \rangle$. Similarly, the page assignment of an extension $\langle \prec_G, \sigma_G \rangle$ induces a truth assignment Γ that satisfies φ. An intuitive example of the reduction is provided in Figure 5, and we obtain the following theorem.

▶ **Theorem 4.2 (★).** SLE *is* NP*-complete even if we have just two new vertices and* $E^H_{add} = \emptyset$*.*

Finally, we want to remark that Theorem 4.2 is tight in the sense that SLE with only one new vertex v and $E^H_{\text{add}} = \emptyset$ can be solved in polynomial time. To that end, we can branch over all $\mathcal{O}(n)$ possible spine positions where v can be placed. For each of these, the observation that edges incident to the same vertex can never cross each other allows us to greedily assign a new edge uv to the first page p where v can see u. Recall that we only add one new vertex v. Hence, u is an old vertex whose spine position is known. Clearly, an extension exists if and only if there exists a spine position for v such that our greedy page assignment can find a page for all new edges.

▶ Remark 4.3. Let $\mathcal{I} = (\ell, G, H, \langle \prec, \sigma \rangle)$ be an instance of SLE with $n_{\text{add}} = 1$ and $E^H_{\text{add}} = \emptyset$. We can find an ℓ-page stack layout of G that extends $\langle \prec, \sigma \rangle$ or report that none exists in $\mathcal{O}(n \cdot m_{\text{add}} \cdot |\mathcal{I}|)$ time.

5 SLE Parameterized by Missing Vertices and Edges is in XP

In the light of Theorem 4.2, which excludes the use of the vertex+edge deletion distance as a pathway to tractability, we consider parameterizing by the total number of missing vertices and edges $\kappa := n_{\text{add}} + m_{\text{add}}$. As our first result in this direction, we show that parameterizing SLE by κ makes it XP-tractable. To this end, we combine a branching-procedure with the fixed parameter algorithm for the special case obtained in Theorem 3.2.

▶ **Theorem 5.1.** *Let* $\mathcal{I} = (\ell, G, H, \langle \prec, \sigma \rangle)$ *be an instance of* SLE*. We can find an* ℓ*-page stack layout of* G *that extends* $\langle \prec, \sigma \rangle$ *or report that none exists in* $\mathcal{O}(|\mathcal{I}|^{n_{add}+1} {m_{add}}^{m_{add}})$ *time.*

Proof. We branch over the possible assignments of new vertices to the intervals in $\prec_H$. As a solution could assign multiple vertices to the same interval, we also branch over the order in which the new vertices will appear in the spine order $\prec_G$. Observe that $\prec_H$ induces

$|V(H)|+1$ different intervals, out of which we have to choose n_{add} with repetition. Together with the possible orders of the new vertices, we can bound the number of branches by $n_{\text{add}}! \cdot \binom{|V(H)|+n_{\text{add}}}{n_{\text{add}}}$. We can simplify this expression to

$$\frac{n_{\text{add}}! \cdot (|V(H)|+n_{\text{add}})!}{n_{\text{add}}! \cdot ((|V(H)|+n_{\text{add}})-n_{\text{add}})!} = \frac{(|V(H)|+n_{\text{add}})!}{|V(H)|!} = \Pi_{i=1}^{n_{\text{add}}}(|V(H)|+i) = \mathcal{O}(|\mathcal{I}|^{n_{\text{add}}}).$$

In each branch, the spine order $\prec_G$ is fixed and extends $\prec_H$. Hence, it only remains to check whether $\prec_G$ allows for a valid page assignment σ_G. As each branch corresponds to an instance of SLE where only edges are missing, we use Theorem 3.2 to check in $\mathcal{O}({m_{\text{add}}}^{m_{\text{add}}} \cdot |\mathcal{I}|)$ time whether such an assignment σ_G exists. The overall running time now follows readily. ◀

The running time stated in Theorem 5.1 not only proves that SLE is in XP when parameterized by κ, but also FPT when parameterized by m_{add} for constant n_{add}. However, common complexity assumptions rule out an efficient algorithm parameterized by κ, as we show next.

6 SLE Parameterized by Missing Vertices and Edges is W[1]-hard

In this section, we show that SLE parameterized by the number κ of missing vertices and edges is W[1]-hard. To show W[1]-hardness, we reduce from the Multi-colored Clique (McC) problem. Here, we are given a graph G_C, an integer $k > 0$, and a partition of $V(G_C)$ into k independent subsets $V_1, \ldots, V_k$, and ask whether there exists a *colorful k-clique* $\mathcal{C} \subseteq V(G_C)$ in G_C, i.e., a clique on k vertices that contains exactly one vertex of every set V_i, $i \in [k]$. It is known that McC is W[1]-hard when parameterized by k [15]. In the following, we will use Greek letters for the indices of the partition and denote with n_α the number of vertices in V_α, i.e., $n_\alpha = |V_\alpha|$. Observe that $\sum_{\alpha \in [k]} n_\alpha = N$ with $N = |V(G_C)|$. As we can interpret the partitioning of the vertices into $V_1, \ldots, V_k$ as assigning to them one of k colors, we will call a vertex v_α^i with $\alpha \in [k]$ and $i \in [n_\alpha]$ a vertex with *color* α. Our construction will heavily use the notion of a successor and predecessor of a vertex in a given spine order $\prec$. For a vertex u, the function $\text{succ}(\prec,\ u)$ returns the *successor* of u in the spine order $\prec$, i.e., the consecutive vertex in $\prec$ after u. Note that $\text{succ}(\prec,\ u)$ is undefined if there is no vertex $v \in V(G)$ with $u \prec v$. We write $\text{succ}(u)$ if $\prec$ is clear from context. The *predecessor* function $\text{pred}(\prec,\ u)$ is defined analogously. In the following, we first give an overview of and intuition behind our reduction in Section 6.1, before we show its correctness in Section 6.2. Note that the full details of the construction can be found in the full version [17]. Furthermore, as in the reduction from Section 4, we will allow multi-edges in the graph H to facilitate the presentation of the reduction. The procedure for removing multi-edges by distributing the individual edges over auxiliary vertices is also detailed in the full version [17].

6.1 An Overview of the Construction

Let $(G_C, k, (V_1, \ldots, V_k))$ be an instance of McC. We will construct an SLE instance $(\ell, G, H, \langle \prec, \sigma \rangle)$ parameterized by κ that will fulfill two crucial properties to ensure its correctness. While, at the time of stating the property, our construction might not yet fulfill it, we show in Section 6.2 that in the end it indeed has the desired properties.

First, we define the base layout of our reduction. In the base layout, we create the $N+2k+3$ vertices $\{u_\alpha^j \mid \alpha \in [k],\ j \in [n_\alpha+1]_0\} \cup \{u_0^0, u_\perp^0, u_\perp^1\}$ in H. Note that for each *original* vertex $v_\alpha^i \in V(G_C)$, we have a *copy* u_α^i. We will refer to the vertices u_0^0, $u_\perp^0$, and $u_\perp^1$ as *dummy vertices* and set, for ease of notation, $\perp = k+1$ and $n_\perp = 1$. The vertices are placed on the spine based on their color α and index i; see Figure 6. Finally, observe that $\text{succ}(u_\alpha^i) = u_\alpha^{i+1}$ for every $v_\alpha^i \in V(G_C)$. Furthermore, every vertex $v_\alpha^i \in V(G_C)$ *induces* the interval $[u_\alpha^i, u_\alpha^{i+1}]$

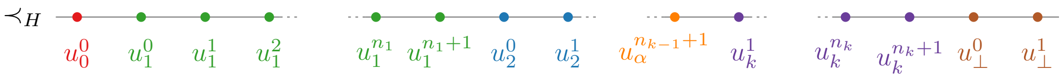

Figure 6 The base layout of our reduction. We use colors to additionally differentiate vertices that originate from different vertex sets V_α, for $\alpha \in [k]$, and the dummy vertices u_0^0, $u_\perp^0$ and $u_\perp^1$.

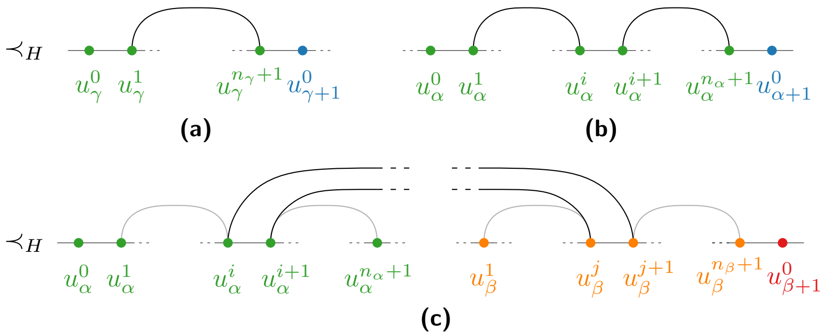

Figure 7 Edges of H that model the adjacency given by the edge $e = v_\alpha^i v_\beta^j \in E(G_C)$. All of these edges are placed on the page p_e. Intuitively, we span the intervals induced **(a)** by all vertices for each color $\gamma \in [k] \setminus \{\alpha, \beta\}$ and **(b)** by vertices of the colors α and β that are not incident to e, here visualized for the color α. **(c)** Furthermore, we create a tunnel that connects $\curlyvee(v_\alpha^i)$ with $\curlyvee(v_\beta^j)$. The gray edges in **(c)** are from **(a)** and **(b)**.

in $\prec_H$, which we denote with $\curlyvee(v_\alpha^i)$. The equivalence between the two problems will be obtained by adding a k-clique to G that consists of the k new vertices $\mathcal{X} = \{x_1, \ldots, x_k\}$. Placing $x_\alpha \in \mathcal{X}$ in $\curlyvee(v_\beta^i)$ indicates that v_β^i will be part of the colorful k-clique in G_C, i.e., we will have the equivalence $u_\alpha^i \prec x_\alpha \prec \mathrm{succ}(\prec_H, u_\alpha^i) \overset{\text{p. d.}}{\Longleftrightarrow} x_\alpha$ is placed in $\curlyvee(v_\alpha^i) \iff v_\alpha^i \in \mathcal{C}$ between a solution $\langle \prec_G, \sigma_G \rangle$ to SLE and a solution $\mathcal{C}$ to MCC. To guarantee that $\mathcal{C}$ is colorful, i.e., contains exactly one vertex from each color, we will ensure the following property with our construction.

▶ **Property 2.** *In a solution $\langle \prec, \sigma \rangle$ to* SLE *we have $u_\alpha^0 \prec x_\alpha \prec u_{\alpha+1}^0$ for every $\alpha \in [k]$.*

To establish the correctness of our reduction, we have to ensure two things. First, we have to model the adjacencies in G_C. In particular, two new vertices x_α and x_β, with $\alpha < \beta$, should only be placed in intervals induced by vertices adjacent in G_C. We enforce this by adding for every edge $e = v_\alpha^i v_\beta^j \in E(G_C)$ a page p_e. On this page p_e, we create the following edges in H; see also Figure 7 for a visualization. Firstly, we create for every color $\gamma \in [k] \setminus \{\alpha, \beta\}$ an edge that spans exactly the intervals induced by vertices of color γ, thereby intuitively blocking visibility to any interval induced by a vertex of a color different to α and β; see Figure 7a. Secondly, we create up to two edges that span all intervals induced by vertices of color α except $\curlyvee(v_\alpha^i)$; see Figure 7b. We do so similarly for color β. These edges in concert with a *tunnel* that we create on page p_e, see Figure 7c, allow us to place the edge $x_\alpha x_\beta \in E(G)$ in the page p_e if and only if x_α is placed in $\curlyvee(v_\alpha^i)$ and x_β in $\curlyvee(v_\beta^j)$. More formally, our construction will ensure the following property.

▶ **Property 3.** *Let $\langle \prec, \sigma \rangle$ be a solution to an instance of* SLE *that fulfills Property 2 and for which we have $e = v_\alpha^i v_\beta^j \in E(G_C)$, $1 \leq \alpha < \beta \leq k$, and $x_\alpha, x_\beta \in \mathcal{X}$. If $\sigma(x_\alpha x_\beta) = p_e$ then x_α is in $\curlyvee(v_\alpha^i)$ and x_β is in $\curlyvee(v_\beta^j)$.*

Second, we have to ensure that we select exactly one vertex $v_\alpha^i \in V_\alpha$ for every color $\alpha \in [k]$. In particular, the new vertex x_α should only be placed in intervals that are induced by vertices from V_α. To this end, we modify H to include a fixation gadget on $F = k$ vertices by re-using some vertices of the base layout. Most importantly, we identify $v_\alpha = u_\alpha^0$ for every $\alpha \in [k+1]$ and $f_\alpha = x_\alpha$ for every $\alpha \in [k]$; see the full version [17] for details. As the whole base layout thereby forms the fixation gadget, our construction trivially satisfies Property 1.

6.2 Bringing It Together: Showing Correctness of the Reduction

With the overview of the construction and the intuition behind the reduction settled, we now proceed to show its correctness in Theorem 6.4. In the proof, we make use of Properties 2 and 3. Therefore, on our path to obtain Theorem 6.4, we first have to show that our construction fulfills them. Recall that Property 2 is defined as follows.

▶ **Property 2.** *In a solution $\langle \prec, \sigma \rangle$ to* SLE *we have $u_\alpha^0 \prec x_\alpha \prec u_{\alpha+1}^0$ for every $\alpha \in [k]$.*

When incorporating the fixation gadget on $F = k$ vertices in our construction, we identified $v_\alpha = u_\alpha^0$ and $f_\alpha = x_\alpha$ for every $\alpha \in [k]$. Similarly, we identified $v_{F+1} = u_{k+1}^0$. The fixation gadget now guarantees thanks to Lemma 4.1 that we have $v_\alpha \prec_G f_\alpha \prec_G v_{\alpha+1}$, i.e., $u_\alpha^0 \prec x_\alpha \prec u_{\alpha+1}^0$, in any solution $\langle \prec_G, \sigma_G \rangle$. Hence, we can observe the following.

▶ **Observation 6.1.** *Our instance $\mathcal{I}$ of* SLE *fulfills Property 2.*

Recall that Lemma 4.1 furthermore tells us that we have in any solution $\langle \prec_G, \sigma_G \rangle$ the page assignment $\sigma(x_\alpha u_\alpha^0) = \sigma(x_\alpha u_{\alpha+1}^0) = p_d$ for every $\alpha \in [k]$. As we have by Property 2 $u_\alpha^0 \prec x_\alpha \prec u_{\alpha+1}^0$ and furthermore by the construction of the fixation gadget $\sigma_H(\text{pred}(u_\alpha^0)u_\alpha^0) = \sigma_H(u_\alpha^0\text{succ}(u_\alpha^0)) = p_d$ for every $\alpha \in [k]$, we cannot have in $\langle \prec_G, \sigma_G \rangle$ $u_\alpha^0 \prec x_\alpha \prec \text{succ}(\prec_H,\ u_\alpha^0)$ or $\text{pred}(\prec_H,\ u_{\alpha+1}^0) \prec x_\alpha \prec u_{\alpha+1}^0$, as this would introduce a crossing on page p_d. As we have in $\prec_H$ the equality $\text{succ}(u_\alpha^0) = u_\alpha^1$ and $\text{pred}(u_{\alpha+1}^0) = u_\alpha^{n_\alpha+1}$ for every $\alpha \in [k]$, we can strengthen Property 2 and obtain the following.

▶ **Corollary 6.2.** *In a solution $\langle \prec, \sigma \rangle$ to* SLE *we have $u_\alpha^1 \prec x_\alpha \prec u_\alpha^{n_\alpha+1}$ for every $\alpha \in [k]$.*

Finally, we now show that our construction fulfills Property 3, which was defined as follows.

▶ **Property 3.** *Let $\langle \prec, \sigma \rangle$ be a solution to an instance of* SLE *that fulfills Property 2 and for which we have $e = v_\alpha^i v_\beta^j \in E(G_C)$, $1 \leq \alpha < \beta \leq k$, and $x_\alpha, x_\beta \in \mathcal{X}$. If $\sigma(x_\alpha x_\beta) = p_e$ then x_α is in $\curlyvee(v_\alpha^i)$ and x_β is in $\curlyvee(v_\beta^j)$.*

▶ **Lemma 6.3.** *Our instance $\mathcal{I}$ of* SLE *fulfills Property 3.*

Proof. First, recall that we made Observation 6.1, i.e., our construction fulfills Property 2. Let $\langle \prec_G, \sigma_G \rangle$ be a solution to SLE with $\sigma(x_\alpha x_\beta) = p_e$, for $e = v_\alpha^i v_\beta^j \in E(G_C)$, $1 \leq \alpha < \beta \leq k$. Corollary 6.2 tells us that $u_\alpha^1 \prec x_\alpha \prec u_\alpha^{n_\alpha+1}$ and $u_\beta^1 \prec x_\beta \prec u_\beta^{n_\beta+1}$ holds. Corollary 6.2 also holds for any new vertices x_γ and x_δ with $\gamma, \delta \in [k] \setminus \{\alpha, \beta\}$ and $\gamma \neq \delta$. Furthermore, we have the edges $u_\gamma^1 u_\gamma^{n_\gamma+1}$ and $u_\delta^1 u_\delta^{n_\delta+1}$ on page p_e. Hence, all new edges on page p_e must be among new vertices placed in intervals induced by vertices of color α or β.

Now assume that we have $u_\alpha^1 \preceq x_\alpha \preceq u_\alpha^i$. Using $\sigma_H(u_\alpha^1 u_\alpha^i) = p_e$ together with $u_\alpha^1 \preceq x_\alpha \preceq u_\alpha^i \prec x_\beta$, we derive that $u_\alpha^1 \preceq x_\alpha \preceq u_\alpha^i$ results in a crossing on page p_e. Hence, $u_\alpha^1 \preceq x_\alpha \preceq u_\alpha^i$ cannot hold. Now assume that we have $u_\alpha^{i+1} \preceq x_\alpha \preceq u_\alpha^{n_\alpha+1}$. From $\sigma_H(u_\alpha^{i+1} u_\alpha^{n_\alpha+1}) = p_e$ and $u_\alpha^{i+1} \preceq x_\alpha \preceq u_\alpha^{n_\alpha+1} \prec x_\beta$ we get that $u_\alpha^{i+1} \preceq x_\alpha \preceq u_\alpha^{n_\alpha+1}$ results in a crossing on page p_e. Hence, $u_\alpha^{i+1} \preceq x_\alpha \preceq u_\alpha^{n_\alpha+1}$ cannot hold. Since we can exclude $u_\alpha^1 \preceq x_\alpha \preceq u_\alpha^i$ and $u_\alpha^{i+1} \preceq x_\alpha \prec u_\alpha^{n_\alpha+1}$ by the construction of the tunnel on page p_e, we can derive that x_α must be placed in $\curlyvee(v_\alpha^i)$. As similar arguments can be made for x_β, we can conclude that we get a crossing on page p_e unless x_α is placed in $\curlyvee(v_\alpha^i)$ and x_β in $\curlyvee(v_\beta^i)$. ◀

▶ **Theorem 6.4 (★).** SLE *parameterized by the number* κ *of missing vertices and edges is* W[1]*-hard.*

Proof sketch. Let $(G_C, k, (V_1, \ldots, V_k))$ be an instance of McC with $N = |V(G_C)|$ and $M = |E(G_C)|$ and let $\mathcal{I} = (\ell, G, H, \langle\prec, \sigma\rangle)$ be the instance of SLE parameterized by the number κ of missing vertices and edges created by our construction described above. Closer analysis reveals that the size of $\mathcal{I}$ is bounded by $\mathcal{O}(N + Mk + k^2)$, and we have $\kappa = 3k + \binom{k}{2}$ as $n_{\text{add}} = k$ and $m_{\text{add}} = \binom{k}{2} + 2k$; recall that the fixation gadget contributes $2k$ new edges.

Towards arguing correctness, assume that $(G_C, k, (V_1, \ldots, V_k))$ contains a colorful k-clique $\mathcal{C}$. We construct a solution to $\mathcal{I}$ by, for every new vertex $x_\alpha \in \mathcal{X}$, considering the vertex $v^i_\alpha \in \mathcal{C}$ and placing x_α immediately to the right of the copy u^i_α of v^i_α in H. The fact that $\mathcal{C}$ is a clique then guarantees that, for each edge $e \in E(G_C[\mathcal{C}])$, there exists the page p_e in which the corresponding edge $e' \in E(G[\mathcal{X}])$ can be placed in. For the remaining edges from the fixation gadget, we can use the page assignment from Lemma 4.1.

For the converse (and more involved) direction, assume that SLE admits a solution $\langle\prec_G, \sigma_G\rangle$. By Property 2, we have that each $x_\alpha \in V_{\text{add}}$ must be placed between u^0_α and $u^0_{\alpha+1}$. Moreover, our construction together with the page assignment forced by Lemma 4.1 guarantees that x_α is placed between precisely one pair of consecutive vertices $u^{i_\alpha}_\alpha$ and $u^{i_\alpha+1}_\alpha$, for some $i_\alpha \in [n_\alpha]$; recall Corollary 6.2. Our solution $\mathcal{C}$ to the instance of McC will consist of the vertices $v^{i_\alpha}_\alpha$, i.e., exactly one vertex per color α. Moreover, each new edge $x_\alpha x_\beta \in E(G[\mathcal{X}])$ must be placed by σ_G on some page, and as our construction satisfies Properties 1 and 3, this page must be one that is associated to one edge $e = v^{i_\alpha}_\alpha v^{i_\beta}_\beta$ of G_C. Property 3 now also guarantees that this page assignment enforces that x_α and x_β are placed precisely between the consecutive vertices $u^{i_\alpha}_\alpha$ and $u^{i_\alpha+1}_\alpha$ and $u^{i_\beta}_\beta$ and $u^{i_\beta+1}_\beta$ of H, respectively. This means that the vertices in $\mathcal{C}$ are pairwise adjacent, which implies that $\mathcal{C}$ is a colorful k-clique. ◀

Figure 8 shows an example of the reduction for a small graph G_C with three colors. Since in a stack layout constructed by our reduction each line perpendicular to the spine intersects a constant number of edges, see also Figure 8, we also obtain:

▶ **Corollary 6.5.** SLE *parameterized by the number* κ *of missing vertices and edges and the page width* ω *of the given layout, i.e., by* $\kappa + \omega$*, is* W[1]*-hard.*

7 Adding the Number of Pages as Parameter for SLE

In this section, we complete the landscape of Figure 2 by showing that SLE becomes fixed-parameter tractable once we add ℓ to the parameterization considered by Corollary 6.5. We will make use of the following concepts.

Consider a page p of a stack layout $\langle\prec, \sigma\rangle$ of G and recall that we can interpret it as a plane drawing of the graph G' with $V(G') = V(G)$ and $E(G') = \{e \in E(G) \mid \sigma(e) = p\}$ on a half-plane, where the edges are drawn as (circular) arcs. A *face* on the page p in $\langle\prec, \sigma\rangle$ coincides with the notion of a face in the drawing (on the half-plane p) of G'. This also includes the definition of the *outer face*. See Figure 9 for a visualization of this concept and observe that we can identify every face, except the outer face, by the unique edge $e = uv \in E(G)$ with $u \prec v$ and $\sigma(e) = p$ that bounds it from upwards.

Let $V_{\text{inc}} \subseteq V(H)$ be the vertices of H that are incident to new edges, i.e., $V_{\text{inc}} := \{u \in V(H) \mid \text{there is an edge } e = uv \in E_{\text{add}}\}$. The size of V_{inc} is upper-bounded by $2m_{\text{add}}$. We will define an equivalence class on the intervals of $\prec_H$ based on the location of the vertices from V_{inc}. Consider the two intervals $[u_1, v_1]$ and $[u_2, v_2]$ defined by the old vertices u_1, v_1, u_2 and v_2, respectively. These two intervals are in the same equivalence class if and only if

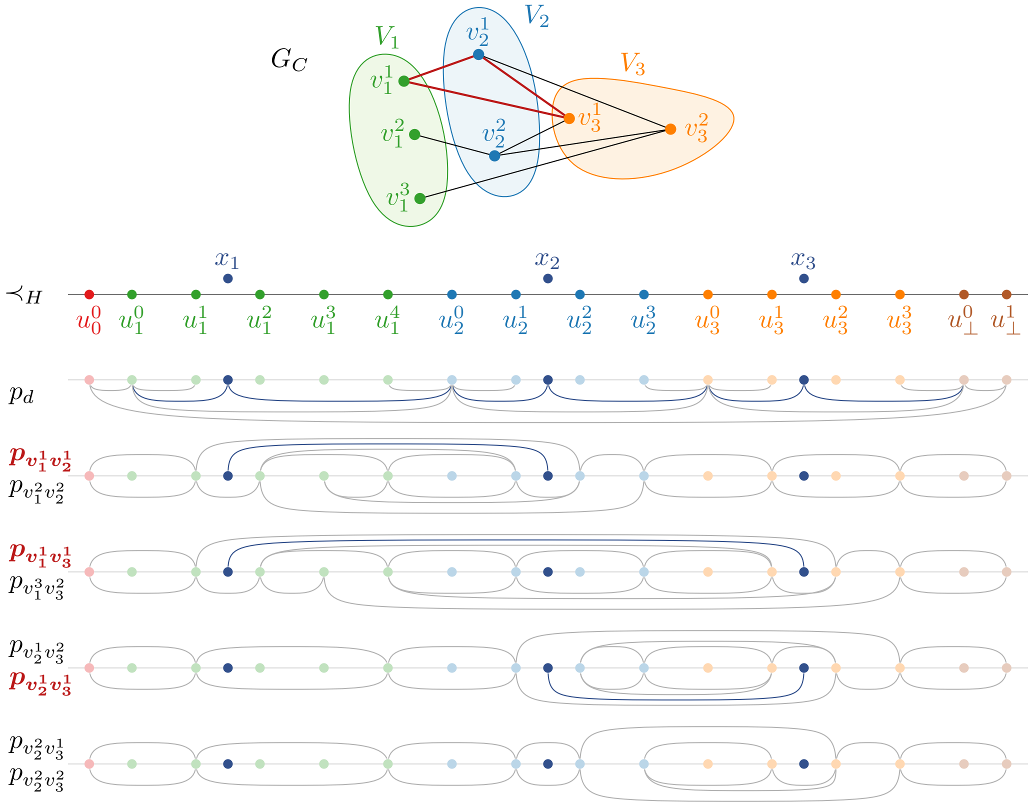

Figure 8 An instance $(G_C, 3, (V_1, V_2, V_3))$ of McC (top) and the SLE instance resulting from our construction (bottom). Colors indicate (correspondence to) the partition. The extension $\langle\prec,\sigma\rangle$ indicated in saturated colors induces the colorful 3-clique $\mathcal{C} = \{v_1^1, v_2^1, v_3^1\}$ in G_C. The edges in $G_C[\mathcal{C}]$ and their corresponding pages are highlighted in red.

$\{w \in V_{\text{inc}} \mid w \preceq u_1\} = \{w \in V_{\text{inc}} \mid w \preceq u_2\}$ and $\{w \in V_{\text{inc}} \mid v_1 \preceq w\} = \{w \in V_{\text{inc}} \mid v_2 \preceq w\}$ holds. Each equivalence class, which we call *super interval*, consists of a set of consecutive intervals delimited by (up to) two old vertices; see Figure 10. Note that the first and last super interval are defined by a single vertex $v \in V_{\text{inc}}$. The number of super intervals is bounded by $2m_{\text{add}} + 1$. Furthermore, for a given $\prec_G$, we define $\prec_{G\setminus H}$ to be its restriction to new vertices, i.e., for every two vertices $u, v \in V_{\text{add}}$ we have that $u \prec v$ implies $u \prec_{G\setminus H} v$.

The Algorithm. With the above concepts at hand, we can now describe our algorithm. It consists of a branching step, where we consider all possible page assignments for the new edges, all relative orders among the new vertices, all their possible assignments to super intervals, and all distances new edges can have from the outer face. In the following, we show that we can verify in polynomial-time whether a branch can be extended to a solution $\langle\prec,\sigma\rangle$ or not. The core of our algorithm is a dynamic program that we apply in each branch.

▶ **Lemma 7.1** (★). *Given an instance $\mathcal{I} = (\ell, G, H, \langle\prec,\sigma\rangle)$ of* SLE*,* **(i)** *a page assignment σ_G for all edges,* **(ii)** *an order $\prec_{G\setminus H}$ in which the new vertices will appear along the spine,* **(iii)** *for every new vertex $v \in V_{add}$ an assignment to a super interval, and* **(iv)** *for every new edge e an assigned distance ω_e to the outer face with respect to H and $\langle\prec,\sigma\rangle$. In $\mathcal{O}(n_{add} \cdot m_{add} \cdot |\mathcal{I}|)$ time we can compute an ℓ-page stack layout of G that extends $\langle\prec,\sigma\rangle$ and respects the given assignments* **(i)**–**(iv)** *or report that no such layout exists.*

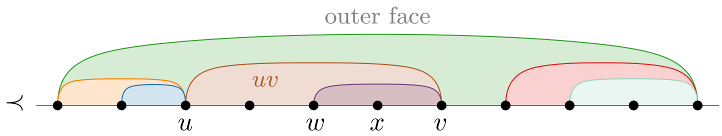

Figure 9 A stack layout $\langle \prec, \sigma \rangle$ and the faces on page p. Note that each edge has the same color as the face it identifies.

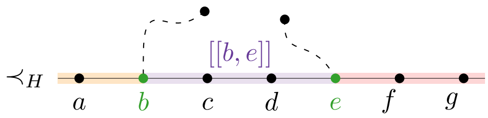

Figure 10 Visualization of super intervals. Each color represents one super interval. Vertices from V_{inc} are marked in green.

Proof sketch. We first observe that assignments (i)–(iv) fix everything except for the actual position of the new vertices within their super interval. Especially, assignment (i) allows us to check whether an edge $e \in E^H_{\text{add}}$ incident to two old vertices crosses any old edge or another new edge from E^H_{add}. Furthermore, assignments (i) and (ii) allow us to check whether two new edges $e = ua, e' = vb \in E_{\text{add}}$ with $u, a, v, b \in V_{\text{add}}$ will cross. Adding assignment (iii), we can also check this for new edges with some endpoints in $V(H)$, i.e., extend this to all $u, a, v, b \in V(G)$. If the assignments imply a crossing or contradict each other, we can directly return that no desired layout exists. These checks can be performed in $\mathcal{O}({n_{\text{add}}}^2 + m_{\text{add}} \cdot |\mathcal{I}|)$ time. It remains to check whether there exists a stack layout in which no edge of $E_{\text{add}} \setminus E^H_{\text{add}}$ intersects an old edge. This depends on the exact intervals new vertices are placed in.

To do so, we need to assign new vertices to faces such that adjacent new vertices are in the exact same face and not two different faces with the same distance to the outer face. We will find this assignment using a dynamic program that models whether there is a solution that places the first j new vertices (according to $\prec_{G \setminus H}$) within the first i intervals in $\prec_H$. When placing vertex v_{j+1} in the ith interval, we check that all preceding neighbors are visible in the faces assigned by (iv). When advancing to the interval $i + 1$, we observe that when we leave a face, all edges with the same or a higher distance to the outer face need to have both endpoints placed or none. We thus ensure that for no edge only one endpoint has been placed; see also Figure 11. These checks require $\mathcal{O}(m_{\text{add}})$ time for each of the $\mathcal{O}(n_{\text{add}} \cdot |V(H)|)$ combinations of j and i. Once we reach the interval $|V(H)| + 1$ and have successfully placed all n_{add} new vertices, we know that there exists an ℓ-page stack layout of G that extends $\langle \prec, \sigma \rangle$ and respects the assignments. Finally, by applying standard backtracing techniques, we can extract the spine positions of the new vertices to also obtain the layout. ◀

We observe that there are $\mathcal{O}(\ell^{m_{\text{add}}} \cdot n_{\text{add}}! \cdot {m_{\text{add}}}^{n_{\text{add}}} \cdot \omega^{m_{\text{add}}})$ different possibilities for assignments (i)–(iv). Applying Lemma 7.1 to each of these, we get the following theorem.

Figure 11 Illustration of advancing from the ith interval, marked in blue, to the interval $i + 1$. In **(a)** and **(b)**, we leave the green face and there exists an edge $e \in E_{\text{add}}$, marked in orange, with the same distance to the outer face as the green face. However, in **(a)**, both end points of the edge e have already been placed, whereas in **(b)** only one has, which implies a crossing.

▶ **Theorem 7.2** (★). *Let $\mathcal{I} = (\ell, G, H, \langle \prec, \sigma \rangle)$ be an instance of* SLE. *We can find an ℓ-page stack layout of G that extends $\langle \prec, \sigma \rangle$ or report that none exists in $\mathcal{O}(\ell^{m_{add}} \cdot n_{add}! \cdot {m_{add}}^{n_{add}} \cdot \omega^{m_{add}} \cdot (n_{add} \cdot m_{add} \cdot |\mathcal{I}|))$ time.*

8 Concluding Remarks

Our results provide the first investigation of the drawing extension problem for stack layouts through the lens of parameterized algorithmics. We show that the complexity-theoretic behavior of the problem is surprisingly rich and differs from that of previously studied drawing extension problems. One prominent question left for future work is whether one can still achieve fixed-parameter tractability for SLE when parameterizing by $\kappa + \ell$, thus generalizing Theorem 7.2. As our final result, we show that this is indeed possible at least in the restricted case where no two missing vertices are adjacent, as we can then greedily assign the first "possible" interval to each vertex that complies with assignment (i)–(iii).

▶ **Theorem 8.1** (★). *Let $\mathcal{I} = (\ell, G, H, \langle \prec, \sigma \rangle)$ be an instance of* SLE *where $G[V_{add}]$ is an independent set. We can find an ℓ-page stack layout of G that extends $\langle \prec, \sigma \rangle$ or report that none exists in $\mathcal{O}(\ell^{m_{add}} \cdot n_{add}! \cdot {m_{add}}^{n_{add}} \cdot (m_{add}|\mathcal{I}|^2))$ time.*

A further natural and promising direction for future work is to consider generalizing the presented techniques to other types of linear layouts, such as queue layouts. Finally, future work could also investigate the following generalized notion of extending linear layouts: Given a graph G, the spine order for some subset of its vertices and the page assignment for some subset of its edges, does there exist a linear layout of G that extends both simultaneously?

References

1 Patrizio Angelini, Giuseppe Di Battista, Fabrizio Frati, Vít Jelínek, Jan Kratochvíl, Maurizio Patrignani, and Ignaz Rutter. Testing Planarity of Partially Embedded Graphs. *ACM Transactions on Algorithms*, 11(4):1–42, 2015. doi:10.1145/2629341.

2 Patrizio Angelini, Ignaz Rutter, and T. P. Sandhya. Extending Partial Orthogonal Drawings. In David Auber and Pavel Valtr, editors, *Proc. 28th International Symposium on Graph Drawing and Network Visualization (GD'20)*, volume 12590 of *Lecture Notes in Computer Science*, pages 265–278. Springer, 2020. doi:10.1007/978-3-030-68766-3_21.

3 Patrizio Angelini, Ignaz Rutter, and T. P. Sandhya. Extending Partial Orthogonal Drawings. *Journal of Graph Algorithms and Applications*, 25(1):581–602, 2021. doi:10.7155/jgaa.00573.

4 Michael Baur and Ulrik Brandes. Crossing Reduction in Circular Layouts. In Juraj Hromkovič, Manfred Nagl, and Bernhard Westfechtel, editors, *Proc. 30th International Workshop on Graph-Theoretic Concepts in Computer Science (WG'04)*, volume 3353 of *Lecture Notes in Computer Science*, pages 332–343. Springer, 2004. doi:10.1007/978-3-540-30559-0_28.

5 Michael A. Bekos, Michael Kaufmann, Fabian Klute, Sergey Pupyrev, Chrysanthi N. Raftopoulou, and Torsten Ueckerdt. Four Pages Are Indeed Necessary for Planar Graphs. *Journal of Computational Geometry*, 11(1):332–353, 2020. doi:10.20382/JOCG.V11I1A12.

6 Frank Bernhart and Paul C. Kainen. The Book Thickness of a Graph. *Journal of Combinatorial Theory, Series B*, 27(3):320–331, 1979. doi:10.1016/0095-8956(79)90021-2.

7 Sujoy Bhore, Giordano Da Lozzo, Fabrizio Montecchiani, and Martin Nöllenburg. On the Upward Book Thickness Problem: Combinatorial and Complexity Results. In Helen C. Purchase and Ignaz Rutter, editors, *Proc. 29th International Symposium on Graph Drawing and Network Visualization (GD'21)*, volume 12868 of *Lecture Notes in Computer Science*, pages 242–256. Springer, 2021. doi:10.1007/978-3-030-92931-2_18.

8 Sujoy Bhore, Giordano Da Lozzo, Fabrizio Montecchiani, and Martin Nöllenburg. On the Upward Book Thickness Problem: Combinatorial and Complexity Results. *European Journal of Combinatorics*, 110:103662, 2023. doi:10.1016/j.ejc.2022.103662.

9 Sujoy Bhore, Robert Ganian, Liana Khazaliya, Fabrizio Montecchiani, and Martin Nöllenburg. Extending Orthogonal Planar Graph Drawings Is Fixed-Parameter Tractable. In Erin W. Chambers and Joachim Gudmundsson, editors, *Proc. 39th International Symposium on Computational Geometry (SoCG'23)*, volume 258 of *LIPIcs*, pages 18:1–18:16. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2023. doi:10.4230/LIPICS.SOCG.2023.18.

10 Sujoy Bhore, Robert Ganian, Fabrizio Montecchiani, and Martin Nöllenburg. Parameterized Algorithms for Book Embedding Problems. In Daniel Archambault and Csaba D. Tóth, editors, *Proc. 27h International Symposium on Graph Drawing and Network Visualization (GD'19)*, volume 11904 of *Lecture Notes in Computer Science*, pages 365–378. Springer, 2019. doi:10.1007/978-3-030-35802-0_28.

11 Sujoy Bhore, Robert Ganian, Fabrizio Montecchiani, and Martin Nöllenburg. Parameterized Algorithms for Book Embedding Problems. *Journal of Graph Algorithms and Applications*, 24(4):603–620, 2020. doi:10.7155/jgaa.00526.

12 Tomasz Bilski. Embedding graphs in books: a survey. *IEE Proceedings E (Computers and Digital Techniques)*, 139(2):134, 1992. doi:10.1049/ip-e.1992.0021.

13 Guido Brückner and Ignaz Rutter. Partial and Constrained Level Planarity. In Philip N. Klein, editor, *Proc. 28th ACM-SIAM Symposium on Discrete Algorithms (SODA'17)*, pages 2000–2011. SIAM, 2017. doi:10.1137/1.9781611974782.130.

14 Fan R. K. Chung, Frank Thomson Leighton, and Arnold L. Rosenberg. Embedding Graphs in Books: A Layout Problem with Applications to VLSI Design. *SIAM Journal on Algebraic Discrete Methods*, 8(1):33–58, 1987. doi:10.1137/0608002.

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

16 Giordano Da Lozzo, Giuseppe Di Battista, and Fabrizio Frati. Extending Upward Planar Graph Drawings. *Computational Geometry Theory and Applications*, 91:101668, 2020. doi:10.1016/j.comgeo.2020.101668.

17 Thomas Depian, Simon D. Fink, Robert Ganian, and Martin Nöllenburg. The Parameterized Complexity of Extending Stack Layouts, 2024. doi:10.48550/arXiv.2409.02833.

18 Reinhard Diestel. *Graph Theory, 4th Edition*, volume 173 of *Graduate Texts in Mathematics*. Springer, 2012.

19 Rodney G. Downey and Michael R. Fellows. *Fundamentals of Parameterized Complexity*. Texts in Computer Science. Springer, 2013. doi:10.1007/978-1-4471-5559-1.

20 Vida Dujmović and David R. Wood. On Linear Layouts of Graphs. *Discrete Mathematics & Theoretical Computer Science*, Vol. 6 no. 2, 2004. doi:10.46298/dmtcs.317.

21 Eduard Eiben, Robert Ganian, Thekla Hamm, Fabian Klute, and Martin Nöllenburg. Extending Partial 1-Planar Drawings. In Artur Czumaj, Anuj Dawar, and Emanuela Merelli, editors, *Proc. 47th International Colloquium on Automata, Languages and Programming (ICALP'20)*, volume 168 of *LIPIcs*, pages 43:1–43:19. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2020. doi:10.4230/LIPICS.ICALP.2020.43.

22 Eduard Eiben, Robert Ganian, Thekla Hamm, Fabian Klute, and Martin Nöllenburg. Extending Nearly Complete 1-Planar Drawings in Polynomial Time. In Javier Esparza and Daniel Král', editors, *Proc. 45th Mathematical Foundations of Computer Science (MFCS'20)*, volume 170 of *LIPIcs*, pages 31:1–31:16. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2020. doi:10.4230/LIPICS.MFCS.2020.31.

23 Robert Ganian, Thekla Hamm, Fabian Klute, Irene Parada, and Birgit Vogtenhuber. Crossing-Optimal Extension of Simple Drawings. In Nikhil Bansal, Emanuela Merelli, and James Worrell, editors, *Proc. 48th International Colloquium on Automata, Languages and Programming (ICALP'21)*, volume 198 of *LIPIcs*, pages 72:1–72:17. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2021. doi:10.4230/LIPICS.ICALP.2021.72.

24 Robert Ganian, Haiko Müller, Sebastian Ordyniak, Giacomo Paesani, and Mateusz Rychlicki. A Tight Subexponential-Time Algorithm for Two-Page Book Embedding. In Karl Bringmann, Martin Grohe, Gabriele Puppis, and Ola Svensson, editors, *Proc. 51st International Colloquium on Automata, Languages and Programming (ICALP'24)*, volume 297 of *LIPIcs*, pages 68:1–68:18. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2024. `doi:10.4230/LIPICS.ICALP.2024.68`.

25 Emden R. Gansner and Yehuda Koren. Improved Circular Layouts. In Michael Kaufmann and Dorothea Wagner, editors, *Proc. 14th International Symposium on Graph Drawing and Network Visualization (GD'06)*, volume 4372 of *Lecture Notes in Computer Science*, pages 386–398. Springer, 2006. `doi:10.1007/978-3-540-70904-6_37`.

26 Michael R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman, 1979.

27 Christian Haslinger and Peter F. Stadler. RNA Structures with Pseudo-knots: Graph-theoretical, Combinatorial, and Statistical Properties. *Bulletin of Mathematical Biology*, 61(3):437–467, 1999. `doi:10.1006/bulm.1998.0085`.

28 Vít Jelínek, Jan Kratochvíl, and Ignaz Rutter. A Kuratowski-Type Theorem for Planarity of Partially Embedded Graphs. *Computational Geometry Theory and Applications*, 46(4):466–492, 2013. `doi:10.1016/j.comgeo.2012.07.005`.

29 Yunlong Liu, Jie Chen, Jingui Huang, and Jianxin Wang. On Parameterized Algorithms for Fixed-Order Book Thickness with respect to the Pathwidth of the Vertex Ordering. *Theoretical Computer Science*, 873:16–24, 2021. `doi:10.1016/J.TCS.2021.04.021`.

30 Taylor L. Ollmann. On the Book Thicknesses of Various Graphs. In *Proc. 4th Southeastern Conference on Combinatorics, Graph Theory and Computing*, volume 8, page 459, 1973.

31 Maurizio Patrignani. On Extending a Partial Straight-Line Drawing. In Patrick Healy and Nikola S. Nikolov, editors, *Proc. 13th International Symposium on Graph Drawing and Network Visualization (GD'05)*, volume 3843 of *Lecture Notes in Computer Science*, pages 380–385. Springer, 2005. `doi:10.1007/11618058_34`.

32 Maurizio Patrignani. On Extending a Partial Straight-Line Drawing. *International Journal of Foundations of Computer Science*, 17(5):1061–1070, 2006. `doi:10.1142/S0129054106004261`.

33 Sergey Pupyrev. A Collection of Existing Results on Stack and Queue Numbers, 2023. Last accessed: 2024-05-20. URL: `https://spupyrev.github.io/linearlayouts.html`.

34 Elena Stöhr. The pagewidth of trivalent planar graphs. *Discrete Mathematics*, 89(1):43–49, 1991. `doi:10.1016/0012-365X(91)90398-L`.

35 Elena Stöhr. A Trade-off between Page Number and Page Width of Book Embeddings of Graphs. *Information and Computation*, 79(2):155–162, 1988. `doi:10.1016/0890-5401(88)90036-3`.

36 Walter Unger. On the k-Colouring of Circle-Graphs. In Robert Cori and Martin Wirsing, editors, *Proc. 5th Symposium on Theoretical Aspects of Computer Science (STACS'88)*, volume 294 of *Lecture Notes in Computer Science*, pages 61–72. Springer, 1988. `doi:10.1007/BFB0035832`.

37 Walter Unger. The Complexity of Colouring Circle Graphs (Extended Abstract). In Alain Finkel and Matthias Jantzen, editors, *Proc. 9th Symposium on Theoretical Aspects of Computer Science (STACS'92)*, volume 577 of *Lecture Notes in Computer Science*, pages 389–400. Springer, 1992. `doi:10.1007/3-540-55210-3_199`.

38 Martin Wattenberg. Arc Diagrams: Visualizing Structure in Strings. In Pak Chung Wong and Keith Andrews, editors, *Proc. 8th IEEE Information Visualization Conference (InfoVis'02)*, pages 110–116. IEEE Computer Society, 2002. `doi:10.1109/INFVIS.2002.1173155`.

39 Avi Widgerson. The Complexity of the Hamiltonian Circuit Problem for Maximal Planar Graphs. Technical Report 298, Princeton University, 1982.

40 Mihalis Yannakakis. Embedding Planar Graphs in Four Pages. *Journal of Computer and System Sciences*, 38(1):36–67, 1989. `doi:10.1016/0022-0000(89)90032-9`.