A Graph Width Perspective on Partially Ordered Hamiltonian Paths and Cycles II

Vertex and Edge Deletion Numbers

Jesse Beisegel ✉ 🄌
Institute of Mathematics, Brandenburg University of Technology, Cottbus, Germany

Katharina Klost ✉ 🄌
Institute of Computer Science, Freie Universität Berlin, Germany

Kristin Knorr ✉ 🄌
Institute of Computer Science, Freie Universität Berlin, Germany

Fabienne Ratajczak ✉ 🄌
Institute of Mathematics, Brandenburg University of Technology, Cottbus, Germany

Robert Scheffler ✉ 🄌
Institute of Mathematics, Brandenburg University of Technology, Cottbus, Germany

──── **Abstract** ────

We consider the problem of finding a Hamiltonian path or cycle with precedence constraints in the form of a partial order on the vertex set. We study the complexity for graph width parameters for which the ordinary problems HAMILTONIAN PATH and HAMILTONIAN CYCLE are in FPT. In particular, we focus on parameters that describe how many vertices and edges have to be deleted to become a member of a certain graph class. We show that the problems are W[1]-hard for such restricted cases as vertex distance to path and vertex distance to clique. We complement these results by showing that the problems can be solved in XP time for vertex distance to outerplanar and vertex distance to block. Furthermore, we present some FPT algorithms, e.g., for edge distance to block. Additionally, we prove para-NP-hardness when considered with the edge clique cover number.

1 Introduction

Hamiltonian Paths and Cycles with Precedence. For some applications of the well-known TRAVELING SALESMAN PROBLEM (TSP) it is necessary to add additional precedence constraints to the vertices which ensure that some vertices are visited before others in a tour. Examples are *Pick-up and Delivery* Problems [42, 43] or the *Dial-a-Ride* problem [45], where goods or people have to be picked up before they can be brought to their destination.

The generalization of TSP with this type of constraint is known as TRAVELING SALESMAN PROBLEM WITH PRECEDENCE CONSTRAINTS (TSP-PC) and has been studied, e.g., in [1, 9]. If we are looking for a path instead of a cycle, this is known as the SEQUENTIAL ORDERING PROBLEM (SOP) or the MINIMUM SETUP SCHEDULING PROBLEM and has been studied, e.g., in [2, 11, 22, 23].

As TSP is already NP-complete without the precedence constraints, research on these problems has mainly focused on the practical applications and solution methods are restricted to heuristic algorithms and integer programming. When moving to the structurally different Hamiltonian path (cycle) problem, only very restricted precedence constraints have been studied, e.g., where one or both endpoints of the Hamiltonian path are fixed. For these problems, polynomial-time algorithms have been presented for several graph classes including (proper) interval graphs [3, 4, 38, 39] and rectangular grid graphs [33].

In [7], Beisegel et al. introduce the PARTIALLY ORDERED HAMILTONIAN PATH PROBLEM (POHPP), where we are given a graph together with a partial order and search for a Hamiltonian path that is a linear extension of the partial order. Additionally, they introduced the edge-weighted variant MINIMUM PARTIALLY ORDERED HAMILTONIAN PATH PROBLEM (MinPOHPP). The authors show that POHPP is already NP-hard for complete bipartite graphs and complete split graphs – graph classes where HAMILTONIAN PATH is trivial. They also show that POHPP is W[1]-hard when parameterized by the *width* of the partial order, i.e., the largest number of pairwise incomparable elements. Furthermore, they show that the XP algorithm for that parameterization presented in the 1980s by Coulbourn and Pulleyblank [11] is asymptotically optimal – assuming the Exponential Time Hypothesis (ETH). They improve the algorithm to FPT time if the problem is parameterized by the partial order's *distance to linear order*. Finally, the authors present a polynomial-time algorithm for MinPOHPP on outerplanar graphs.

Graph Width Parameters and Hamiltonicity.   In [7], Beisegel et al. imply that their result for outerplanar graphs might be extended to series parallel graphs, otherwise known as the graphs of treewidth two. Furthermore, they suggest further research on the complexity of (Min)POHPP for graphs of bounded treewidth as well as other graph width parameters. Just as for specific graph classes, the prerequisite to solving (Min)POHPP for some graph parameter is the tractability of the base problems HAMILTONIAN PATH and HAMILTONIAN CYCLE. These belong to the most famous NP-complete graph problems and, hence, have been studied for a wide range of graph classes and graph width parameters.[1]

One of the first results for width parameters were polynomial-time algorithms for HAMILTONIAN CYCLE and TSP on graphs of bounded *bandwidth* [31, 40]. Since both HAMILTONIAN CYCLE and HAMILTONIAN PATH are expressible in MSO_2, Courcelle's theorem [12, 13] implies FPT algorithms for these problems when parameterized by *treewidth*. The running time bounds depending on the *treewidth* k given by Courcelle's theorem are quite bad. However, there are also FPT algorithms with single exponential dependency on k [10, 14, 15]. Probably, these results cannot be transferred to *cliquewidth*: In contrast to MSO_2, both HAMILTONIAN CYCLE and HAMILTONIAN PATH are not expressible in MSO_1 [21, Corollary 5.3.5]. In fact, HAMILTONIAN CYCLE is W[1]-hard when parameterized by *cliquewidth* [27] and – unless FPT = W[1] – it is not solvable in FPT time. Furthermore, it was shown that – unless the ETH fails – the problem cannot be solved in $f(k)n^{o(k)}$ time where k is the *cliquewidth* and n is the number of vertices [28]. This lower bound is matched by an $f(k)n^{\mathcal{O}(k)}$ algorithm given by Bergougnoux et al. [8]. XP algorithms for *cliquewidth* with worse exponents have been presented earlier by Wanke [48] (for the equivalent NLC-width) and by Espelage et al. [24].

As FPT algorithms for HAMILTONIAN PATH and HAMILTONIAN CYCLE parameterized by *cliquewidth* are highly unlikely, the research has focused on upper bounds of *cliquewidth*. FPT algorithms were presented for *neighborhood diversity* [37], *distance to cluster* [19, 34], *modular*

---

[1] In general, HAMILTONIAN PATH seems to lead a shadowy existence since many positive and negative algorithmic results are only given for its more popular sibling HAMILTONIAN CYCLE.
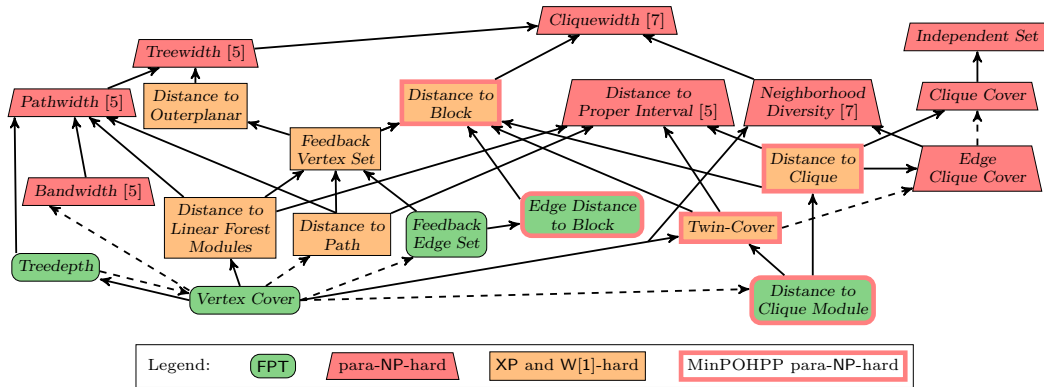
Figure 1 Diagram illustrating the complexity results for (Min)POHPP for different graph width parameters. A directed solid edge from parameter P to parameter Q means that a bounded value of P implies a bounded value for Q. A directed dashed edge implies that this relation does not hold in general but for traceable graphs, i.e., graphs having a Hamiltonian path. If a directed solid path from P to Q is missing, then parameter Q is unbounded for the graphs of bounded P. The same holds for the traceable graphs if there is also no path using dashed edges.

width [29] and *split matching width* [46, 47]. Besides this, parameterized algorithms have also been given for graph width parameters incomparable to *cliquewidth* and *treewidth*. Examples are FPT algorithms for *distance to proper interval graphs* [32] and for two parameters that describe the distance to Dirac's condition on the existence of Hamiltonian paths [35]. Most recently, FPT algorithms for the *independence number* have been presented [26].

The parameters *treewidth*, *pathwidth*, and *bandwidth* in the context of (Min)POHPP have been studied in the companion paper [5], where we showed that POHPP is NP-hard for small constants. Furthermore, it follows directly from the NP-completeness of POHPP on complete bipartite graphs that it is also NP-complete on graphs of *cliquewidth* 2, *neighborhood diversity* 2, *modular width* 2 and *split matching width* 1. As implied before, the next likely candidates to yield FPT or XP algorithms are the parameters that consider the smallest number of vertices or edges that have to be removed such that the resulting graph is in some given graph class. We refer to these parameters as *(vertex) distance to $\mathcal{G}$* and *edge distance to $\mathcal{G}$* where $\mathcal{G}$ is the respective graph class.[2] Here, we will also consider a special variant of these distance parameters that is motivated by the parameter *twin-cover number* introduced by Ganian [30]. We say a graph has *distance to $\mathcal{G}$ module(s)*[3] k if there is a set $W \subseteq V(G)$ of k vertices such that $G - W$ is in $\mathcal{G}$ and every component of $G - W$ forms a module in G. Using this terminology, the *twin-cover number* is equal to the *distance to cluster modules*.

Our Contribution. We study the complexity of (Min)POHPP and its cycle variant for graph width parameters where HAMILTONIAN PATH and HAMILTONIAN CYCLE can be solved in FPT time (see Figure 1 for an overview). We will show that the computational complexity of the path and cycle problems are essentially equivalent (see Observations 2.1 and 2.2) and therefore all proofs will focus on the path version.

---

[2] Alternative terms for these parameters are $\mathcal{G}$ *vertex deletion number* and $\mathcal{G}$ *edge deletion number*.

[3] If the class $\mathcal{G}$ contains only connected graphs, then we use *distance to $\mathcal{G}$ module* otherwise we use *distance to $\mathcal{G}$ modules*.

Section 3 is dedicated to sparse parameters, i.e., graph parameters that are unbounded for complete graphs. We show that POHPP is W[1]-hard for *distance to path* and *distance to linear forest modules*. We present simple arguments why MinPOHPP can be solved in FPT time for *feedback edge set number* and *treedepth*. Furthermore, we present an XP algorithm for *distance to outerplanar*. We also give an FPT algorithm for plane graphs parameterized by the number of vertices that lie in the interior of the outer face.

In Section 4, we consider graph width parameters that are bounded for cliques. We show that POHPP is NP-complete for graphs of *edge clique cover number* 3 and *clique cover number* 2. Furthermore, we prove W[1]-hardness for *distance to clique* and *distance to cluster modules* aka *twin cover number*. On the positive side, we present an XP algorithm for POHPP when parameterized by *distance to block* and FPT algorithms when parameterized by *edge distance to block* or *distance to clique module*.

Due to space constraints, we omit some of the proofs and sketch the others. The detailed proofs can be found in the full version [6].

2   Preliminaries

General Notation and Partial Orders.   For $n \in \mathbb{N}$, the notation $[n]$ refers to the set $\{i \in \mathbb{N} \mid 1 \leq i \leq n\}$. A *partial order* π on a set X is a reflexive, antisymmetric and transitive relation on X. The tuple (X, π) is then called a *partially ordered set*. We also denote $(x, y) \in \pi$ by $x \prec_\pi y$ if $x \neq y$. If it is clear which partial order is meant, then we sometimes omit the index. A *minimal element* of a partial order π on X is an element $x \in X$ for which there is no element $y \in X$ with $y \prec_\pi x$.

A *linear ordering* of a finite set X is a bijection $\sigma : X \to \{1, 2, \ldots, |X|\}$. We will often refer to linear orderings simply as orderings. Furthermore, we will denote an ordering by a tuple $(x_1, \ldots, x_n)$ which means that $\sigma(x_i) = i$. Given two elements x and y in X, we say that x is *to the left* (resp. *to the right*) of y if $\sigma(x) < \sigma(y)$ (resp. $\sigma(x) > \sigma(y)$) and we denote this by $x \prec_\sigma y$ (resp. $x \succ_\sigma y$). A *linear extension* of a partial order π is a linear ordering σ of X that fulfills all conditions of π, i.e., if $x \prec_\pi y$, then $x \prec_\sigma y$.

Graphs and Graph Classes.   All the graphs considered here are finite. For the standard notation of graphs we refer to the book of Diestel [18].

For a subgraph H of G, we say that two vertices $v, w \in V(H)$ are *H-neighbors* if $vw \in E(H)$. A vertex v of a connected graph G is a *cut vertex* if $G - v$ is not connected. If G does not contain a cut vertex, then G is *2-connected*. A *block* of a graph is an inclusion-maximal 2-connected induced subgraph. The *block-cut tree* $\mathcal{T}$ of G is the bipartite graph that contains a vertex for every cut vertex of G and a vertex for every block of G and the vertex of block B is adjacent to the vertex of a cut vertex v in $\mathcal{T}$ if B contains v.

A graph is a *linear forest* if its connected components are paths. A graph is a *cluster graph* if its components are cliques. A graph is a *block graph* if its blocks are cliques.

A graph is *planar* if it has a crossing-free embedding in the plane, and together with this embedding it is called a *plane graph*. For a plane graph G we call the regions of $\mathbb{R}^2 \setminus G$ the *faces* of G. Every plane graph has exactly one unbounded face which is called the *outer face*. Note that we will use the face and the closed walk of G that forms the border of that face interchangeably. A graph is called *outerplanar* if it has a crossing-free embedding such that all of the vertices belong to the outer face and such an embedding is also called *outerplanar*.

Graph Width Parameters. An *(edge) clique cover* of a graph G is a set of cliques of G that cover all vertices (all edges) of the graph. The *(edge) clique cover number* of G is the minimal size of an (edge) clique cover of G.

Let $\mathcal{G}$ be a graph class. The *distance to $\mathcal{G}$* of a graph G is the minimal number of vertices that need to be removed from G so that the resulting graph belongs to $\mathcal{G}$. Similarly, the *edge distance to $\mathcal{G}$* of a graph G is the minimal number of edges that need to be removed from G so that the resulting graph belongs to $\mathcal{G}$. A non-trivial *module* of a graph is a strict subset $M \subsetneq V(G)$ such that for all vertices $u, v \in M$ it holds that every neighbor of u outside of M is also a neighbor of v. The *distance to $\mathcal{G}$ module(s)* is the minimal number of vertices that need to be removed from G so that the resulting graph is in $\mathcal{G}$ and every component of the resulting graph is a non-trivial module in G. Several of the distance parameters have their own name. *Vertex cover number* is equivalent to *distance to edgeless*, *feedback vertex (edge) set number* is equivalent to *(edge) distance to tree* and *twin-cover number* [30] is equivalent to *distance to cluster modules*.

Hamiltonian Paths and Cycles. A *Hamiltonian path (cycle)* of a graph G is a path (cycle) that contains all the vertices of G. A graph is *traceable* if it has a Hamiltonian path and *Hamiltonian* if it has a Hamiltonian cycle. Here, we only consider *ordered* Hamiltonian paths, i.e., one of the two possible orderings of the path is fixed. Given a partial order π on a graph's vertex set, an ordered Hamiltonian path is a *π-extending Hamiltonian path* if its order is a linear extension of π. Using this definition we can generalize the classical HAMILTONIAN PATH using precedence constraints.

▶ **Problem 1.** PARTIALLY ORDERED HAMILTONIAN PATH PROBLEM (POHPP)
Instance: A graph G, a partial order π on the vertex set of G.
Question: Is there an ordered Hamiltonian path $(v_1, \ldots, v_n)$ in G such that for all $i, j \in [n]$ it holds that if $(v_i, v_j) \in \pi$, then $i \leq j$?

A similar definition can be derived for HAMILTONIAN CYCLE.

▶ **Problem 2.** PARTIALLY ORDERED HAMILTONIAN CYCLE PROBLEM (POHCP)
Instance: A graph G, a partial order π on the vertex set of G.
Question: Is there an ordered Hamiltonian path $(v_1, \ldots, v_n)$ in G such that v_1 and v_n are adjacent and for all $i, j \in [n]$ it holds that if $(v_i, v_j) \in \pi$, then $i \leq j$?

The problems HAMILTONIAN PATH and HAMILTONIAN CYCLE without precedence constraints are independent from each other, i.e., there exist graph classes where one of the problems is trivial while the other is NP-hard. It is easy to show that HAMILTONIAN PATH is NP-complete on the graphs that are not Hamiltonian. In contrast, HAMILTONIAN CYCLE is NP-complete on traceable graphs [36, Lemma 21.18]. Using the same arguments as in [5] for graph classes, we can show that for POHPP and POHCP the status of being in FPT or XP is equivalent for many graph width parameters.

To this end, let $\mathbb{G}$ be the family of all graphs. Let $\xi : \mathbb{G} \to \mathbb{N}$ be a graph width parameter. A *graph operation* Ω is a function $\mathbb{G} \to \mathbb{G}$. We say that ξ is *closed under graph operation* Ω if there is a computable function $h : \mathbb{N} \to \mathbb{N}$ such that for all graphs G, it holds that $\xi(\Omega(G)) \leq h(\xi(G))$.

▶ **Observation 2.1** (see Observation 2.4 [5]). *Let ξ be a graph width parameter that is closed under the addition of a universal vertex. Then there is a polynomial-time FPT-reduction from the POHPP parameterized by ξ to the POHCP parameterized by ξ.*
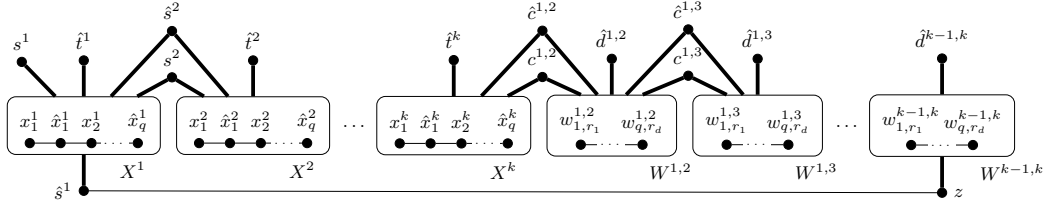
Figure 2 Construction of Theorem 3.1. If a vertex is adjacent to a box via a thick edge, then the vertex is adjacent to all vertices in that box. For *distance to path*, the ends of consecutive subpaths of the X^i and $W^{i,j}$ are adjacent, for *distance to linear forest modules* they are not adjacent.

For the converse direction, we are not able to give a (many-one) FPT-reduction. Nevertheless, we can give an FPT-Turing reduction. Furthermore, we do not need to restrict the parameter ξ in any form.

▶ **Observation 2.2** (see Observation 2.3 [5]). *Let ξ be a graph width parameter. If we can solve (Min)POHPP in time $f(\xi(G)) \cdot n^{g(\xi(G))}$ on a graph G, then we can solve (Min)POHCP in time $f(\xi(G)) \cdot n^{g(\xi(G))+2}$ on G.*

Note that Observation 2.1 does also hold for the classical problems HAMILTONIAN PATH and HAMILTONIAN CYCLE. In contrast, Observation 2.2 does not hold for the classical problems since HAMILTONIAN CYCLE is NP-complete on traceable graphs [36, Lemma 21.18].

These two observations allow us to restrict our proofs to the path case. Whenever we prove W[1]-hardness for POHPP, the respective width parameter is closed under the addition of a universal vertex. Therefore, Observation 2.1 also implies W[1]-hardness for the cycle case. If we present some FPT or XP algorithm for (Min)POHPP, then Observation 2.2 also implies an FPT or XP algorithm for (Min)POHCP.

3   Sparse Width Parameters

We start with parameters that are unbounded for cliques. As we have shown in [5], both the POHPP and the POHCP are para-NP-hard when parameterized by *treewidth*. Therefore, we focus on parameters that form upper bounds of *treewidth*.

3.1   Hardness

We start by showing that even for very restricted *distance to $\mathcal{G}$* parameters, there is no hope for FPT algorithms.

▶ **Theorem 3.1.** *POHPP and POHCP are W[1]-hard when parameterized by either distance to path or distance to linear forest modules.*

Proof. We sketch the reduction for *distance to path* from the W[1]-hard MULTICOLORED CLIQUE PROBLEM (MCP) [25, 44]. In MCP the input is a graph G whose vertex set is partitioned into k color classes $V_1, \ldots, V_k$ with $V_i := \{v_1^i, \ldots, v_q^i\}$ for all $i \in [k]$. MCP asks whether there is a clique in G containing exactly one vertex per color class. Given an instance of MCP, we construct a graph G' using the following gadgets (see Figure 2):

Selection Gadget For every color class $i \in [k]$, we have a selection gadget consisting of a vertex $\hat{t}^i$ and a path X^i that contains for every $v_p^i \in V_i$ a vertex x_p^i. These vertices are ordered by their index p and after every vertex x_p^i (and, thus, before x_{p+1}^i) there is a vertex $\hat{x}_p^i$ in that path. The vertex $\hat{t}^i$ is adjacent to all vertices in the path X^i.

Verification Gadget For each pair $i, j \in [k]$ with $i < j$, we have a verification gadget consisting of a vertex $\hat{d}^{i,j}$ and a path $W^{i,j}$ containing for every edge $v_p^i v_r^j \in E(G)$ a vertex $w_{p,r}^{i,j}$ in arbitrary order. The vertex $\hat{d}^{i,j}$ is adjacent to all vertices in the path $W^{i,j}$.

We order the subpaths described in the selection and verification gadgets as follows:

$$X^1, X^2, \ldots, X^k, W^{1,2}, W^{1,3}, \ldots, W^{1,k}, W^{2,3}, W^{2,4}, \ldots, W^{k-1,k}.$$

The last vertex of one of the paths is adjacent to the first vertex of the succeeding path and, thus, they form one large path Ψ in G'. Furthermore, G' contains vertices s^1 and $\hat{s}^1$ that are adjacent to all vertices in X^1. For every $i \in [k]$ with $i > 1$, G' contains vertices s^i and $\hat{s}^i$ that are adjacent to all vertices in X^i and X^{i-1}. For all $i, j \in [k]$ with $i < j$, the graph contains vertices $c^{i,j}$ and $\hat{c}^{i,j}$ that are adjacent to all vertices in $W^{i,j}$ and to all vertices in the path preceding $W^{i,j}$ in the ordering described above. Finally, G contains a vertex z that is adjacent to all vertices in $W^{k-1,k}$ and to $\hat{s}^1$. As there are $\mathcal{O}(k^2)$ vertices in G' that are not part of the path Ψ, the *distance to path* of G' is $\mathcal{O}(k^2)$.

We define Y as the set containing all the vertices defined above that have a hat in their name, i.e., $Y := \{\hat{s}^i \mid i \in [k]\} \cup \{\hat{t}^i \mid i \in [k-1]\} \cup \{\hat{c}^{i,j}, \hat{d}^{i,j} \mid i, j \in [k], i < j\}$. The partial order π is the reflexive and transitive closure of the following constraints:

(P1) $s^1 \prec v$ for all $v \in V(G') \setminus \{s^1\}$,

(P2) $z \prec y$ for all $y \in Y$,

(P3) $x_p^i \prec w_{p,r}^{i,j}$ for all $i, j \in [k]$ with $i < j$ and all $p, r \in [q]$ with $v_p^i v_r^j \in E(G)$

(P4) $x_r^j \prec w_{p,r}^{i,j}$ for all $i, j \in [k]$ with $i < j$ and all $p, r \in [q]$ with $v_p^i v_r^j \in E(G)$

Every π-extending Hamiltonian path has to visit exactly one vertex in each X^i and $W^{i,j}$ on its way from s^1 to z as all the vertices with hats cannot be visited before z. Due to the constraints (P3) and (P4), a vertex of $W^{i,j}$ corresponding to an edge $v_p^i v_r^j$ can only be visited if both x_p^i and x_r^j have already been visited. Therefore, we only can reach z if the visited vertices of the selection gadgets form a a multicolored clique of G.

Conversely, a multicolored clique implies that we can reach z from s^1 using such a path. We can visit the remaining vertices by traversing the path Ψ and using $\hat{c}^{i,j}, \hat{d}^{i,j}, \hat{s}^i$ and $\hat{t}^i$ outside of Ψ to jump over vertices that have already been visited. For details see the full version [6]. For *distance to linear forest modules* we can adapt the construction by deleting the edges between consecutive gadgets since these edges have not been used in the proof. The resulting graph has *distance to linear forest modules* $\mathcal{O}(k^2)$.   ◀

3.2   Algorithms

We start this section with some easy observations about width values of traceable graphs. A traceable graph with *vertex cover number* k has at most $2k + 1$ vertices. Therefore, MinPOHPP and MinPOHCP parameterized by *vertex cover number* allow kernels with a linear number of vertices and, thus, FPT algorithms. It follows from Lemma 6.2 and Equation 6.2 by Nešetřil and de Mendez [41] that a traceable graph of *treedepth* k has less than 2^k vertices. As both MinPOHPP and MinPOHCP can be solved in exponential time [7], there are simple double-exponential FPT algorithms for *treedepth*. For the *feedback edge set number* we can give a single-exponential FPT algorithm. As was shown in [16], an n-vertex graph of *feedback edge set number* k has at most $2^k \cdot \binom{n}{2}$ different paths.[4] Enumerating these paths leads to the following result.

---

[4] Note that the result is only given in the arXiv version of the paper.

▶ **Theorem 3.2.** *MinPOHPP and MinPOHCP can be solved in $2^k \cdot n^{\mathcal{O}(1)}$ time on an n-vertex graph of feedback edge set number k.*

Unless $\mathsf{W}[1] = \mathsf{FPT}$, this result cannot be extended to *feedback vertex set number* since POHPP and POHCP are $\mathsf{W}[1]$-hard for this parameter, due to Theorem 3.1. However, we can give an XP algorithm for MinPOHPP when parameterized by *distance to outerplanar*, a more general parameter than *feedback vertex set number*. Before we present this algorithm, we first focus on a more restricted case that allows an FPT algorithm.

It has been shown by Beisegel et al. [7] that MinPOHPP can be solved in $\mathcal{O}(n^2)$ time on outerplanar graphs. The key ingredient of that algorithm is the observation that the vertices of the outer face contained in a prefix of a Hamiltonian path must be an *interval* of the outer face, i.e., a consecutive part of the closed walk that forms the face (see Lemma 5.2 in [7]). This observation does not only hold for outerplanar graphs but for all plane graphs and all faces. It is important to note that this does not mean that the prefix contains this interval as a subpath. In fact, there can be vertices in between that are not part of the boundary of the face. This general result allows us to extend the polynomial-time algorithm for outerplanar graphs to an FPT algorithm for plane graphs parameterized by the number of vertices that are not on the outer face. Note that our algorithm uses ideas similar to those of [17].

▶ **Theorem 3.3.** *Given a plane graph G with k vertices not on the outer face, we can solve MinPOHPP on G in $\mathcal{O}(2^k k n^3)$ time and MinPOHCP on G in $\mathcal{O}(2^k k n^5)$ time.*

Proof. We only sketch the idea of the algorithm as it is a quite straightforward adaption of Algorithm 2 in [7]. For details, we refer to that publication. First note that an induced subgraph does not have more inner vertices for a fixed embedding. Hence, it suffices to deal with 2-connected graphs (Theorem 5.1 in [7]). Let $C = (v_0, \ldots, v_\ell)$ be the outer face of the planar embedding and let W be the set of vertices that are not on the outer face. Since G is 2-connected, the face C forms a cycle. Furthermore, let c be the cost function on the edges of G.

The idea of Algorithm 2 in [7] is to use a dynamic programming approach. To this end, the authors consider tuples (a, b, ω), which represent prefixes of Hamiltonian paths. Here, a and b are the indices of the endpoints of the interval on C associated with that prefix (see Lemma 5.2 in [7]) and ω is either 1 or 2 depending on whether the prefix ends in v_a or v_b. For the problem considered here, we simply need to adjust these to tuples of the form (a, b, X, t), where a and b again represent the indices of the endpoints of the interval on C associated with the prefix, $X \subseteq W$ is the set of inner vertices in the prefix and t forms the endpoint of the prefix. There are at most n^2 intervals and at most 2^k subsets of W. The endpoint t either has to be a vertex of X or one of the vertices v_a and v_b. Therefore, there are at most $(k + 2)$ endpoints and the total number of tuples is bounded by $2^k(k + 2)n^2$. Similar as in Algorithm 2 of [7], we compute for every tuple (a, b, X, t) the minimal cost $M(a, b, X, t)$ of an ordered path $\mathcal{P}$ of G fulfilling the following properties (or ∞ if no such path exists):

 (i) $\mathcal{P}$ consists of the vertices in the interval between v_a and v_b of C and in the set X,

 (ii) $\mathcal{P}$ is a prefix of a linear extension of π,

 (iii) $\mathcal{P}$ ends in t.

This is done inductively by the size of $\mathcal{P}$. Let a', b', and X' be the updated values if t is removed from the potential prefix. We first have to check whether t is minimal in π if all the vertices of $[a', b']$ and X' have been visited. This costs $\mathcal{O}(n)$ time. If this is not the case, we set the M-value to ∞. Otherwise, we check the M-values for all possible vertices t' that are before t in the prefix. Note that t' can only be an element of X' or one of the two
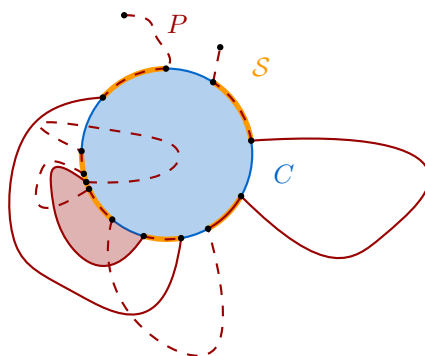
Figure 3 A face C (blue) and a prefix P of a Hamiltonian path (red). The solid parts of P are crests. The dashed parts might contain vertices of W and are thus not necessarily planar.

vertices $v_{a'}$ and $v_{b'}$, due to Lemma 5.2 in [7]. For each choice of t' we compute the value $M(a', b', X', t') + c(tt')$ and set the M-value of (a, b, X, t) to the minimum of these values. Note that this can be done in $\mathcal{O}(k) \subseteq \mathcal{O}(n)$ time using an adjacency matrix.

Summing up, for each of the $\mathcal{O}(2^k k n^2)$ tuples we need $\mathcal{O}(n)$ time which leads to the overall running time of $\mathcal{O}(2^k k n^3)$. The proof of the correctness of the algorithm follows along the lines of the proof of Theorem 5.3 in [7]. ◀

The idea of that algorithm can also be used to give an XP algorithm for *distance to outerplanar*. The crucial difference in the complexity is based on the fact that there is not only one interval of the outer face used by a prefix of a Hamiltonian path but at most $k + 1$. To prove this, we need the following definitions.

Let C be a closed walk of a plane graph G. A subgraph H lies *inside* of C if all vertices and edges of H are elements of C or are placed in a bounded region of $\mathbb{R}^2 \setminus C$. The subgraph H lies *strictly inside* of C if it lies inside of C and does not contain a vertex of C. A subgraph H lies *outside* of C if all vertices and edges of H are elements of C or are placed in the unbounded region of $\mathbb{R}^2 \setminus C$.

▶ **Lemma 3.4.** *Let G be a graph with a set $W \subseteq V(G)$ of size k such that $G - W$ is planar. Let C be a face of a planar embedding of $G - W$ and let $V(C)$ be the vertices on the boundary of C. Let P be a prefix of a Hamiltonian path of G. Then there exists a partition of $V(C) \cap V(P)$ into at most $k + 1$ intervals of C.*

Proof. We use induction over the number of vertices of G. For one vertex the statement is trivial. Assume that the statement holds for all graphs with less than n vertices. Let G be a graph with n vertices and a set $W \subseteq V(G)$ of size k such that $G - W$ is planar. Let C be a face of a planar embedding of $G - W$. Note that C could be a closed walk with repeating vertices, for example, the outer face of a graph that is not 2-connected. To simplify the notation, we assume that C does not repeat vertices. By tweaking the definitions, the other case can be proven analogously.

Let P be a prefix of a Hamiltonian path $\mathcal{P}$ of G that does not contain all vertices of $\mathcal{P}$. Let $\mathcal{S}$ be the smallest partition of $V(C) \cap V(P)$ into intervals of C, i.e., the intervals are inclusion-maximal. See Figure 3 for an illustration. Let $|\mathcal{S}| = \ell$. If we remove the elements of C from P, we get pairwise disjoint subpaths of P and the endpoints of these subpaths have neighbors on P that lie on C. We ignore the subpath that contains all vertices before the first vertex of C on P if it exists and the subpath containing all vertices after the last vertex of C on P if it exists. We also ignore all subpaths where the neighbor of the first and the

last vertex lie on the same interval of $\mathcal{S}$. Thus, we only consider subpaths that connect two intervals of $\mathcal{S}$. As all intervals of $\mathcal{S}$ have to be connected via such paths, there are at least $\ell - 1$ of these subpaths. If each of these subpaths contains a vertex of W, then $\ell \le k + 1$ and we are done. Otherwise, consider a subpath P_{sub} that does not contain vertices of W. We define the *crest* induced by P_{sub} as the subpath of P containing P_{sub} and the P-neighbors of its endpoints. Let Q be such a crest. Let v_i be the first vertex and v_j be the last vertex of Q with regard to the ordering of P. By definition, both v_i and v_j lie on C. We define $C[v_i, v_j]$ as the part of C between v_i and v_j in the cyclic ordering and $C[v_j, v_i]$ as the part of C between v_j and v_i in the cyclic ordering. Further note that v_i and v_j lie in different intervals of $\mathcal{S}$. Therefore, there are vertices not in P, both in $C[v_i, v_j]$ and in $C[v_j, v_i]$. Let t be the end vertex of P or, in case this end vertex is v_j, let t be the successor of v_j in the Hamiltonian path $\mathcal{P}$. Note that t exists as P does not contain all vertices of $\mathcal{P}$. W.l.o.g. we may assume that t lies outside of $\widetilde{Q} = Q \cup C[v_i, v_j]$. If this is not the case, then we define one face inside of $\widetilde{Q}$ as outer face without changing the faces of the embedding. Then, we can consider $Q \cup C[v_j, v_i]$ as $\widetilde{Q}$.

Let R be a *minimal* crest inside of $\widetilde{Q}$, i.e., there is no crest that lies inside of $\widetilde{R} = R \cup C[x, y]$, where x and y are the endpoints of R and $C[x, y]$ is the part of C between x and y that is inside of $\widetilde{Q}$. Note that $C[x, y]$ contains vertices that are not part of P as the crest R connects two intervals, say S_x and S_y where $x \in S_x$ and $y \in S_y$.

We will now show that we can always find either a vertex to delete or an edge to contract in order to apply the induction hypothesis.

Assume that there is an edge $ab \in E(P)$ such that w.l.o.g. b is strictly inside of $\widetilde{R}$. We contract the edge ab in G and P. The resulting graph G' and resulting path P' fulfill the induction hypothesis, i.e., $W \subseteq V(G')$, P' is a prefix of a Hamiltonian path of G' and C is a face of an embedding of G'. Therefore, we can partition $V(C) \cap V(P')$ into at most $k + 1$ intervals of C. These intervals map bijectively to intervals of $V(C) \cap V(P)$. Hence, we can assume that P does not contain any vertices strictly inside of $\widetilde{R}$.

Suppose there is an interval $S \in \mathcal{S}$ that is contained completely in $C[x, y] - \{x, y\}$. As $\widetilde{R}$ separates S from the vertices outside of $\widetilde{R}$ and no vertices in P are strictly inside of $\widetilde{R}$, all P-neighbors of vertices of S are either in S or in W. As t lies outside of $\widetilde{R}$, the interval S has at least one P-neighbor in W. If S has exactly one P-neighbor $w \in W$ – i.e. P starts with S – then we delete w and the whole interval S and connect the neighbors of S on C with an edge. This gives a graph G' with one less vertex in W and one less interval. The suffix of P that starts at the successor of w is a prefix of a Hamiltonian path in G'. The statement follows by induction.

Suppose S has more than one P-neighbor in W. We connect these neighbors to form a clique. Furthermore, we sort them by their appearance in P and contract the last and the penultimate vertex of these neighbors. As t lies outside of $\widetilde{R}$, the last neighbor of S cannot be used to enter $\tilde{R}$. Therefore, all vertices visited in P between these two neighbors lie in S. We now delete S and connect its neighbors on C with an edge. We can transform P into P' by replacing a sequence of S-vertices between two P-neighbors of S by one of the clique edges. The statement follows by induction as there is one vertex less in W and one interval less, see Figure 4.

If there is no interval that is contained completely in $C[x, y] - \{x, y\}$, then all vertices on $C[x, y]$ are either in S_x, S_y or not in P. In this case, we can apply similar arguments to contract the vertices that are not in P and, thus, join S_x and S_y to reduce the number of intervals. The details can be found in the full version [6]. ◀
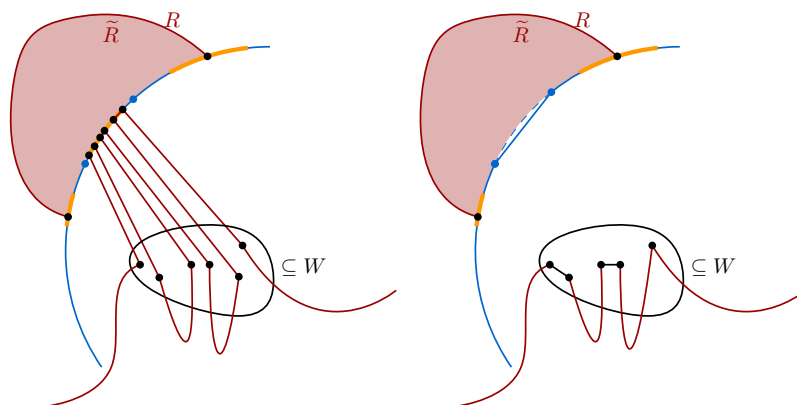
■ **Figure 4** Illustration of Lemma 3.4.

Using this lemma, we can adapt the algorithm given in Theorem 3.3 such that it solves the MinPOHPP in XP time for *distance to outerplanar*. This adaption is rather straightforward. In the dynamic program, we replace the vertices representing the single interval on C by a set of at most $k + 1$ intervals. It is not difficult to observe that the number of those sets is bounded by $n^{\mathcal{O}(k)}$. This leads to the following running time bound.

▶ **Theorem 3.5.** *Given an n-vertex graph G with distance to outerplanar k, we can solve MinPOHPP and MinPOHCP on G in $n^{\mathcal{O}(k)}$ time.*

4   Non-Sparse Width Parameters

All parameters considered so far are sparse, i.e. unbounded for cliques. As POHPP and POHCP are trivial on cliques, it makes sense to consider also non-sparse parameters. Both MinPOHPP and MinPOHCP are NP-hard on cliques as they form generalizations of (Path) TSP. Thus, we only consider the unweighted problems in this section.

4.1   Hardness

As mentioned in the introduction, HAMILTONIAN PATH and HAMILTONIAN CYCLE can be solved in FPT time when parameterized by the *independence number*, that is the size of the largest independent set [26]. For traceable graphs with at least two vertices, this parameter is a lower bound on the *edge clique cover number*.[5] Thus, both problems can also be solved in FPT time for this parameter. Here, we show that these results cannot be extended to POHPP or POHCP unless P = NP.

▶ **Theorem 4.1.** *POHPP and POHCP are NP-complete on graphs of edge clique cover number 3.*

Proof. We reduce from a special case of the ALTERNATING LINEAR EXTENSION PROBLEM that was introduced and shown to be NP-complete in [7]. In this case of the problem, we are given two non-empty disjoint sets A and B with $|A| = |B|$ and a partial order $\pi \subseteq A \times B$ and we ask for a linear extension of π that alternates between elements of A and B starting in some vertex of A.

---

[5] This does not only hold for traceable graphs but for all graphs that do not contain isolated vertices. An edge clique cover of such a graph is also a clique cover of the graph and an independent set can contain at most one vertex of every clique in that clique cover.
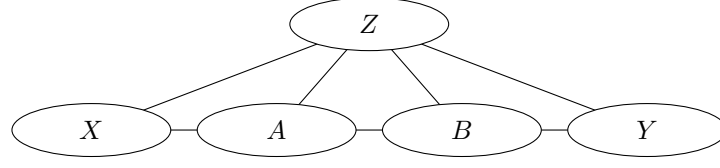
Figure 5 Construction of Theorem 4.1. The ellipses form cliques. The connection of an ellipse with another implies that all edges between the vertices of the two cliques are present.

Let (A, B, π) be an instance of this problem with $A = \{a_1, \ldots, a_n\}$ and $B = \{b_1, \ldots, b_n\}$. We build a graph G as follows (see Figure 5 for an illustration). The vertex set of G is the disjoint union of A, B, $X = \{x_1, \ldots, x_n\}$, $Y = \{y_1, \ldots, y_n\}$, and $Z = \{z_1, \ldots, z_n\}$. The graph G contains all edges except for those between X and B, between X and Y as well as those between Y and A. In other words, the edges of G can be covered by the three cliques $C_1 = A \cup X \cup Z$, $C_2 = A \cup B \cup Z$, and $C_3 = B \cup Y \cup Z$.

The partial order π' contains all tuples of π. Additionally, it fixes x_1 as start vertex and contains the following chain of constraints: $x_1 \prec y_1 \prec z_1 \prec x_2 \prec y_2 \prec z_2 \cdots \prec x_n \prec y_n \prec z_n$.

Assume the instance (A, B, π) has an alternating linear extension σ. W.l.o.g. we may assume that $\sigma = (a_1, b_1, a_2, b_2, \ldots, a_n, b_n)$. Inserting x_i before a_i and (y_i, z_i) after b_i leads to a π'-extending Hamiltonian path and cycle of G.

Conversely, assume we have a π'-extending Hamiltonian path $\mathcal{P}$ in G. We observe that $\mathcal{P}$ uses one vertex of A and one vertex of B between x_i and y_i as all vertices z_j with $j < i$ are to the left of x_i and all vertices z_j with $j \geq i$ are to the right of y_i. Since X and Y have the same size as A and B, we have to use exactly one vertex of A and one of B between every x_i and y_i. Thus, $\mathcal{P}$ contains an alternating linear extension of π. ◄

We have to leave open the case of *edge clique cover number* 2. However, we can observe that the graph G constructed in the proof of Theorem 4.1 has *clique cover number* 2 since the cliques C_1 and C_3 cover all vertices. This implies the following result.

▶ **Corollary 4.2.** *POHPP and POHCP are* NP-*complete on graphs of clique cover number 2.*

Note that for graphs of *clique cover number* 1 (i.e., cliques), POHPP and POHCP are trivial as every linear extension of π induces a Hamiltonian cycle.

It follows from Theorem 3.1 that POHPP and POHCP are W[1]-hard when parameterized by *distance to block*. Here, we extend this result to *distance to clique*.

▶ **Theorem 4.3.** *POHPP and POHCP are* W[1]-*hard when parameterized by either distance to clique or distance to cluster modules also known as twin-cover number.*

Proof. As for Theorem 3.1, we use a reduction from MULTICOLORED CLIQUE to POHPP and then use Observation 2.1 to derive W[1]-hardness also for POHCP. Let G be an instance of MCP with color classes $V_1, \ldots, V_k$ where $V_i = \{v_1^i, \ldots, v_q^i\}$. The construction works similar as for Theorem 3.1. The main difference is how we encode the selection of a vertex from a color class. While in the proof of Theorem 3.1 the representative of the selected vertex was visited, here we visit all representatives except from the one of the selected vertex.

We construct a graph G' using the following gadgets:

Selection Gadget For every color class $i \in [k]$, we have a clique X^i that contains for every $v_p^i \in V_i$ a vertex x_p^i.

Verification Gadget For each pair $i, j \in [k]$ with $i < j$, we have a clique $W^{i,j}$ containing for every edge $v_p^i v_r^j \in E(G)$ a vertex $w_{p,r}^{i,j}$.
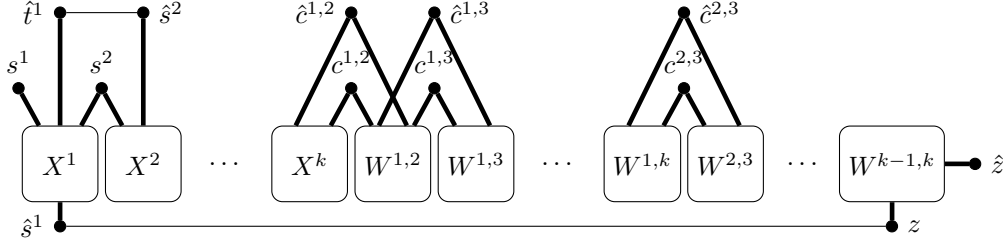
Figure 6 Construction of Theorem 4.3. If a vertex is adjacent to a box via a thick edge, then the vertex is adjacent to all vertices in that box. For *distance to clique*, all the boxes are pairwise adjacent, for *distance to linear forest modules* they are not adjacent.

Next we describe how these gadgets are connected to each other (see Figure 6). The union of the selection gadgets and the verification gadgets forms one large clique. We order the subcliques described in the selection and verification gadgets as follows:

$$X^1, X^2, \ldots, X^k, W^{1,2}, W^{1,3}, \ldots, W^{1,k}, W^{2,3}, W^{2,4}, \ldots, W^{k-1,k}.$$

We have the following additional vertices in G'. First, we have s^1, $\hat{s}^1$, and $\hat{t}^1$ that are adjacent to all vertices in X^1. For every i with $2 \le i \le k-1$, we have vertices s^i, $\hat{s}^i$ and $\hat{t}^i$. For $i = k$, we have only s^i and $\hat{s}^i$. Vertex s^i is adjacent to all vertices in X^i and X^{i-1} and the vertices $\hat{s}^i$ and $\hat{t}^i$ are adjacent to all vertices in X^i. Furthermore, $\hat{s}^i$ is adjacent to $\hat{t}^{i-1}$. For all $i, j \in [k]$ with $i < j$, there are vertices $c^{i,j}$ and $\hat{c}^{i,j}$ that are adjacent to all vertices in $W^{i,j}$ and to all vertices in the clique to left of $W^{i,j}$ in the ordering described above. Finally, we have vertices z and $\hat{z}$ that are adjacent to all vertices in $W^{k-1,k}$. Furthermore, z is adjacent to $\hat{s}^1$. Observe that the graph G' without the selection and verification gadgets contains $3k + 2\binom{k}{2} + 2$ vertices. Therefore, the *distance to clique* of G' is $\mathcal{O}(k^2)$.

The partial order π is the reflexive and transitive closure of the following constraints.

(P1) $s^1 \prec v$ for all $v \in V(G') \setminus \{s^1\}$,

(P2) $v \prec \hat{z}$ for all $v \in V(G') \setminus \{\hat{z}\}$,

(P3) $s^1 \prec s^2 \prec \cdots \prec s^k$,

(P4) $z \prec \hat{s}^1 \prec \hat{t}^1 \prec \hat{s}^2 \prec \hat{t}^2 \prec \cdots \prec \hat{s}^k$,

(P5) $c^{i,j} \prec c^{i',j'}$ for all $i, j, i', j' \in [k]$ with $i < i'$ or $i = i'$ and $j < j'$,

(P6) $c^{k-1,k} \prec z$,

(P7) $c^{i,j} \prec w^{i,j}_{p,r}$ for all $i, j \in [k]$ with $i < j$ and all $w^{i,j}_{p,r} \in W^{i,j}$,

(P8) $x^i_p \prec w^{i,j}_{r,s}$ for all $i, j \in [k]$ with $i < j$ and all $w^{i,j}_{r,s} \in W^{i,j}$ with $p \ne r$,

(P9) $x^j_p \prec w^{i,j}_{r,s}$ for all $i, j \in [k]$ with $i < j$ and all $w^{i,j}_{r,s} \in W^{i,j}$ with $p \ne s$.

First assume that there is a multicolored clique $\{v^1_{p_1}, \ldots, v^k_{p_k}\}$ in G. Then we claim that there is π-extending Hamiltonian path in G'. We start our path in s^1 and then we visit all the vertices in X^1 except for the vertex $x^1_{p_1}$. Then we visit s^2. We repeat this for all $i \in [k]$. Next we visit $c^{1,2}$. Now we visit $w^{1,2}_{p_1,p_2}$. Note that this is possible since all the vertices that have to be to the left of $w^{1,2}_{p_1,p_2}$ by (P8) and (P9) are already visited. Next we visit $c^{1,3}$ and afterwards $w^{1,3}_{p_1,p_3}$ which is possible for the same reason as mentioned above. We repeat this procedure until we reach $w^{1,k}_{p_1,p_k}$. Next, we visit $c^{2,3}$. The same procedure works until we reach $w^{k-1,k}_{p_{k-1},p_k}$. Next, we visit z. Now, starting with $i = 1$, we visit for all $i \in [k]$ the vertices $\hat{s}^i$, followed by $x^i_{p_i}$ and $\hat{t}^i$ until we reach $x^k_{p_k}$. Then, we visit $\hat{c}^{1,2}$ followed by all remaining vertices $w^{1,2}_{p,r} \in W^{1,2}$ which is possible as all the left vertices in the constraints (P8) and (P9)

have already been visited. We repeat this for all i, j with $i < j$ in the lexicographic order and eventually we have visited all vertices in $W^{k-1,k}$. Finally, we visit $\hat{z}$. It is easy to check that this Hamiltonian paths fulfills all constraints of π.

Now assume that there is a π-extending Hamiltonian path $\mathcal{P}$ in G'. Let $i \in [k]$ be arbitrary. By (P4), $\hat{s}^i$ is to the right of z. By (P2), $\hat{s}^i$ is not the last vertex of $\mathcal{P}$ and, thus, it has two neighbors in $\mathcal{P}$. One of these two neighbors has to be in X^i. Therefore, for each $i \in [k]$, there is at least one vertex in X^i that is to the right of z in $\mathcal{P}$.

Using the constraints (P8) and (P9), we can observe that there can be at most one vertex of $W^{i,j}$ to the left of z for every choice of i and j with $i < j$. Otherwise, all vertices of X^i and X^j would be to the left of z contradicting our observation above. This also implies that there is exactly one vertex of $W^{i,j}$ to the left of z since $c^{i,j}$ has two neighbors in $\mathcal{P}$ and one of these neighbors has to be an element of $W^{i,j}$. The constraints (P8) and (P9) imply now that there can be at most one vertex of every X^i to the right of z. Combining this with the existence of such a vertex shown above, we see that in every X^i there is exactly one vertex $x_{p_i}^i$ to the right of z. Since there is a vertex of every $W^{i,j}$ to the left of z, it is not difficult to see that the set $\{v_{p_1}^1, \ldots, v_{p_k}^k\}$ forms a multicolored clique in G.

The arguments above show that the graph (G', π) is a valid reduction. Since the *distance to clique* of G' is $\mathcal{O}(k^2)$ it also a proper FPT reduction from MCP to POHPP parameterized by the *distance to clique*.

Similar as in Theorem 3.1, we can adapt the construction to show that POHPP is also W[1]-hard for *distance to cluster modules*. To this end, we delete all the edges in G' between different gadgets. This works since these edges have not been used in the proof and since the resulting graph has *distance to cluster modules* $\mathcal{O}(k^2)$. ◄

4.2   Algorithms

We start with an FPT algorithm for *edge distance to block* when given a respective edge set.

▶ **Theorem 4.4.** *POHPP and POHCP can be solved in $(k+1)! \cdot 2^k \cdot n^{\mathcal{O}(1)}$ time on an n-vertex graph when given a set $F \subseteq E(G)$ with $|F| \leq k$ and $G - F$ is a block graph.*

Proof. We consider all possible options for the start vertex of the Hamiltonian path and how the path traverses through the edges of F, i.e., we fix the set of edges $\widehat{F} \subseteq F$ that are used and also fix in which direction and in which order they are used. There are at most $k \cdot k! \cdot 2^k \cdot n$ many of these options. Let $\widehat{W}$ be the set of vertices incident to edges of $\widehat{F}$. Note that we only continue if every vertex in $\widehat{W}$ has at most two incident edges in $\widehat{F}$. Furthermore, the ordering on $\widehat{F}$ implies a vertex ordering σ on $\widehat{W}$.

First, we check whether σ is valid for π. If so, in the rest of the procedure for that option, we try to fill in all the other vertices of G. Whenever there are two consecutive vertices x and y in σ that are not adjacent via an edge of $\widehat{F}$, we have to use edges of $G - F$. Since the path between the blocks of these vertices in the block-cut tree of $G - F$ is unique, we know which cut vertices have to be used on that path and we reserve these cut vertices for that path. The algorithm now traverses the ordering σ and, if necessary, uses paths in $G - F$. These paths visit all vertices of a block that are possible due to π except those that are reserved for later paths. The details can be found in the full version [6]. ◄

Note that Dumas et al. [20] present a quadratic kernel for deciding whether the *edge distance to block* of a graph G is at most k. This implies an FPT algorithm with running time $k^{\mathcal{O}(k)} \cdot n^{\mathcal{O}(1)}$ that decides whether the *edge distance to block* is at most k. Calling this algorithm $\mathcal{O}(m)$ times, we can also construct the respective set F. This implies the following.

▶ **Corollary 4.5.** *POHPP and POHCP can be solved in $((k+1)! \cdot 2^k + k^{\mathcal{O}(k)}) \cdot n^{\mathcal{O}(1)}$ time on an n-vertex graph with edge distance to block k.*

Next we give an XP algorithm for the *(vertex) distance to block k*. We can compute a set W of size $\leq k$ such that $G - W$ is a block graph in $n^{\mathcal{O}(k)}$ time. For each vertex of W, the algorithm fixes its predecessor and successor in a potential Hamiltonian path. This choice can be transferred to an instance of POHPP with *edge distance to block $\leq k$* with a fixed edge choice $\widehat{F}$. Then, it is sufficient to apply the subroutine of Theorem 4.4.

▶ **Theorem 4.6.** *If the distance to block of a graph G with n vertices is k, then POHPP and POHCP can be solved in $n^{\mathcal{O}(k)}$ time.*

As we have seen in Theorem 4.3, there is no FPT algorithm for *distance to cluster modules* unless W[1] = FPT. However, we can give such an algorithm for *distance to clique module*.

▶ **Theorem 4.7.** *Given an n-vertex graph G with distance to clique module k, POHPP and POHCP can be solved in time $k! \cdot n^{\mathcal{O}(1)}$.*

Proof. Let $W \subseteq V(G)$ be a set of size k such that $G - W$ is a clique module of G. Note that such a set can be computed in polynomial time since the largest clique module is equivalent to the largest set of vertices with the same closed neighborhood. We now consider every ordering $\sigma = (w_1, \ldots, w_k)$ of the vertices in W that fulfills the constraints of π. To decide whether σ is an eligible choice, we have to check whether we can insert the vertices of $G - W$ into σ in such a way that the result is a π-extending Hamiltonian path. Some of the vertices of W may be *secluded*, i.e., they are not adjacent to any vertices in $G - W$. As secluded vertices have to induce subpaths in σ, we can contract them and only consider non-secluded vertices of W in the following. By adapting the partial order, we prevent that some vertex of $G - W$ is added between the vertices of the edge that results from that contraction.

We try to insert vertices of $G - W$ into the ordering σ as rightmost as possible, i.e., we try to only insert such a vertex into the gap between w_{i-1} and w_i if it is a predecessor of w_i in π. However, it might be that w_{i-1} and w_i are not adjacent and all predecessors of w_i in $G - W$ are already visited. Then we have to take some other vertex from $G - W$ between w_{i-1} and w_i. For all $x \in V(G - W)$, we define $r(x)$ as the minimum index j such that x is to the left of w_j in the partial order. Among all remaining vertices of $G - W$ that are allowed to be to the left of w_i, we take a vertex with minimal $r(x)$ value. The details of the algorithm and the proof of correctness can be found in the full version [6]. ◀

5   Conclusion

We have presented a comprehensive classification of the parameterized complexity of finding Hamiltonian paths or cycles with precedence constraints from the perspective of graph width parameters (see Figure 1). We were able to prove that all our XP algorithms cannot be extended to FPT algorithms (unless W[1] = FPT). Nevertheless, our W[1]-hardness proofs cannot be used to show – under the assumption of the Exponential Time Hypothesis – that the algorithms' exponents are asymptotically optimal since these proofs increase the parameter value quadratically. Therefore, one may ask whether there are also FPT-reductions where the parameter increases only linearly.

So far, our work has focused on the existence of parameterized algorithms. As a next step, one could also consider the question whether the FPT results could be extended to kernels of polynomial size. As mentioned, there is a trivial polynomial kernel for *vertex cover number*. We are rather optimistic that such a kernel can also be given for *feedback edge set number*.

One could also try to generalize FPT or XP algorithms to distance parameters for more general graph classes. One option are the *series-parallel graphs*, a generalization of outerplanar graphs. These graphs are exactly the graphs of *treewidth* 2. Note that a generalization to *distance to treewidth 3* is hopeless as POHPP and POHCP are already para-NP-hard for *edge distance to bandwidth 3* [5]. Another option would be to consider graph width parameters not on all graphs but only on some subclass. In particular, the planar case of *distance to outerplanar* could be interesting as it lies between the general W[1]-hard case and the FPT-case where the deleted vertices all lie in the interior of the outer face.

As POHPP and POHCP are already hard for very restricted graphs, one may ask how their complexity behaves if we restrict both the graph and the partial order. It has been shown [7] that POHPP is in XP and W[1]-hard when parameterized by the partial order's *width*. Using similar arguments as for Observations 2.1 and 2.2, we can show that the same holds for POHCP. However, it is open what happens if the problems are parameterized by both the partial order's *width* and some graph width parameter such as *treewidth* or *distance to clique*.

─── **References** ───

1   Zakir Hussain Ahmed and S. N. Narahari Pandit. The travelling salesman problem with precedence constraints. *Opsearch*, 38(3):299–318, 2001. `doi:10.1007/BF03398638`.

2   Norbert Ascheuer, Laureano F. Escudero, Martin Grötschel, and Mechthild Stoer. A cutting plane approach to the sequential ordering problem (with applications to job scheduling in manufacturing). *SIAM Journal on Optimization*, 3(1):25–42, 1993. `doi:10.1137/0803002`.

3   Katerina Asdre and Stavros D. Nikolopoulos. The 1-fixed-endpoint path cover problem is polynomial on interval graphs. *Algorithmica*, 58(3):679–710, 2010. `doi:10.1007/s00453-009-9292-5`.

4   Katerina Asdre and Stavros D. Nikolopoulos. A polynomial solution to the *k*-fixed-endpoint path cover problem on proper interval graphs. *Theoretical Computer Science*, 411(7):967–975, 2010. `doi:10.1016/j.tcs.2009.11.003`.

5   Jesse Beisegel, Katharina Klost, Kristin Knorr, Fabienne Ratajczak, and Robert Scheffler. A graph width perspective on partially ordered Hamiltonian paths and cycles I: Treewidth, pathwidth, and grid graphs, 2025. `arXiv:2506.23790`.

6   Jesse Beisegel, Katharina Klost, Kristin Knorr, Fabienne Ratajczak, and Robert Scheffler. A graph width perspective on partially ordered Hamiltonian paths and cycles II: Vertex and edge deletion numbers, 2025. `arXiv:2510.08378`.

7   Jesse Beisegel, Fabienne Ratajczak, and Robert Scheffler. Computing Hamiltonian paths with partial order restrictions. *ACM Transactions on Computation Theory*, 17(1), 2025. `doi:10.1145/3711844`.

8   Benjamin Bergougnoux, Mamadou Moustapha Kanté, and O-joung Kwon. An optimal XP algorithm for Hamiltonian cycle on graphs of bounded clique-width. *Algorithmica*, 82(6):1654–1674, 2020. `doi:10.1007/S00453-019-00663-9`.

9   Lucio Bianco, Aristide Mingozzi, Salvatore Ricciardelli, and Massimo Spadoni. Exact and heuristic procedures for the traveling salesman problem with precedence constraints, based on dynamic programming. *INFOR: Information Systems and Operational Research*, 32(1):19–32, 1994. `doi:10.1080/03155986.1994.11732235`.

10 Hans L. Bodlaender, Marek Cygan, Stefan Kratsch, and Jesper Nederlof. Deterministic single exponential time algorithms for connectivity problems parameterized by treewidth. *Information and Computation*, 243:86–111, 2015. `doi:10.1016/J.IC.2014.12.008`.

11 Charles J. Colbourn and William R. Pulleyblank. Minimizing setups in ordered sets of fixed width. *Order*, 1:225–229, 1985. `doi:10.1007/BF00383598`.

12   Bruno Courcelle. The monadic second-order logic of graphs. I. Recognizable sets of finite graphs. *Information and Computation*, 85(1):12–75, 1990. `doi:10.1016/0890-5401(90)90043-H`.

13   Bruno Courcelle. The monadic second-order logic of graphs III: Tree-decompositions, minors and complexity issues. *Informatique Théorique et Applications/Theoretical Informatics and Applications*, 26(3):257–286, 1992. `doi:10.1051/ITA/1992260302571`.

14   Marek Cygan, Stefan Kratsch, and Jesper Nederlof. Fast Hamiltonicity checking via bases of perfect matchings. *Journal of the ACM*, 65(3):12:1–12:46, 2018. `doi:10.1145/3148227`.

15   Marek Cygan, Jesper Nederlof, Marcin Pilipczuk, Michał Pilipczuk, Johan M. M. van Rooij, and Jakub Onufry Wojtaszczyk. Solving connectivity problems parameterized by treewidth in single exponential time. *ACM Transactions on Algorithms*, 18(2):17:1–17:31, 2022. `doi:10.1145/3506707`.

16   Erik D. Demaine, David Eppstein, Adam Hesterberg, Kshitij Jain, Anna Lubiw, Ryuhei Uehara, and Yushi Uno. Reconfiguring undirected paths. In Zachary Friggstad, Jörg-Rüdiger Sack, and Mohammad R. Salavatipour, editors, *Algorithms and Data Structures – 16th International Symposium, WADS 2019*, volume 11646 of *LNCS*, pages 353–365. Springer, 2019. `arXiv:1905.00518`, `doi:10.1007/978-3-030-24766-9_26`.

17   Vladimir G. Deĭneko, Michael Hoffmann, Yoshio Okamoto, and Gerhard J. Woeginger. The traveling salesman problem with few inner points. *Operations Research Letters*, 34(1):106–110, 2006. `doi:10.1016/j.orl.2005.01.002`.

18   Reinhard Diestel. *Graph Theory*, volume 173 of *GTM*. Springer, fourth edition, 2012. `doi:10.1007/978-3-662-70107-2`.

19   Martin Doucha and Jan Kratochvíl. Cluster vertex deletion: A parameterization between vertex cover and clique-width. In Branislav Rovan, Vladimiro Sassone, and Peter Widmayer, editors, *Mathematical Foundations of Computer Science 2012 – 37th International Symposium, MFCS 2012*, volume 7464 of *LNCS*, pages 348–359. Springer, 2012. `doi:10.1007/978-3-642-32589-2_32`.

20   Maël Dumas, Anthony Perez, Mathis Rocton, and Ioan Todinca. Polynomial kernels for edge modification problems towards block and strictly chordal graphs. *Discrete Mathematics & Theoretical Computer Science*, 27(2), 2025. `doi:10.46298/DMTCS.12998`.

21   Heinz-Dieter Ebbinghaus and Jörg Flum. *Finite Model Theory*. Perspectives in Mathematical Logic. Springer, 1995. `doi:10.1007/978-3-662-03182-7`.

22   Laureano F. Escudero. An inexact algorithm for the sequential ordering problem. *European Journal of Operational Research*, 37(2):236–249, 1988. `doi:10.1016/0377-2217(88)90333-5`.

23   Laureano F. Escudero. On improving a solution to the ATSP with fixed origin and precedence relationships. *Trabajos de Investigacion Operativa*, 3:117–140, 1988. `doi:10.1007/BF02888669`.

24   Wolfgang Espelage, Frank Gurski, and Egon Wanke. How to solve NP-hard graph problems on clique-width bounded graphs in polynomial time. In Andreas Brandstädt and Van Bang Le, editors, *Graph-Theoretic Concepts in Computer Science, 27th International Workshop, WG 2001*, volume 2204 of *LNCS*, pages 117–128. Springer, 2001. `doi:10.1007/3-540-45477-2_12`.

25   Michael R. Fellows, Danny Hermelin, Frances Rosamond, and Stéphane Vialette. On the parameterized complexity of multiple-interval graph problems. *Theoretical Computer Science*, 410(1):53–61, 2009. `doi:10.1016/j.tcs.2008.09.065`.

26   Fedor V. Fomin, Petr A. Golovach, Nikola Jedličková, Jan Kratochvíl, Danil Sagunov, and Kirill Simonov. Path cover, Hamiltonicity, and independence number: An FPT perspective, 2025. `arXiv:2403.05943`.

27   Fedor V. Fomin, Petr A. Golovach, Daniel Lokshtanov, and Saket Saurabh. Intractability of clique-width parameterizations. *SIAM Journal on Computing*, 39(5):1941–1956, 2010. `doi:10.1137/080742270`.

28   Fedor V. Fomin, Petr A. Golovach, Daniel Lokshtanov, Saket Saurabh, and Meirav Zehavi. Clique-width III: Hamiltonian cycle and the odd case of graph coloring. *ACM Transactions on Algorithms*, 15(1), 2018. `doi:10.1145/3280824`.

29   Jakub Gajarský, Michael Lampis, and Sebastian Ordyniak. Parameterized algorithms for modular-width. In Gregory Z. Gutin and Stefan Szeider, editors, *Parameterized and Exact Computation – 8th International Symposium, IPEC 2013*, volume 8246 of *LNCS*, pages 163–176. Springer, 2013. `doi:10.1007/978-3-319-03898-8_15`.

30   Robert Ganian. Improving vertex cover as a graph parameter. *Discrete Mathematics & Theoretical Computer Science*, 17(2):77–100, 2015. `doi:10.46298/dmtcs.2136`.

31   Paul C. Gilmore, Eugene L. Lawler, and David B. Shmoys. Well-solved special cases. In Eugene L. Lawler, Jan Karel Lenstra, Alexander H. G. Rinnooy Kan, and David B. Shmoys, editors, *The traveling salesman problem. A guided tour of combinatorial optimization*, pages 87–143. Wiley, Chichester, 1985.

32   Petr A. Golovach, R. Krithika, Abhishek Sahu, Saket Saurabh, and Meirav Zehavi. Graph Hamiltonicity parameterized by proper interval deletion set. In Yoshiharu Kohayakawa and Flávio Keidi Miyazawa, editors, *LATIN 2020: Theoretical Informatics – 14th Latin American Symposium*, volume 12118 of *LNCS*, pages 104–115. Springer, 2020. `doi:10.1007/978-3-030-61792-9_9`.

33   Alon Itai, Christos H. Papadimitriou, and Jayme Luiz Szwarcfiter. Hamilton paths in grid graphs. *SIAM Journal on Computing*, 11(4):676–686, 1982. `doi:10.1137/0211056`.

34   Bart M. P. Jansen. *The Power of Data Reduction. Kernels for Fundamental Graph Problems*. PhD thesis, Utrecht University, 2013. URL: `https://hdl.handle.net/1874/276438`.

35   Bart M. P. Jansen, László Kozma, and Jesper Nederlof. Hamiltonicity below Dirac's condition. In Ignasi Sau and Dimitrios M. Thilikos, editors, *Graph-Theoretic Concepts in Computer Science – 45th International Workshop, WG 2019*, volume 11789 of *LNCS*, pages 27–39. Springer, 2019. `doi:10.1007/978-3-030-30786-8_3`.

36   Bernhard Korte and Jens Vygen. *Combinatorial Optimization: Theory and Algorithms*. Springer, 6th edition, 2018. `doi:10.1007/978-3-662-56039-6`.

37   Michael Lampis. Algorithmic meta-theorems for restrictions of treewidth. *Algorithmica*, 64(1):19–37, 2012. `doi:10.1007/S00453-011-9554-X`.

38   Peng Li and Yaokun Wu. A linear time algorithm for the 1-fixed-endpoint path cover problem on interval graphs. *SIAM Journal on Discrete Mathematics*, 31(1):210–239, 2017. `doi:10.1137/140981265`.

39   George B. Mertzios and Walter Unger. An optimal algorithm for the *k*-fixed-endpoint path cover on proper interval graphs. *Mathematics in Computer Science*, 3(1):85–96, 2010. `doi:10.1007/s11786-009-0004-y`.

40   Burkhard Monien and Ivan Hal Sudborough. Bounding the bandwidth of NP-complete problems. In Hartmut Noltemeier, editor, *Graphtheoretic Concepts in Computer Science – International Workshop, WG 80*, volume 100 of *LNCS*, pages 279–292. Springer, 1980. `doi:10.1007/3-540-10291-4_20`.

41   Jaroslav Nešetřil and Patrice Ossona de Mendez. *Sparsity. Graphs, Structures, and Algorithms*, volume 28 of *Algorithms and Combinatorics*. Springer, 2012. `doi:10.1007/978-3-642-27875-4`.

42   Sophie N. Parragh, Karl F. Doerner, and Richard F. Hartl. A survey on pickup and delivery problems. Part I: Transportation between customers and depot. *Journal für Betriebswirtschaft*, 58(1):21–51, 2008. `doi:10.1007/s11301-008-0033-7`.

43   Sophie N. Parragh, Karl F. Doerner, and Richard F. Hartl. A survey on pickup and delivery problems. Part II: Transportation between pickup and delivery locations. *Journal für Betriebswirtschaft*, 58(2):81–117, 2008. `doi:10.1007/s11301-008-0036-4`.

44   Krzysztof Pietrzak. On the parameterized complexity of the fixed alphabet shortest common supersequence and longest common subsequence problems. *Journal of Computer and System Sciences*, 67(4):757–771, 2003. `doi:10.1016/S0022-0000(03)00078-3`.

45   Harilaos N. Psaraftis. A dynamic programming solution to the single vehicle many-to-many immediate request dial-a-ride problem. *Transportation Science*, 14(2):130–154, 1980. `doi:10.1287/trsc.14.2.130`.

46　Sigve Hortemo Sæther and Jan Arne Telle. Between treewidth and clique-width. *Algorithmica*, 75(1):218–253, 2016. `doi:10.1007/S00453-015-0033-7`.

47　Sigve Hortemo Sæther. Solving Hamiltonian Cycle by an EPT algorithm for a non-sparse parameter. *Discrete Applied Mathematics*, 228:88–97, 2017. `doi:10.1016/j.dam.2016.02.008`.

48　Egon Wanke. k-NLC graphs and polynomial algorithms. *Discrete Applied Mathematics*, 54(2):251–266, 1994. `doi:10.1016/0166-218X(94)90026-4`.