Hamiltonicity Parameterized by Mim-Width Is (Indeed) Para-NP-Hard

Benjamin Bergougnoux
Aix-Marseille Université, CNRS, LIS, Marseille, France

Lars Jaffke
NHH Norwegian School of Economics, Bergen, Norway

Abstract

We prove that Hamiltonian Path and Hamiltonian Cycle are NP-hard on graphs of linear mim-width 26, even when a linear order of the input graph with mim-width 26 is provided together with input. This fills a gap left by a broken proof of the para-NP-hardness of Hamiltonicity problems parameterized by mim-width.

2012 ACM Subject Classification Theory of computation → Problems, reductions and completeness; Mathematics of computing → Graph algorithms

Keywords and phrases Hamiltonian Path, Hamiltonian Cycle, Mim-Width, Para-NP-Hardness

Digital Object Identifier 10.4230/LIPIcs.IPEC.2025.31

Related Version *Full Version*: https://arxiv.org/abs/2507.00612

Acknowledgements We thank Richard Wols for the counterexample presented in Section 2.1.

1 Introduction

Maximum induced matching width (mim-width) which was introduced in 2012 by Vatshelle [28] is by now a common staple among width parameters of graphs (e.g. [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26]). One pillar in the understanding of its algorithmic use and limitations is that finding simply structured *induced* subgraphs tends to be tractable (solvable in XP time by the width of a given decomposition), while finding *non-induced* subgraphs is usually hard (NP-hard for constant mim-width). Most notably, finding long induced paths and cycles is in XP [6], while in [21], it has been claimed that Hamiltonian Cycle is NP-hard on graphs of linear mim-width 1. The latter claim rests on a proof that Hamiltonian Cycle is NP-hard on rooted directed path graphs given in [27], and in [21] it is shown that the graphs resulting from this reduction have linear mim-width 1. Unfortunately, as we show below, the reduction from [27] has a flaw, which means that the complexity of Hamiltonicity problems (or, more generally, problems of finding long non-induced paths or cycles) parameterized by mim-width remained open. In this paper, we repair this gap in the literature by proving the following result.

▶ **Theorem 1.** *The Hamiltonian Path and Hamiltonian Cycle problems are NP-hard on graphs of linear mim-width 26, even when a linear order of the input graph of mim-width 26 is provided as part of the input.*

Observe that hardness holds even when a linear order of small mim-width is provided at the input. Recently, Bergougnoux, Bonnet, and Duron [3] showed that computing the mim-width of a graph exactly is para-NP-hard as well. Note however that there might still be some f such that there is an XP time $f(k)$-approximation algorithm to mim-width k.

We would like to mention that the constant on the mim-width in our statements can likely be improved, and it is an interesting question where exactly the boundary between tractability and intractability lies for Hamiltonicity problems. Our results very much allow the possibility that Hamiltonicity is polynomial-time solvable on rooted directed path graphs or more generally, on graphs of mim-width 1. For Clique and a number of related problems, Otachi, Suzuki, and Tamura [26] recently showed that the boundary between tractable and intractable cases lies between mim-width 1 and 2. While Clique was known to be solvable in polynomial time on graphs of mim-width 1 (given a decomposition), they showed that it becomes NP-hard on graphs of mim-width 2. It would be interesting to establish a similar dividing line for Hamiltonicity problems.

Proofs of statements marked with "♠" are deferred to the full version.

2 Preliminaries

We denote the set of natural numbers by $\mathbb{N} = \{0, 1, \ldots\}$, and for $n \in \mathbb{N}$, we use the shorthand $[n] = \{1, \ldots, n\}$. All graphs considered in this paper are finite, undirected, and simple. For a graph G, we denote by $V(G)$ its vertices and by $E(G)$ its edges. For a vertex $v \in V(G)$, we denote its neighborhood by $N_G(v) = \{u \mid uv \in E(G)\}$. The *degree* of v is $|N_G(v)|$. For a set $X \subseteq V(G)$, the *subgraph of G induced by X*, denoted by $G[X]$, is the graph on vertex set X and edge set $\{uv \mid uv \in E(G), \{u, v\} \subseteq X\}$. A set $X \subseteq V(G)$ is *independent* if $G[X]$ has no edges and X is a *clique* if every pair of vertices in X is adjacent in G.

A *walk* in a graph G is a sequence of vertices $W = (v_1, \ldots, v_r)$ such that for all $i \in [r-1]$, $v_i v_{i+1} \in E(G)$. If all vertices in W are pairwise distinct, then W is called a *path*, and if all vertices in $\{v_1, \ldots, v_{r-1}\}$ are pairwise distinct and $v_1 = v_r$, then W is a *cycle*. A path/cycle is *Hamiltonian* if it contains all vertices of G.

A graph G is *bipartite* if there is a partition (A, B) of $V(G)$ such that A and B are independent. G is further called a *chain graph* if there is a linear order $a_1, \ldots, a_n$ of A such that for all $i \in [n-1]$: $N(a_i) \subseteq N(a_{i+1})$.

Let G be a graph. Two distinct vertices $u, v \in V(G)$ are *false twins* if $N_G(u) = N_G(v)$ and $uv \notin E(G)$. Let $uv \in E(G)$. The *subdivision of uv* is the operation of replacing uv with a path (u, x, v) where x is a newly created vertex.

Linear Mim-width. Let G be a graph and $A \subseteq V(G)$. We let $\overline{A} = V(G) \setminus A$. We denote by $G[A, \overline{A}]$ the bipartite subgraph of G with vertex bipartition $(A, \overline{A})$ whose edges are $\{ab \mid ab \in E(G), a \in A, b \in \overline{A}\}$. A set of edges $M = \{a_i b_i \mid i \in [r]\} \subseteq E(G)$ is a *matching* if all vertices in $V(M) = \bigcup_{i \in [r]} \{a_i, b_i\}$ are pairwise distinct. A matching M is *induced* if $E(G[V(M)]) = M$. The *mim-value* of a set $A \subseteq V(G)$, denoted by $\mathsf{mim}_G(A)$, is the largest size of any induced matching in $G[A, \overline{A}]$. Let $\lambda = (v_1, \ldots, v_n)$ be a linear order of $V(G)$. The *maximum induced matching width (mim-width) of* λ is $\max_{i \in [n]} \mathsf{mim}_G(\{v_1, \ldots, v_i\})$.

The following observation will be helpful in a later proof.

▶ **Observation 2.** *Let H be a chain graph with bipartition (A, B). Then, $\mathsf{mim}_H(A) \leq 1$.*

2.1 Counterexample to the existing reduction

We recall the reduction from [27]. It is from Hamiltonian Cycle on bipartite graphs of maximum degree 3 and minimum degree 2 to Hamiltonian Cycle on rooted directed path graphs, which are intersection graphs of directed paths in a rooted tree.

Let G be a bipartite graph of maximum degree 3 and minimum degree 2, and let (A, B) be the vertex bipartition of G. We may assume that $|A| = |B| = n$, and fix arbitrary orderings on A and B, that is, we let $A = \{a_1, \ldots, a_n\}$ and $B = \{b_1, \ldots, b_n\}$ throughout. We construct a graph H as follows.

1. Let H be a copy of G.
2. Subdivide each edge $a_ib_j \in E(H)$ once. For convenience, we use a_ib_j to both denote the edge in G and the vertex in H created in the subdivision of this edge.
3. Make $\{a_ib_j \mid i, j \in [n], a_ib_j \in E(G)\}$ a clique in H.
4. For each $i \in [n]$, make a_i adjacent to all vertices $a_{i'}b_j \in V(H)$ where $1 \leq i' \leq i$, $j \in [n]$, $a_{i'}b_j \in E(G)$.
5. For each $i \in [n]$ such that b_i has degree 3 in G, add the vertex b'_i with $N_H(b'_i) = N_H(b_i)$ to H.

In [21] it is shown that the graph H as constructed above has linear mim-width 1.

In Figure 1, we show a bipartite graph G of maximum degree 3 and minimum degree 2 that has no Hamiltonian cycle, while the graph obtained from running the above reduction on input G has one, thus providing a counterexample to the proof in [27].

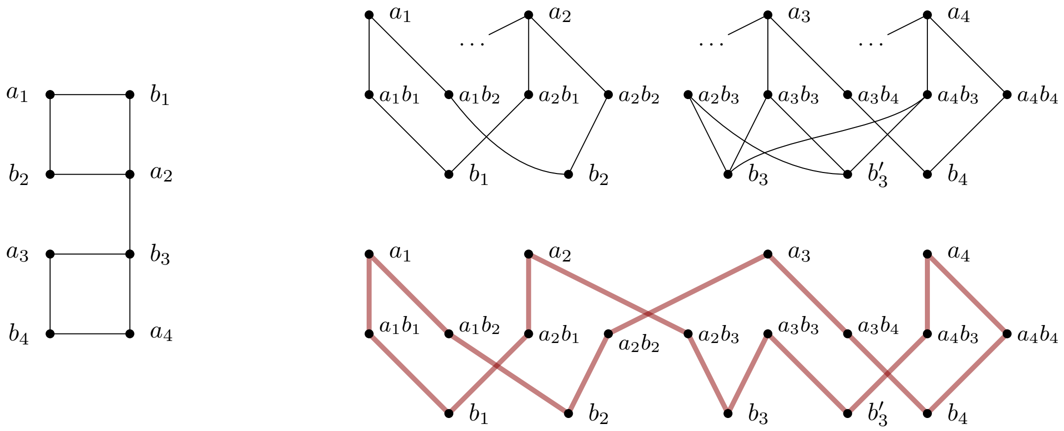

Figure 1 Counterexample to the construction from [27]. The three dots symbolize adjacency to all vertices to the left on the middle layer.

2.2 3-SAT

A 3-*CNF formula* is a Boolean formula in conjunctive normal form such that each clause has at most three pairwise distinct literals. Given a formula ϕ, we denote by $\mathsf{Var}(\phi)$ its set of variables. A truth assignment to the variables of ϕ is a a function $\alpha\colon \mathsf{Var}(\phi) \to \{0, 1\}$. We say that a clause C is satisfied by α if C contains a literal ℓ with variable x such that $\alpha(x) = 1$ if $\ell = x$ and $\alpha(x) = 0$ if $\ell = \overline{x}$. Moreover, α is said to be a satisfying assignment of ϕ if it satisfies all the clauses of ϕ. We will reduce from the following problem.

3-SAT	
Input:	A 3-CNF formula ϕ.
Question:	Does ϕ admit a satisfying assignment?

3 Main result

In this section, we prove that the following problem is NP-hard for parameter value $w = 25$.

Hamiltonian Path/Linear Mim-Width	
Input:	A graph G and a linear order λ of $V(G)$.
Parameter:	The mim-width w of λ.
Question:	Does G have a Hamiltonian path?

We give a polynomial-time reduction from the 3-SAT problem. Let ϕ be a 3-CNF formula with variables $\mathsf{Var}(\phi) = \{x_1, \ldots, x_n\}$ and clauses $C_1, \ldots, C_m$.

Variable gadgets. For each $i \in [n]$, we create a variable gadget as follows. It consists of a sequence of m cycles on six vertices such that traversing each cycle clockwise corresponds to setting x_i to true, and traversing them counterclockwise corresponds to setting x_i to false. The construction ensures that for each variable, all cycles are traversed in the same way.

Let $i \in [n]$. We show how to construct the variable gadget $\mathtt{V}_i$. For each $j \in [m]$, it contains a cycle D_i^j whose vertices (listed in order) are called $0\text{-}in(i,j)$, $0\text{-}out(i,j)$, $z_{out}(i,j)$, $1\text{-}out(i,j)$, $1\text{-}in(i,j)$, $z_{in}(i,j)$. For each $j \in [m-1]$, we add an edge between $1\text{-}out(i,j)$ and $1\text{-}in(i,j+1)$, and between $0\text{-}out(i,j)$ and $0\text{-}in(i,j+1)$. See Figure 2 for an illustration. Moreover, there is a vertex s_i, adjacent to $0\text{-}in(i,1)$ and $1\text{-}in(i,1)$, and a vertex t_i, adjacent to $0\text{-}out(i,m)$ and $1\text{-}out(i,m)$.

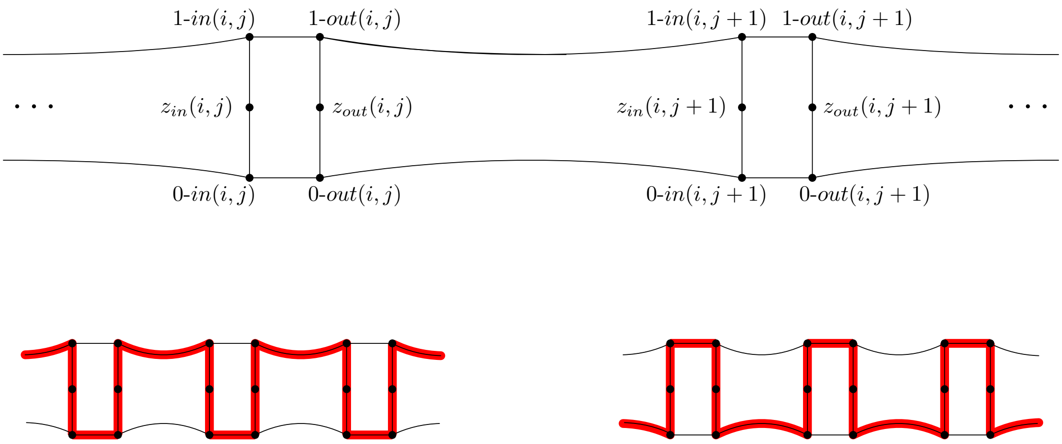

Figure 2 Illustration of the variable gadget. Bottom left (resp., right): a traversal of the sequence corresponding to setting the variable to false (resp., true).

Clause gadgets. For each clause C_j, we add a gadget $\mathtt{C}_j$ with the following property. Say variable x_i appears in C_j. If a given truth assignment to x_i satisfies C_j, then we can enter and collect the vertices of $\mathtt{C}_j$ from D_i^j (in the variable gadget $\mathtt{V}_i$) under the traversal of D_i^j corresponding to that truth assignment to x_i. To avoid "cheating", we need $\mathtt{C}_j$ to have the following property: once a path enters a vertex of $\mathtt{C}_j$, then we have to immediately collect all vertices from $\mathtt{C}_j$, and leave again via a prescribed exit, depending on where we entered $\mathtt{C}_j$. The following clause gadget due to Cygan, Kratsch, and Nederlof [14] achieves just that. We visualize these graphs in Figure 3.

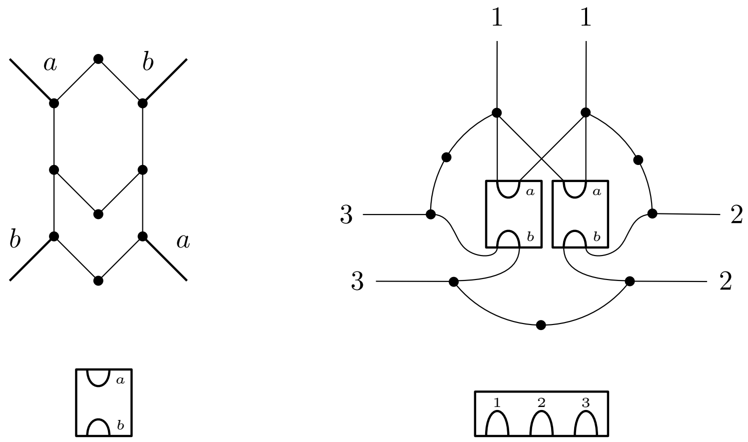

Figure 3 The clause gadgets due to Cygan, Kratsch, and Nederlof [14]. On the left, a gadget such that each Hamiltonian path that enters it via an edge labeled a (resp., b) must immediately collect all vertices in the gadget and leave via the other edge labeled a (resp, b). This gadget can be used for clauses of size two. On the right, a gadget for a clause of size three, with three entry-exit pairs and the analogous functionality.

▶ **Lemma 3** (Cygan, Kratsch, and Nederlof [14]). *For each $k \in \{2, 3\}$, there is a graph Γ_k on at most 27 vertices with k distinguished pairs of unique vertices $\{(\sigma_1, \tau_1), \ldots, (\sigma_k, \tau_k)\}$ with the following property. Let G be a graph with an induced subgraph Γ_k, and let P be a Hamiltonian path of G. Then, $P \cap \Gamma_k$ is a (σ_i, τ_i)-path for some $i \in [k]$.*

Main construction. We now construct the graph G from the formula ϕ, see Figure 4 for an illustration. First, we add two vertices s and t to G, which will have degree one and are therefore the endpoints of any Hamiltonian path in G, if one exists. For each $i \in [n]$, we add a variable gadget $\mathtt{V}_i$. We furthermore add the edges ss_1, t_nt, and for all $i \in [n-1]$, we proceed as follows. We add a vertex p_{i+1} to G, and add the edges t_ip_{i+1} and $p_{i+1}s_{i+1}$.

Next, for each clause C_j with literals $\ell_1, \ldots, \ell_k$, we proceed as follows. We let the clause gadget $\mathtt{C}_j$ be a copy of Γ_k from Lemma 3; let (σ_1, τ_1), …, (σ_k, τ_k) be its distinguished vertices. For each $h \in [k]$, let x_{j_h} be the variable of literal ℓ_h. If x_{j_h} appears positively in C_j, then we add the edges $\{\sigma_h, 0\text{-}in(i,j)\}$ and $\{\tau_h, 0\text{-}out(i,j)\}$, and if x_{j_h} appears negated in C_j, then we add $\{\sigma_h, 1\text{-}in(i,j)\}$ and $\{\tau_h, 1\text{-}out(i,j)\}$. This finishes the main part of the construction.

Dummy edges. At this point, G may have arbitrarily high mim-width. To reduce the mim-width of the graph, we add lots of dummy edges. These edges are introduced in such a way that they decrease the mim-width without allowing to "cheat". That is, no Hamiltonian path of G will ever use any of these edges. We proceed as follows. For each $j \in [m-1]$, each $i \in [2..n]$, and each $h < i$, we add the edges $\{b_1\text{-}out(i,j), b_2\text{-}in(h,j+1)\}$, for any $b_1, b_2 \in \{0,1\}$. Moreover, for each $i, j \in [t]$ with $i < j$, we add the edge $\{t_i, p_j\}$.

This finishes the construction. Clearly, this reduction can be performed in polynomial time. We show that it is correct.

▶ **Lemma 4.** *If ϕ is satisfiable, then G has a Hamiltonian path.*

Proof. Suppose that ϕ has a satisfying assignment $\alpha\colon \mathsf{Var}(\phi) \to \{0,1\}$. We construct a Hamiltonian path P of G as follows. See Figure 5 for an illustration of this construction. First, P starts in s, s_1. If $\alpha(x_1) = 1$, then the next vertex of P is $1\text{-}in(1,1)$, and if $\alpha(x_1) = 0$, then the next vertex of P is $0\text{-}in(1,1)$. Suppose the former is the case, the latter case is symmetric.

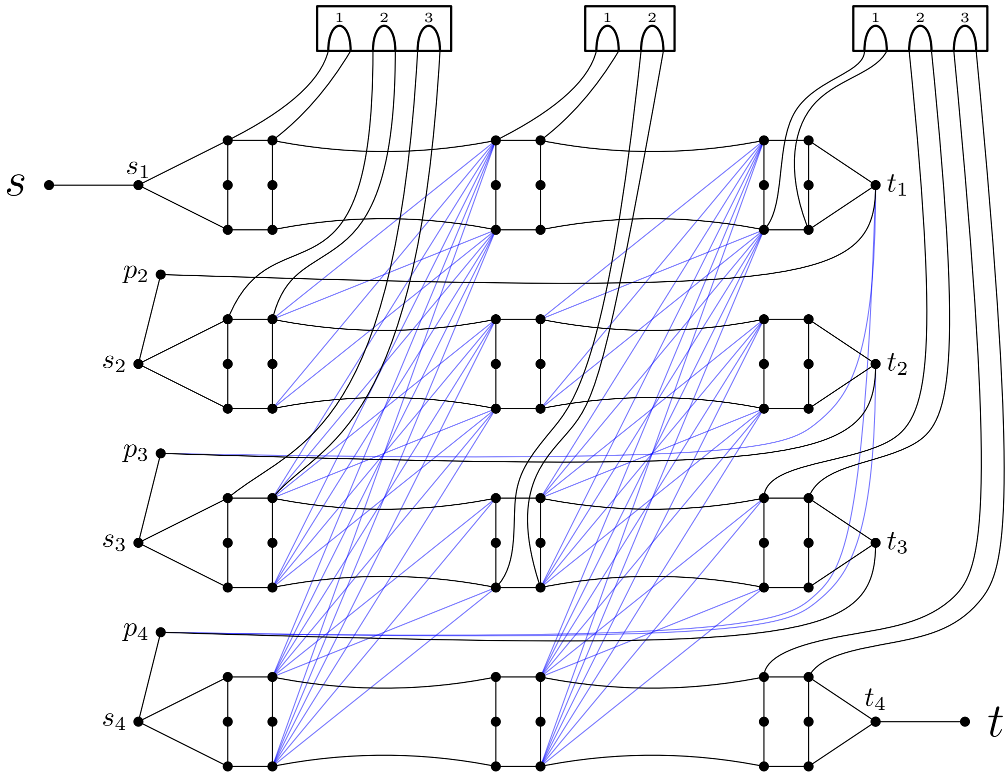

Figure 4 Overview of the construction. This is the example for the formula $(x_1 \vee x_2 \vee x_3) \wedge (x_1 \vee \overline{x_3}) \wedge (\overline{x_1} \vee x_3 \vee x_4)$. The dummy edges are depicted in blue.

The next vertices on P are $z_{in}(1,1)$, 0-$in(1,1)$. Now, if x_1 appears positively in C_1, then by construction, we can enter the gadget $\mathtt{C}_1$ from 0-$in(1,1)$, collect all its vertices, and leave it to arrive at 0-$out(1,1)$. Note that in this case, x_1's truth assignment satisfied C_1. P then collects $z_{out}(1,1)$, 1-$out(1,1)$, and takes the edge to 1-$in(1,2)$. We proceed analogously in sequence on all D_1^j, until we reach t_1. Here we follow the edge to p_2, s_2, and proceed on $\mathtt{V}_2$ as we did on $\mathtt{V}_1$, now based on which value α assigns x_2. We repeat this process until we finally reach t.

Whenever the truth assignment $\alpha(x_i)$ satisfies a clause C_j, we are able to collect the vertices of $\mathtt{C}_j$. Therefore, the procedure described above indeed produces a Hamiltonian path of G. ◀

Let $s, t \in V(G)$, and let P be an (s,t)-path in G. For distinct $u, v \in V(P)$, we write $u \prec_P v$ if the path from s to v contains u. For two sets $X, Y \subseteq V(G)$, we write $X \prec_P Y$ if for all $x \in X$, $y \in Y$, $x \prec_P y$. For intuition on the following notions, recall Figure 2.

▶ **Definition 5.** *Let $a \in [n]$ and $b \in [m]$, and let P be a Hamiltonian path of G starting in s. We say that P* traverses $D_a^1, \ldots, D_a^b$ in true-order *if P contains the following as subpaths for every $j \in [b]$*

(P_{in}) 1-$in(a,j)$, $z_{in}(a,j)$, 0-$in(a,j)$

(P_{out}) 0-$out(a,j)$, $z_{out}(a,j)$, 1-$out(a,j)$

and for every $j \in [b-1]$, P contains the edge $\{$1-$out(i,j)$, 1-$in(i,j+1)\}$. We say that P traverses $D_a^1, \ldots, D_a^b$ in false-order *if P contains the following as subpaths:*

(P_{in}) 0-$in(a,j)$, $z_{in}(a,j)$, 1-$in(a,j)$

(P_{out}) 1-$out(a,j)$, $z_{out}(a,j)$, 0-$out(a,j)$

and for every $j \in [b-1]$, P contains the edge $\{$0-$out(i,j)$, 0-$in(i,j+1)\}$.

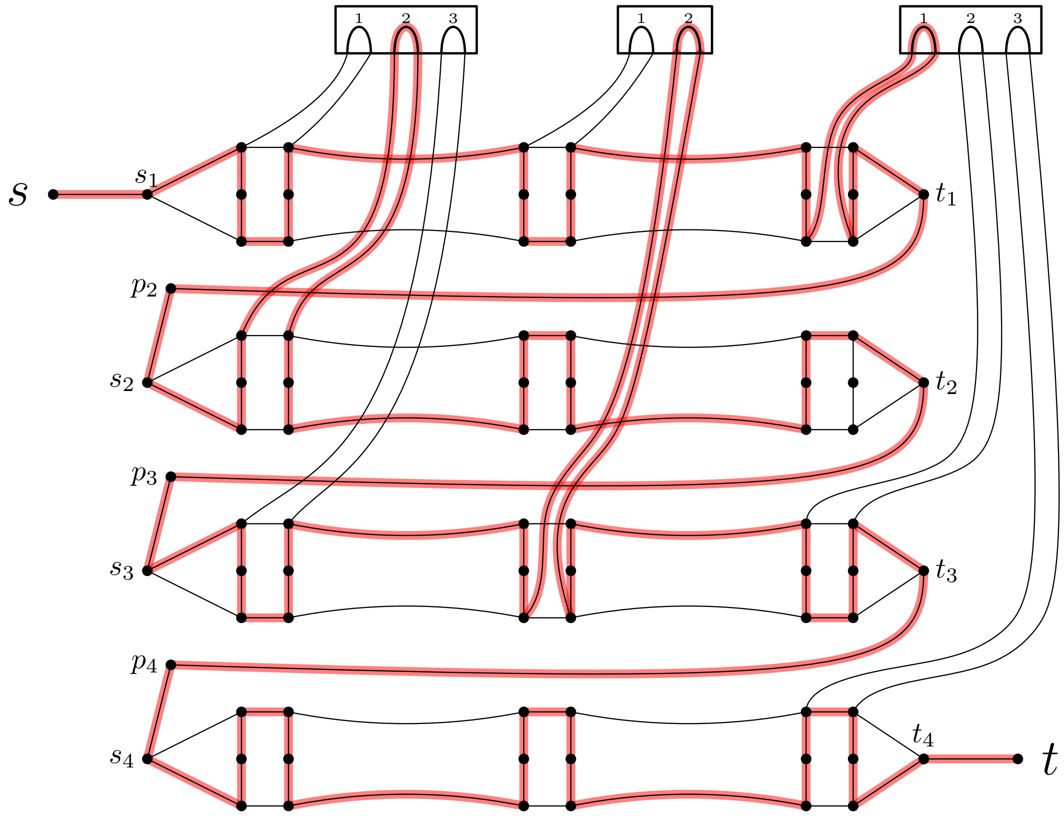

Figure 5 Example of the Hamiltonian path – of the graph presented in Figure 4 – constructed in the proof of Lemma 4 from the satisfying assignment α with $\alpha(x_1) = \alpha(x_3) = 0$ and $\alpha(x_2) = \alpha(x_4) = 1$. We omit the dummy edges to improve the legibility.

▶ **Definition 6.** *Let $a \in [n], b \in [m]$ and let P be a Hamiltonian (s,t)-path of G. We say that P is (a,b)-*respectable *if the following are satisfied.*

1. *For all $i \in [n]$ with $i < a$, we have $V(D_i^1) \prec_P V(D_i^2) \prec_P \cdots \prec_P V(D_i^m)$.*
2. *We have $V(D_a^1) \prec_P \cdots \prec_P V(D_a^b)$.*
3. *We have $V(\mathtt{V}_1) \prec_P \cdots \prec_P V(\mathtt{V}_{a-1}) \prec_P V(\mathtt{V}_a)$.*
4. *For every $i < a$, P traverses $D_i^1, \ldots, D_i^m$ in either true order or false order.*
5. *P traverses $D_a^1, \ldots, D_a^b$ in either true order or false order.*

▶ **Lemma 7** (♠)**.** *Let P be a Hamiltonian (s,t)-path of G. Then, P is (n,m)-respectable.*

▶ **Lemma 8.** *If G has a Hamiltonian path, then ϕ is satisfiable.*

Proof. Let P be a Hamiltonian path in G. By Lemma 7, P is (n,m)-respectable and in particular, for every $i \in [n]$, P traverses $D_i^1, \ldots, D_i^m$ in either true-order or false-order. We construct a truth assignment $\alpha\colon \mathsf{Var}(\phi) \to \{0,1\}$ as follows. For each $i \in [n]$, if P traverses $D_i^1, \ldots, D_i^m$ in true-order, we set $\alpha(x_i) = 1$, and if P traverses $D_i^1, \ldots, D_i^m$ in false-order, we set $\alpha(x_i) = 0$. Let $j \in [m]$. Since P is a Hamiltonian path, it collects the vertices in $\mathtt{C}_j$. By Lemma 3, there is some $i \in [n]$ such that P contains a path starting with a vertex in D_i^j, traversing all vertices in $\mathtt{C}_j$, and ending with a vertex in D_i^j. By construction, the two vertices in D_i^j are 0-$in(i,j)$, 0-$out(i,j)$, if x_i appears positively in C_j, and 1-$in(i,j)$, 1-$out(i,j)$, if x_i appears negated in C_j. Suppose without loss of generality that the former is the case. For P

to contain the described subpath, it must have entered D_i^j at vertex 1-$in(i,j)$, which implies that P traverses $D_i^1, \dots, D_i^m$ in true-order. By our construction, $\alpha(x_i) = 1$, meaning that the literal of C_j containing x_i satisfies C_j. ◀

▶ **Lemma 9.** *We can construct in polynomial time a linear order on $V(G)$ of mim-width at most* 25.

Proof. Let λ be any linear order on $V(G)$ extending the following partial order:

$$\begin{aligned}
s, s_1, & D_1^1, p_2, s_2, D_2^1, \dots, p_n, s_n, D_n^1, V(\mathtt{C}_1), \dots, \\
& D_1^2, D_2^2, \dots, D_n^2, V(\mathtt{C}_2), \\
& \dots, \\
& D_1^{m-1}, D_2^{m-1}, \dots, D_n^{m-1}, V(\mathtt{C}_{m-1}), \\
& D_1^m, t_1, D_2^m, t_2, \dots, D_n^m, t_n, t, V(\mathtt{C}_m)
\end{aligned}$$

Throughout the following, we let $D = \bigcup_{(i,j)\in[n]\times[m]} D_i^j$, $P = \{p_i \mid i \in [n]\}$, $S = \{s\} \cup \{s_i \mid i \in [n]\}$, $T = \{t_i \mid i \in [n]\} \cup \{t\}$. Let $(A, \overline{A})$ be any cut induced by λ (such that A contains the first t vertices of λ for some $t \leq |V(G)|$), and let M be an induced matching in $G[A, \overline{A}]$. Each edge of M is one of the following types, and below we justify the bounds on their number indicated in the respective parenthesis.

1. Between $S \cup P$ and $S \cup T$ (≤ 1).
2. Between $S \cup T$ and D (≤ 1).
3. Between D and $V(\mathtt{C}_j)$, for some $j \in [m]$ (≤ 6).
4. Inside $V(\mathtt{C}_j)$ for some $j \in [m]$ (≤ 13).
5. Inside D (≤ 4).

For edges of type 1, note that $G[A \cap (S \cup P), \overline{A} \cup (S \cup T)]$ is a chain graph plus possibly one edge $p_i s_i$, for some $i \in [n]$. Recall by Observation 2, M can contain at most one edge from any chain graph. Furthermore, if $p_i s_i \in M$, then M has no other edge from this subgraph, since in this case, $p_i \in A$ is adjacent to all vertices in $T \cap \overline{A}$ that have a neighbor in $(S \cup P) \cap A$. So in either case, there is at most one edge of type 1 in M.

For edges of type 2, such edges can only occur if either some s_i is the last vertex of A or if some t_i is the first vertex of $\overline{A}$; in any case, there is at most one such edge in M. The number of edges of types 3 and 4 are trivial upper bounds.

Lastly, consider edges of type 5. Let (j, i) be the lexicographically largest pair such that A contains vertices from D_i^j. First, there are at most two edges from D_i^j itself in M. Next, consider the edges in M that are between $D^j = \bigcup_{i\in[n]} D_i^j$ and $D^{j+1} = \bigcup_{i\in[n]} D_i^{j+1}$. Suppose there is some $h \in [n]$ such that there is an edge $d_j d_{j+1}$ in M, where $d_j \in D_h^j$ and $d_{j+1} \in D_h^{j+1}$. Note that there are at most two such edges per h. Moreover, in this case, M has no other edges between D^j and D^{j+1}, since for each such edge, at least one endpoint has a neighbor in $\{d_j, d_{j+1}\}$. So in this case, there are at most four edges of type 5 in total. If there is no such h, then by a similar argument, M contains at most one edge between D^j and D^{j+1}, in which case we have at most three edges of type 5. ◀

This finishes the proof for Hamiltonian Path. By adding another degree-two vertex adjacent to only s and t, we also obtain the same result for the Hamiltonian Cycle problem and the fact that adding a vertex to a graph increase its mim-width by at most 1.

▶ **Theorem 1.** *The Hamiltonian Path and Hamiltonian Cycle problems are* NP*-hard on graphs of linear mim-width 26, even when a linear order of the input graph of mim-width* 26 *is provided as part of the input.*

References

1 Brage I. K. Bakkane and Lars Jaffke. On the hardness of generalized domination problems parameterized by mim-width. In Holger Dell and Jesper Nederlof, editors, *Proceedings of the 17th International Symposium on Parameterized and Exact Computation, IPEC 2022*, volume 249 of *LIPIcs*, pages 3:1–3:19. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2022. doi:10.4230/LIPICS.IPEC.2022.3.

2 Rémy Belmonte and Martin Vatshelle. Graph classes with structured neighborhoods and algorithmic applications. *Theor. Comput. Sci.*, 511:54–65, 2013. doi:10.1016/J.TCS.2013.01.011.

3 Benjamin Bergougnoux, Édouard Bonnet, and Julien Duron. Mim-width is paranp-complete. In Keren Censor-Hillel, Fabrizio Grandoni, Joël Ouaknine, and Gabriele Puppis, editors, *Proceedings of the 52nd International Colloquium on Automata, Languages, and Programming, ICALP 2025*, volume 334 of *LIPIcs*, pages 25:1–25:17. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2025. doi:10.4230/LIPICS.ICALP.2025.25.

4 Benjamin Bergougnoux, Jan Dreier, and Lars Jaffke. A logic-based algorithmic meta-theorem for mim-width. In Nikhil Bansal and Viswanath Nagarajan, editors, *Proceedings of the 34th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA 2023)*, pages 3282–3304. SIAM, 2023. doi:10.1137/1.9781611977554.ch125.

5 Benjamin Bergougnoux, Thekla Hamm, Lars Jaffke, and Paloma T. Lima. On algorithmic applications of $\mathcal{F}$-branchwidth. In *ESA 2025*, 2025. To appear.

6 Benjamin Bergougnoux and Mamadou Moustapha Kanté. More applications of the d-neighbor equivalence: Acyclicity and connectivity constraints. *SIAM J. Discret. Math.*, 35(3):1881–1926, 2021. doi:10.1137/20M1350571.

7 Benjamin Bergougnoux, Charis Papadopoulos, and Jan Arne Telle. Node multiway cut and subset feedback vertex set on graphs of bounded mim-width. *Algorithmica*, 84(5):1385–1417, 2022. doi:10.1007/S00453-022-00936-W.

8 Hans L. Bodlaender, Carla Groenland, Hugo Jacob, Lars Jaffke, and Paloma T. Lima. XNLP-Completeness for parameterized problems on graphs with a linear structure. *Algorithmica*, 87(4):465–506, 2025. doi:10.1007/S00453-024-01274-9.

9 Flavia Bonomo-Braberman, Nick Brettell, Andrea Munaro, and Daniël Paulusma. Solving problems on generalized convex graphs via mim-width. *J. Comput. Syst. Sci.*, 140:103493, 2024. doi:10.1016/J.JCSS.2023.103493.

10 Nick Brettell, Jake Horsfield, Andrea Munaro, Giacomo Paesani, and Daniël Paulusma. Bounding the mim-width of hereditary graph classes. *J. Graph Theory*, 99(1):117–151, 2022. doi:10.1002/JGT.22730.

11 Nick Brettell, Jake Horsfield, Andrea Munaro, and Daniël Paulusma. List k-colouring P_t-free graphs: A mim-width perspective. *Inf. Process. Lett.*, 173:106168, 2022. doi:10.1016/J.IPL.2021.106168.

12 Nick Brettell, Andrea Munaro, Daniël Paulusma, and Shizhou Yang. Comparing width parameters on graph classes. *Eur. J. Comb.*, 127:104163, 2025. doi:10.1016/J.EJC.2025.104163.

13 Binh-Minh Bui-Xuan, Jan Arne Telle, and Martin Vatshelle. Fast dynamic programming for locally checkable vertex subset and vertex partitioning problems. *Theor. Comput. Sci.*, 511:66–76, 2013. doi:10.1016/J.TCS.2013.01.009.

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

15 Eduard Eiben, Robert Ganian, Thekla Hamm, Lars Jaffke, and O-joung Kwon. A unifying framework for characterizing and computing width measures. In Mark Braverman, editor, *Proceedings of the 13th Innovations in Theoretical Computer Science Conference, ITCS 2022*, volume 215 of *LIPIcs*, pages 63:1–63:23. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2022. doi:10.4230/LIPIcs.ITCS.2022.63.

16 Esther Galby, Andrea Munaro, and Bernard Ries. Semitotal domination: New hardness results and a polynomial-time algorithm for graphs of bounded mim-width. *Theor. Comput. Sci.*, 814:28–48, 2020. doi:10.1016/J.TCS.2020.01.007.

17 Petr A. Golovach, Pinar Heggernes, Mamadou Moustapha Kanté, Dieter Kratsch, Sigve Hortemo Sæther, and Yngve Villanger. Output-polynomial enumeration on graphs of bounded (local) linear mim-width. *Algorithmica*, 80(2):714–741, 2018. doi:10.1007/S00453-017-0289-1.

18 Carolina Lucía Gonzalez and Felix Mann. On d-stable locally checkable problems parameterized by mim-width. *Discret. Appl. Math.*, 347:1–22, 2024. doi:10.1016/J.DAM.2023.12.015.

19 Svein Høgemo, Jan Arne Telle, and Erlend Raa Vågset. Linear mim-width of trees. In Ignasi Sau and Dimitrios M. Thilikos, editors, *Proceedings of the 45th International Workshop on Graph-Theoretic Concepts in Computer Science, WG 2019*, volume 11789 of *Lecture Notes in Computer Science*, pages 218–231. Springer, 2019. doi:10.1007/978-3-030-30786-8_17.

20 Lars Jaffke, O-joung Kwon, Torstein J. F. Strømme, and Jan Arne Telle. Mim-width III. Graph powers and generalized distance domination problems. *Theor. Comput. Sci.*, 796:216–236, 2019. doi:10.1016/J.TCS.2019.09.012.

21 Lars Jaffke, O-joung Kwon, and Jan Arne Telle. Mim-Width I. Induced path problems. *Discret. Appl. Math.*, 278:153–168, 2020. doi:10.1016/J.DAM.2019.06.026.

22 Lars Jaffke, O-joung Kwon, and Jan Arne Telle. Mim-width II. The feedback vertex set problem. *Algorithmica*, 82(1):118–145, 2020. doi:10.1007/S00453-019-00607-3.

23 Lars Jaffke, O-joung Kwon, and Jan Arne Telle. Classes of intersection digraphs with good algorithmic properties. *J. Graph Theory*, 106(1):110–148, 2024. doi:10.1002/JGT.23065.

24 Stefan Mengel. Lower bounds on the mim-width of some graph classes. *Discret. Appl. Math.*, 248:28–32, 2018. doi:10.1016/J.DAM.2017.04.043.

25 Andrea Munaro and Shizhou Yang. On algorithmic applications of sim-width and mim-width of (H_1, H_2)-free graphs. *Theor. Comput. Sci.*, 955:113825, 2023. doi:10.1016/J.TCS.2023.113825.

26 Yota Otachi, Akira Suzuki, and Yuma Tamura. Finding induced subgraphs from graphs with small mim-width. In Hans L. Bodlaender, editor, *Proceedings of the 19th Scandinavian Symposium and Workshops on Algorithm Theory (SWAT 2024)*, volume 294 of *LIPIcs*, pages 38:1–38:16. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2024. doi:10.4230/LIPICS.SWAT.2024.38.

27 B. S. Panda and Dinabandhu Pradhan. NP-Completeness of Hamiltonian cycle problem on rooted directed path graphs. *CoRR*, abs/0809.2443, 2008. arXiv:0809.2443.

28 Martin Vatshelle. *New Width Parameters of Graphs.* PhD thesis, University of Bergen, Norway, 2012.