



UNIVERSITY OF LEEDS

This is a repository copy of *Obstructions for linear rank-width at most 1*.

White Rose Research Online URL for this paper:
<http://eprints.whiterose.ac.uk/101103/>

Version: Accepted Version

Article:

Adler, I, Farley, AM and Proskurowski, A (2014) Obstructions for linear rank-width at most 1. *Discrete Applied Mathematics*, 168. pp. 3-13. ISSN 0166-218X

<https://doi.org/10.1016/j.dam.2013.05.001>

© 2013, Elsevier. Licensed under the Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International
<http://creativecommons.org/licenses/by-nc-nd/4.0/>

Reuse

Unless indicated otherwise, fulltext items are protected by copyright with all rights reserved. The copyright exception in section 29 of the Copyright, Designs and Patents Act 1988 allows the making of a single copy solely for the purpose of non-commercial research or private study within the limits of fair dealing. The publisher or other rights-holder may allow further reproduction and re-use of this version - refer to the White Rose Research Online record for this item. Where records identify the publisher as the copyright holder, users can verify any specific terms of use on the publisher's website.

Takedown

If you consider content in White Rose Research Online to be in breach of UK law, please notify us by emailing eprints@whiterose.ac.uk including the URL of the record and the reason for the withdrawal request.



eprints@whiterose.ac.uk
<https://eprints.whiterose.ac.uk/>

Obstructions for linear rank-width at most 1

Isolde Adler^{1*}, Arthur Farley², and Andrzej Proskurowski²

¹ Institut für Informatik, Goethe-Universität, 60325 Frankfurt, Germany
`iadler@informatik.uni-frankfurt.de`

² University of Oregon, Eugene, OR 97403-1202, USA
`andrzej,art@cs.uoregon.edu`

Abstract We establish the set of minimal forbidden induced subgraphs for the class of graphs having linear rank-width at most 1. From these we derive both the vertex-minor and pivot-minor obstructions for the class.

1 Introduction

Rank-width is a graph parameter introduced by Oum and Seymour [16] as an efficient approximation of the clique-width [6] of a graph. Linear rank-width is obtained from rank-width by restricting the trees in the decomposition to caterpillars. Intuitively, linear rank-width is related to rank-width in the same way as path-width is related to tree-width. Large classes of graph problems that are NP-hard in general become tractable when restricted to graphs of bounded (linear) rank-width [5,8,10,13]. While the structure of graphs of bounded path-width is well understood [3,7,12,17], much less is known about graphs of bounded linear rank-width. Indeed, linear rank-width generalizes path-width in the sense that if a graph G has path-width at most k then G has linear rank-width at most k (this is not hard to prove, cf. [1]). Recently, Ganian [9] introduced the class of *thread graphs* and proved that the graphs of linear rank-width at most 1 are precisely the thread graphs.

In this paper, we characterize the graphs of linear rank-width at most 1 in three ways: firstly by induced subgraph obstructions, secondly by vertex-minor obstructions and thirdly by pivot-minor obstructions. More precisely, let \leq be a quasi-ordering (i.e. a reflexive and transitive ordering) on graphs, and let \mathcal{C} be a class of graphs that is downward-closed under \leq . A graph G is an *obstruction* for \mathcal{C} if $G \notin \mathcal{C}$ and $H \in \mathcal{C}$ for every $H \preceq G$. For example, Kuratowski's famous theorem states that the obstructions for planar graphs w.r.t. (topological) minor containment are the two graphs K_5 and $K_{3,3}$.

For the class of graphs of linear rank-width at most 1, we determine the set of obstructions w.r.t. induced subgraph containment (cf. Theorem 4 and Figure 3), w.r.t. vertex-minor containment (cf. Theorem 25 and Figure 5), and w.r.t. pivot-minor containment (cf. Theorem 26 and Figure 9). It is known that the obstruction sets for graphs of bounded linear rank-width w.r.t. vertex-minor and pivot-minor containment are finite [14,15]. However,

* Partially supported by the German Research Council, Project GalA, AD 411/1-1.

until now none of these obstruction sets were known explicitly. Obstruction set characterizations are known for the class of graphs of rank-width at most 1 (i.e. for the distance hereditary graphs) [2] and for the class of circle graphs [4]. To obtain our results, we depend upon the known equivalence between graphs of linear rank-width at most 1 and thread graphs [9] and the fact that we can reduce the search for unknown obstructions to distance-hereditary graphs.

In Section 2, we introduce basic notions and known facts. In Sections 3 and 4 we find the induced subgraph obstructions, from which we obtain the vertex-minor obstructions in Section 5 and the pivot-minor obstructions in Section 6.

2 Definitions

We consider finite, simple graphs, each graph G consisting of the set $V(G)$ of vertices and the set $E(G)$ of edges, every edge $e \in E(G)$ being a two-element subset of $V(G)$. Given an edge $e = \{u, v\}$, we say that u is adjacent to v and e is *incident* with both u and v . A sequence v_0, v_1, \dots, v_k of vertices of G , no two being equal and every consecutive pair being adjacent is called a *path*. The *length* of a path is k , the number of its edges. If additionally v_0 and v_k are adjacent, they form a *cycle*. A graph without a cycle is called *acyclic*. For a vertex v , we let $N_G(v) := \{u \in V(G) \mid \{u, v\} \in E(G)\}$ be the *neighborhood* of v in G . The *degree* of $v \in V(G)$ is $\deg_G(v) := |N_G(v)|$. A vertex of degree 1 is a *pendant vertex*, and the edge incident with a pendant vertex is a *pendant edge*. A graph H is a *subgraph* of G if $V(H) \subseteq V(G)$ and $E(H) \subseteq E(G)$. For a subset $X \subseteq V(G)$, let $G[X]$ be the subgraph of G induced by X , i.e. $V(G[X]) = X$ and $E(G[X]) := \{e \in E(G) \mid e \subseteq X\}$. Such a graph H is an *induced subgraph* of G . For a subset $Y \subseteq V(G)$, we let $G \setminus Y := G[V(G) \setminus Y]$. If $Y = \{y\}$ is a singleton set, then we write $G \setminus y$ instead of $G \setminus \{y\}$.

A graph G is *connected* if $G \neq \emptyset$ and any two vertices of G are connected by a path. A graph G that is not connected is said to be *disconnected*. A *connected component* of G is a maximal connected subgraph of G . We say that a vertex $v \in V(G)$ is a *cut-vertex* if $G \setminus v$ has more connected components than G . A *tree* T is an acyclic connected graph. We denote the set of leaves of T by $L(T)$. Any vertex in $V(T) \setminus L(T)$ is an *internal vertex*. A tree is *cubic* if it has at least two vertices and every internal vertex has degree 3. For an integer $n \geq 3$, we let C_n denote the *n-cycle*, i.e. the cycle with n vertices. For a graph G containing a cycle, a *chord* in the cycle is an edge of G between two vertices of the cycle that are not consecutive. A *complete bipartite* graph is a graph G with a bipartition $[X, Y]$ of $V(G)$ such that $E(G) = \{\{x, y\} \mid x \in X \text{ and } y \in Y\}$.

The *distance* between two vertices u and v of G is the minimum length of a path in G connecting u and v (or infinity if no such path exists). A *distance-hereditary graph* is a graph G for which the distances between vertices in any connected, induced subgraph of G are the same as in G [11]. Every distance-hereditary graph can be constructed from a single vertex by a sequence of the following three *dh-operations* [2]:

\oplus_p : Add a new pendant vertex v connected by an edge to an existing vertex u of the graph.

\oplus_f : Add a new vertex v with the same set of neighbors as a vertex u of the graph. The vertices u and v are called *false twins* of each other.

\oplus_t : Add a new vertex v with the same set of neighbors as a vertex u of the graph, connecting u and v by an edge. The vertices u and v are called *true twins* of each other.

We denote a dh-operation acting on a vertex u of a graph G by $\oplus_x(G, u)$, where $x \in \{p, t, f\}$.

Let $M(G)$ denote the adjacency matrix of a graph G , *i.e.*, $M(G)$ is the $|V(G)| \times |V(G)|$ matrix where the columns and the rows are indexed by the vertices of G , and $M(G)$ has entries in $\{0, 1\}$, where the entry $m_{i,j} = 1$ if and only if the corresponding row vertex is incident to the corresponding column vertex. For a bipartition $[X, Y]$ of $V(G)$, we let $M(G)[X, Y]$ denote the $|X| \times |Y|$ submatrix $(m_{i,j})_{i \in X, j \in Y}$ of $M(G)$. The *cutrank function* of a graph G is defined by $\text{cutrk}_G: 2^{V(G)} \rightarrow \mathbb{N}$ given by $\text{cutrk}_G(X) := \text{rank}(M(G)[X, V(G) \setminus X])$, where rank is the rank function over $\text{GF}(2)$. A *rank-decomposition* of a graph G is a pair (T, λ) , where T is a cubic tree and $\lambda: L(T) \rightarrow V(G)$ is a bijection. For every edge $e \in E(T)$, the two connected components of $T \setminus e$ induce a partition (X_e, Y_e) of $L(T)$. The *width* of e is defined as $\text{cutrk}_G(\lambda(X_e))$. The *width* of a rank-decomposition (T, λ) is the maximum width over all edges of T . The *rank-width* of G is defined as

$$\text{rw}(G) := \min\{\text{width of } (T, \lambda) \mid (T, \lambda) \text{ rank-decomposition of } G\}.$$

If $|V(G)| \leq 1$, then G has no rank-decomposition and we let $\text{rw}(G) = 0$.

Proposition 1 ([14]) *A graph G is distance-hereditary if and only if $\text{rw}(G) \leq 1$.*

Linear rank-width is a related concept that restricts the form of the rank-decomposition. A *caterpillar* is a tree T in which the removal of all leaves results in a path. A *linear rank-decomposition* of a graph G is a rank-decomposition (T, λ) of G , where T is a caterpillar. The *linear rank-width* of G is defined as

$$\text{lrw}(G) := \min\{\text{width of } (T, \lambda) \mid (T, \lambda) \text{ linear rank-decomposition of } G\}.$$

Equivalently, $\text{lrw}(G) \leq k$ if there exists a permutation π of $V(G)$ such that for every prefix X of π $\text{cutrk}_G(X) \leq k$. If $|V(G)| \leq 1$, then G has no linear rank-decomposition and we let $\text{lrw}(G) = 0$.

Obviously, $\text{lrw}(G) \geq \text{rw}(G)$, and $\text{lrw}(G) \leq k$ implies $\text{rw}(G) \leq k$. It is easy to verify that cliques, caterpillars and complete bipartite graphs have linear rank-width at most 1 and that the disjoint union of two graphs G and H has linear rank-width equal to $\max\{\text{lrw}(G), \text{lrw}(H)\}$.

Example 2 *The cycle C_5 satisfies $\text{lrw}(C_5) = 2$. In any linear rank-decomposition (T, λ) of C_5 every edge in $E(T)$ between two internal vertices of T has width 2, and every pendant edge in $E(T)$ has width 1.*

3 Induced subgraph obstructions

The class of graphs with linear rank-width at most k is closed under induced subgraphs, *i.e.*, if a graph G satisfies $\text{lrw}(G) \leq k$ then $\text{lrw}(H) \leq k$ for all induced subgraphs H of G . We define the set of *induced subgraph obstructions* (or *minimal forbidden induced subgraphs*) for the class of graphs G such that $\text{lrw}(G) \leq 1$, denoted as \mathcal{O} , as all graphs G such that $\text{lrw}(G) > 1$ and $\text{lrw}(G \setminus v) = 1$ for all vertices v in G . Such obstructions are necessarily connected.

Lemma 3 *Every graph $G \in \mathcal{O}$ is connected.*

Proof. Assume a graph $G \in \mathcal{O}$ is not connected. Then we can delete a vertex v in one of the connected components C of G and find that the resulting graph has linear rank-width at most 1. Each of the unaffected components must have linear rank-width ≤ 1 . Similarly, by deleting a vertex v' in a different component $C' \neq C$ of G , we find that $\text{lrw}(C) \leq 1$. Since G is the disjoint union of its components, it follows that $\text{lrw}(G) = 1$, a contradiction. \square

We will prove the following theorem.

Theorem 4 *The set of induced subgraph obstructions for graphs having linear rank-width at most 1 consists of the induced subgraph obstructions for distance-hereditary graphs and the graphs shown in Figure 3.*

It follows from Proposition 1 that the class of graphs having linear rank-width at most 1 is a subclass of distance-hereditary graphs. Hence, to prove Theorem 4 it suffices to show that \mathcal{O} consists of the induced subgraph obstructions for distance-hereditary graphs and the obstructions that are themselves distance-hereditary graphs, which are the graphs shown in Figure 3.

The set of induced subgraph obstructions for distance-hereditary graphs, denoted as \mathcal{D} , has been shown to consist of exactly the following elements: [2]:

- any cycle of length five or greater (“holes” of length at least 5),
- a 5-cycle with one chord (“house”),
- a 5-cycle with two non-crossing chords (“gem”), and
- a 6-cycle with a chord connecting two vertices at distance 3 along the cycle (“domino”).

Lemma 5 *\mathcal{D} is a subset of \mathcal{O} .*

Proof. By inspection, for every graph G in \mathcal{D} , $\text{lrw}(G) = 2$ and $\text{lrw}(G \setminus v) = 1$, for all vertices v in G . Therefore, these graphs are also induced subgraph obstructions for graphs having linear rank-width at most 1. \square

To complete the specification of the set \mathcal{O} it is necessary to determine the set of distance-hereditary graphs that are induced subgraph obstructions for graphs with linear rank-width at most 1. We denote this set of obstructions as \mathcal{O}_{dh} .

Lemma 6 $\mathcal{O} = \mathcal{D} \cup \mathcal{O}_{dh}$.

Proof. By Lemma 5, it suffices to show that $\mathcal{O} \setminus \mathcal{D} \subseteq \mathcal{O}_{dh}$. Let $G \in \mathcal{O} \setminus \mathcal{D}$. Then $\text{lrw}(G) \geq 1$, $\text{rw}(G) \leq 1$ by Proposition 1, and, for all $v \in V(G)$, $1 = \text{lrw}(G \setminus v) = \text{rw}(G \setminus v)$. Hence $G \in \mathcal{O}_{dh}$ by definition. \square

To determine the set \mathcal{O}_{dh} , we consider an alternative characterization of graphs G with $\text{lrw}(G) \leq 1$ introduced in [9] called *thread graphs*. Here, we define thread graphs and note some of their relevant properties. It can be easily seen that our definition is equivalent to the original definition in [9].

A *thread block* is the basic building block of a thread graph. A *thread block* B is a tuple $(G, (a, b), \bar{v}, \mathcal{L})$, consisting of a graph G , a distinguished edge $\{a, b\} \in E(G)$, called the *thread edge* of G , an ordering $\bar{v} = (v_1, \dots, v_n)$ of the vertices of $V(G) \setminus \{a, b\}$, called a *thread ordering*, and an associated *thread labeling* $\mathcal{L}: V(G) \setminus \{a, b\} \rightarrow \{L, R, *\}$, such that

- for all $1 \leq i < j \leq n$, $\{v_i, v_j\} \in E(G)$ if and only if $\mathcal{L}(v_i) \in \{R, *\}$ and $\mathcal{L}(v_j) \in \{L, *\}$.
- for all $1 \leq j \leq n$, $\{a, v_j\} \in E(G)$ if and only if $\mathcal{L}(v_j) \in \{L, *\}$.
- for all $1 \leq j \leq n$, $\{v_j, b\} \in E(G)$ if and only if $\mathcal{L}(v_j) \in \{R, *\}$.

Intuitively, in the sequence (a, v_1, \dots, v_n, b) every vertex u with $\mathcal{L}(u) \in \{L, *\}$ ‘sees’ all vertices v to its left that ‘look’ to the right, *i.e.*, that have $\mathcal{L}(v) \in \{R, *\}$. Symmetrically, every vertex v with $\mathcal{L}(v) \in \{R, *\}$ ‘sees’ all vertices u to its right that ‘look’ to the left, *i.e.*, that have $\mathcal{L}(u) \in \{L, *\}$. The label “*” can be understood to represent both labels, *i.e.*, to be the set $\{L, R\}$. Vertices a and b are called *thread vertices* and are implicitly labeled “*”.

A thread block is *trivial* if it consists of only the thread edge. A non-trivial thread block may have a cut-vertex, as it may include pendant vertices. Figure 1 shows graph G with an edge $\{a, b\}$, an ordering $\bar{v} = (a', v, u, b')$ and a labeling \mathcal{L} such that $(G, (a, b), \bar{v}, \mathcal{L})$ is a thread block.

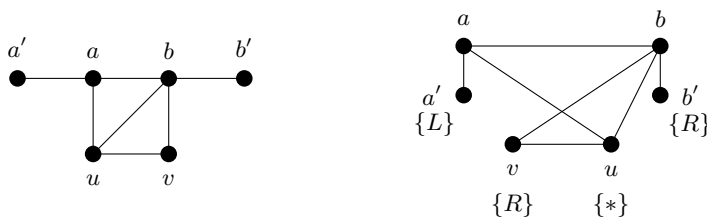


Figure 1. A graph G and corresponding thread block $B = (G, (a, b), (a', v, u, b'), \mathcal{L})$.

A *connected thread graph* G is either a single vertex or a sequence of thread blocks B_i , $1 \leq i \leq n$, such that thread vertex b_i of thread block B_i is identical with thread vertex a_{i+1} of thread block B_{i+1} , for $1 \leq i \leq n - 1$. We call this shared vertex a *bridging vertex*. A *consistent threading* of a connected thread graph G is an assignment of a sequence of edges that form a path to be thread edges, and an ordering and labeling of non-thread vertices, such that the edges implied by the ordering and labeling correspond to the edges of G . From now on we only consider consistent threadings of thread graphs where the first and the last thread block are non-trivial. This is not a restriction, because a trivial thread block is a pendant edge and may be folded into the neighboring thread block.

A *thread graph* G is either the empty graph or a disjoint union of connected thread graphs. A *consistent threading* of a thread graph is a collection of consistent threadings of its connected components.

The following property of thread graphs follows immediately from the definitions.

Remark 7 *Let G be a thread graph. The set of bridging vertices is the same for every consistent threading of G , and every bridging vertex is shared by exactly two thread blocks.* \square

The structure of a thread block can be represented concisely by the sequence (l_1, \dots, l_n) of vertex labels of the non-thread vertices (e.g., $(L, R, *, R)$ for the graph in Figure 1). A sequence of thread blocks can likewise be represented as a sequence of non-thread vertex label sequences of the thread blocks (e.g., $(L, R)(*)$ for a two block sequence). Minimal thread blocks involve single non-thread vertices.

Remark 8 *Every connected thread graph of three or more vertices has one of the following thread blocks as an induced subgraph: (R) , (L) , or $(*)$.* \square

From these graphs the set of non-isomorphic, minimal two-block thread graphs can be determined and are useful for determining certain obstructions.

Lemma 9 *The vertex-minimal thread graphs with two non-trivial thread blocks are $(L)(R)$, $(L)(*)$, and $(*)(*)$, which are unique up to an isomorphism.*

Proof. The thread graphs $(R)(R)$, $(R)(*)$, $(R)(L)$ can be represented as the one-block thread graphs (LLR) , $(LL*)$, and (LLL) , respectively. $(L)(L)$ and $(*)(L)$ are isomorphic to the first two of these graphs, respectively. Also, $(*)(R)$ is isomorphic to $(L)(*)$, so we have accounted for all 9 vertex-minimal non-empty thread graphs with two thread blocks. \square

The three unique vertex-minimal two-thread block thread graphs are presented in Figure 2.

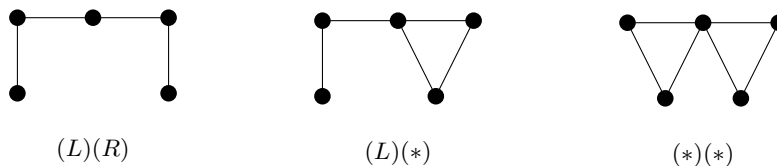


Figure 2. The three graphs of Lemma 9.

The following theorem showing equivalence between the class of thread graphs and graphs G with $\text{lrw}(G) \leq 1$ was proven in [9]. We give a related proof here for completeness.

Theorem 10 (Ganian [9]) *A graph G has $\text{lrw}(G) \leq 1$ if and only if G is a thread graph.*

Proof. We may assume that G is connected and $E(G) \neq \emptyset$.

(\Rightarrow) Assume $\text{lrw}(G) \leq 1$, and let (T, λ) be a linear rank-decomposition witnessing this. Consider a total ordering \prec of the vertices of G that is consistent with the linear structure of T yielding the linear rank-width ≤ 1 . The first vertex in the ordering is a non-bridging thread vertex. We will provide labeling of vertices which yields a consistent threading. Consider further a vertex v being *processed*. There is a unique binary string expressing adjacencies between already processed vertices $u \prec v$ and the vertices w , where $v \preceq w$.

We use $\mathbf{e} = 0^*$ to represent the pattern of all 0's, i.e., no adjacencies (“empty neighborhood”). We use $\mathbf{n} = \mathbf{e}1\{0, 1\}^*$ to mean an arbitrary pattern of 0's and 1's, including at least one 1.

Case 1: The neighborhood of processed vertices is $1\mathbf{n}$. Since v is the first unprocessed vertex it is adjacent to the processed vertices. After v is processed, it could either have no adjacencies to the remaining unprocessed vertices, in which case we label it “ L ” in the corresponding consistent threading, or the neighborhood could be the same as the neighborhood of other processed vertices, in which case it is labeled “ $*$ ” in the corresponding consistent threading.

Case 2: The neighborhood of processed vertices is $1\mathbf{e}$. This identifies v as a thread vertex. After processing, v has either an empty neighborhood, in which case we label it “ L ”, or its adjacencies with unprocessed vertices are expressed by \mathbf{n} , in which case v is labeled “ $*$ ” and is a bridging thread vertex.

Case 3: The neighborhood is $0\mathbf{n}$. After v is processed, it must have a neighborhood \mathbf{n} as do other processed vertices, in which case it is labeled “ R ” in the consistent threading.

Thus, in each case there is a labeling for v , proving that G is a thread graph.

(\Leftarrow) For the converse, assume that G is a thread graph with a given consistent threading that yields an ordering \prec of $V(G)$. We define a linear rank-decomposition (T, λ) by mapping the leaves of T to the vertices of G in such a way that the linear structure of (T, λ) respects \prec . It is straightforward to verify that the width of (T, λ) is ≤ 1 . \square

4 Distance-Hereditary Obstructions \mathcal{O}_{dh}

Let \mathcal{H} be the set of fourteen graphs shown in Figure 3. In this section, we establish the following theorem:

Theorem 11 *The set \mathcal{O}_{dh} is equal to the set \mathcal{H} .*

It is not difficult to verify that each of the graphs in Figure 3 is a member of \mathcal{O}_{dh} .

Lemma 12 *\mathcal{H} is a subset of \mathcal{O}_{dh} .*

Proof. By observation, each graph G in \mathcal{H} has $\text{lrw}(G) > 1$. One can not start at any vertex and visit the other vertices sequentially while keeping width 1. However, after removing any vertex from each graph G , a consistent threading of the remaining graph as a thread graph can be found. \square

In the remainder of this section we complete the proof of Theorem 11, showing that \mathcal{H} is the complete set of graphs in \mathcal{O}_{dh} .

A graph in \mathcal{O}_{dh} is a connected distance-hereditary graph by Lemma 3. Any connected distance-hereditary graph G of more than one vertex can be described as $G = \oplus_x(G', u)$, where \oplus_x is one of the three dh-operations applied to vertex u of a distance-hereditary graph G' .

Lemma 13 *If a distance-hereditary graph $G = \oplus_x(G', u)$ is in \mathcal{O}_{dh} then G' is a connected thread graph.*

Proof. Since G is a minimal forbidden induced subgraph, the graph G' in any construction must have linear rank-width less than or equal to 1. Graph G' must be connected, as the distance hereditary operations maintain connectivity. \square

Throughout this section, let graph G be $\oplus_x(G', u)$, where G' is a distance-hereditary graph, as described above. To discover all distance-hereditary obstructions to linear rank-width 1, we consider each of the three dh-operations applied to all possible cases for vertex u of a thread graph G' . We show that after applying the operation, we have either a thread graph G or a graph G that is one of the 14 graphs of Figure 3 (in set \mathcal{H}) as an induced subgraph. First, we consider applying a dh-operation to a thread vertex u of G' .

Lemma 14 *Applying any one of the three dh-operations to a non-bridging thread vertex $u \in V(G')$ results in a thread graph.*

Proof. Assume wlog. that the non-bridging thread vertex u is at the right end of a consistent threading of G' . A new pendant vertex v can be labeled R as last vertex in the thread block sequence of the thread block including u . A new false (true) twin v of u can be labeled “ L ” (respectively “ $*$ ”) as last vertex in the thread block sequence. \square

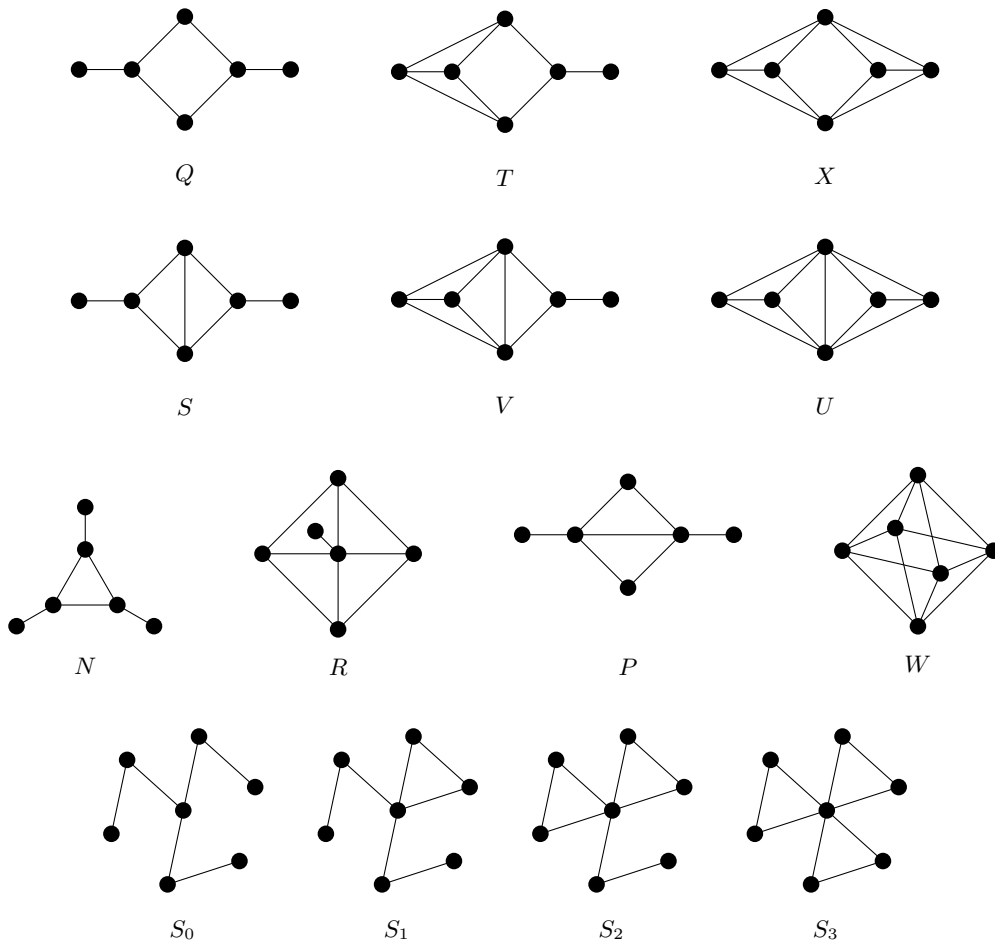


Figure 3. The distance-hereditary induced subgraph obstructions for the class of graphs of linear rank-width at most 1. Together with the induced subgraph obstructions for distance-hereditary graphs they form the set of all induced subgraph obstructions for the class of graphs of linear rank-width at most 1 (cf. Theorem 11).

Lemma 15 Applying $\oplus_p(G', u)$ when u is a bridging thread vertex of G' results in a thread graph.

Proof. The new vertex v can be considered to be a non-thread vertex labeled “ L ” (“ R ”) as the first (last) vertex in the thread block sequence in one of the two consecutive thread blocks of which vertex u is a member. \square

Lemma 16 Applying $\oplus_f(G', u)$ or $\oplus_t(G', u)$ when u is a bridging thread vertex of G' results in a non-thread graph G containing Q, S, T, U, V , or X (shown in Figure 3) as an induced subgraph.

Proof. Choose a consistent threading of G' . By Remark 7, u is a bridging vertex of G' with respect to this threading. Hence u has two neighbors, u_1 and u_2 , that are thread vertices. If both u_1 and u_2 are bridging vertices as well, then $\oplus_f(G', u)$ contains the half-cube Q as an induced subgraph, and $\oplus_t(G', u)$ contains S as an induced subgraph.

If a neighbor, say u_1 is not bridging, but u_2 is bridging, then the thread block involving u_1 is non-trivial, so either u_1 has a pendant vertex or u_1 and u share a common neighbor. If u_1 has a pendant vertex, then $\oplus_f(G', u)$ contains Q as an induced subgraph, and $\oplus_t(G', u)$ contains S as an induced subgraph. If u_1 and u share a common neighbor, then $\oplus_f(G', u)$ contains T as an induced subgraph and $\oplus_t(G', u)$ contains V as an induced subgraph.

If both u_1 and u_2 are non-bridging, then G' has exactly two thread blocks and both thread blocks are non-trivial. If both u_1 and u_2 have pendant vertices, then $\oplus_f(G', u)$ is isomorphic to Q and $\oplus_t(G', u)$ is isomorphic to S . If u_1 has no pendant vertex, but u_2 has a pendant vertex, then we find that $\oplus_f(G', u)$ is isomorphic to T and $\oplus_t(G', u)$ is isomorphic to V . Finally, if neither u_1 nor u_2 has a pendant vertex, then $\oplus_f(G', u)$ is isomorphic to X and $\oplus_t(G', u)$ is isomorphic to U . \square

What remains is to consider the cases for dh-operation \oplus_x being applied to a non-thread vertex u of G' . Assume a given consistent threading of the thread block containing u in the thread graph G' .

We consider the following cases for non-thread vertex u , depending on the structure of the thread block containing u .

1. $\mathcal{L}(u) = “*”$

1.1 u has neither a bridging thread vertex nor a non-thread vertex labeled “ L ” or “ R ” before it (the case for “after” is symmetric)

1.2 u has a bridging thread vertex or a non-thread vertex labeled “ L ” or “ R ” both before and after it

2. $\mathcal{L}(u) = “R”$ (the case for $\mathcal{L}(u) = “L”$ is symmetric)

2.1 u has no vertex labeled “ L ” or “*” after it (i.e., u is a pendant vertex)

2.1.1 u is adjacent to a non-bridging thread vertex

2.1.2 u is adjacent to a bridging thread vertex

2.2 u has a vertex labeled “*” or “ L ” after it (i.e., u is in a 4-cycle)

2.2.1 u has neither a bridging thread vertex nor a non-thread vertex labeled “*” or “ L ” before it

2.2.2 u has a bridging thread vertex or a non-thread vertex labeled “*” or “ L ” before it

We must consider the three dh-operations applied in each of these structural cases. We refer to the above numbering of cases in the lemmas to follow.

Lemma 17 (Case 1.) *Applying $\oplus_t(G', u)$ when u is a non-thread vertex labeled “*” in G' results in a thread graph G .*

Proof. The new vertex v can have the same label and be consecutive with u in a consistent threading of G . \square

Lemma 18 (Case 1.1) *Applying $\oplus_p(G', u)$ or $\oplus_f(G', u)$ when u is a non-thread vertex labeled “*” that has neither a bridging thread vertex nor a non-thread vertex labeled “ L ” or “ R ” before it results in a thread graph G .*

Proof. The vertex u can become the thread vertex at the left end of the thread block. A consistent threading of G will include the new vertex v as the first vertex of the thread block sequence, labeled “ L ” in the pendant vertex case or “ R ” in the false twin case. \square

Lemma 19 (Case 1.2) *Applying $\oplus_p(G', u)$ or $\oplus_f(G', u)$ when u is a non-thread vertex labeled “*” that has a bridging thread vertex or non-thread vertex labeled “ L ” or “ R ” before and after it results in a non-thread graph G that has graph N , R , P , or W shown in Figure 3 as an induced subgraph.*

Proof. Note that the case that u has a bridging vertex v before it is subsumed by the case that it has a pendant vertex labeled “ L ” before it. Vertex v is part of a thread edge of the adjacent thread block, connecting it to another thread vertex w . Deleting all vertices other than v and w of that adjacent thread block yields the case where u has a vertex w labeled “ L ” before it. The case where u has a bridging vertex after it is similarly subsumed by the case that it has a vertex labeled “ R ” after it.

First, consider the cases of applying operation $\oplus_p(G', u)$. If u has a vertex labeled “ L ” before it and a vertex labeled “ L ” after it, applying $\oplus_p(G', u)$ results in a graph containing induced subgraph P . The result is the same if u has a vertex labeled “ R ” before it and a vertex labeled “ R ” after it. If u has a vertex labeled “ L ” before it and a vertex labeled “ R ” after it, applying $\oplus_p(G', u)$ results in a graph G containing induced subgraph N . Finally, if u has a vertex labeled “ R ” before it and a vertex labeled “ L ” after it, applying $\oplus_p(G', u)$ results in a graph G containing induced subgraph R .

Now, consider the cases of applying operation $\oplus_f(G', u)$. If u has a vertex labeled “ L ” before it and a vertex labeled “ L ” after it, applying $\oplus_f(G', u)$ results in a graph containing induced subgraph R . The result is the same if u has a vertex labeled “ R ” before it and a

vertex labeled “ R ” after it. If u has a vertex labeled “ L ” before it and a vertex labeled “ R ” after it, applying $\oplus_f(G', u)$ results in a graph G containing induced subgraph P . Finally, if u has a vertex labeled “ R ” before it and a vertex labeled “ L ” after it, applying $\oplus_f(G', u)$ results in a graph G containing induced subgraph W . \square

Now, consider Case 2, a non-thread vertex u labeled “ R ” in G' .

Lemma 20 (Case 2.) *Applying $\oplus_f(G', u)$ when u is a non-thread vertex labeled “ R ” in G' results in a thread graph G .*

Proof. The vertex v can have the same label and be consecutive with u in a consistent threading of G . \square

Consider Case 2.1 where u is a pendant vertex of G' . In these cases, u is adjacent to a thread vertex w of G' .

Lemma 21 (Case 2.1.1) *Applying $\oplus_p(G', u)$ or $\oplus_t(G', u)$ when u is a pendant vertex in G' labeled “ R ” adjacent to a non-bridging thread vertex w results in a thread graph G .*

Proof. The vertex u will become a thread vertex (with vertex w becoming a bridging thread vertex) in a consistent threading of thread graph G . \square

Lemma 22 (Case 2.1.2) *Applying $\oplus_p(G', u)$ or $\oplus_t(G', u)$ when u is a pendant vertex in G' labeled “ R ” adjacent to a bridging thread vertex w results in a graph G having S_0 , S_1 , S_2 , or S_3 of Figure 3 as an induced subgraph.*

Proof. The vertex w would become a bridging vertex in $G = \oplus_x(G', u)$ shared by three non-trivial thread blocks, contradicting the property of a thread graph of Remark 7. The minimal, non-trivial thread graphs with two thread blocks are given by Lemma 9; these minimal graphs determine the obstruction set generated by the operation. \square

Finally, consider Case 2.2, when u is labeled “ R ” and is in a 4-cycle, having a vertex labeled “ L ” or “ $*$ ” after it in the thread block.

Lemma 23 (Case 2.2.1) *Applying $\oplus_p(G', u)$ or $\oplus_t(G', u)$ to a non-thread vertex u labeled “ R ” that is in a 4-cycle and has neither a bridging thread vertex nor a non-thread vertex labeled “ $*$ ” or “ L ” before it results in a thread graph G .*

Proof. In this case, the left thread vertex of the block is an end vertex on the thread. This case is as that discussed in Lemma 14; vertex u can “switch places” with the original left thread vertex of the thread block. \square

Lemma 24 (Case 2.2.2) *Applying $\oplus_p(G', u)$ or $\oplus_t(G', u)$ to non-thread vertex u labeled “ R ” that is in a 4-cycle and has a bridging thread vertex or a non-thread vertex labeled “ $*$ ”*

or “ L ” before it results in a graph G having one of the graphs Q , S , T , V , X , or U of Figure 3 as an induced subgraph.

Proof. Note that the case that u has a bridging vertex v before it is subsumed by the case that it has a pendant vertex labeled “ L ” before it. Vertex v is part of a thread edge of the adjacent thread block, connecting it to another thread vertex w . Deleting all vertices other than v and w of that adjacent thread block yields the case where u has a vertex w labeled “ L ” before it.

First, consider the cases where operation $\oplus_p(G', u)$ is applied. If there is a vertex labeled “ L ” before u , then G contains induced subgraph Q or S depending on whether the vertex following u is labeled “ L ” or “ $*$ ”, respectively. If there is a vertex labeled “ $*$ ” before u , then G contains induced subgraph T or V depending on whether the vertex following u is labeled “ L ” or “ $*$ ”, respectively.

Now, consider the cases where operation $\oplus_t(G', u)$ is applied. If there is a vertex labeled “ L ” before u , then G contains induced subgraph T or V depending on whether the vertex following u is labeled “ L ” or “ $*$ ”, respectively. If there is a vertex labeled “ $*$ ” before u , then G contains induced subgraph X or U depending on whether the vertex following u is labeled “ L ” or “ $*$ ”, respectively. \square

Proof of Theorem 11. By Lemma 12, \mathcal{H} is a subset of \mathcal{O}_{dh} . There can be no further members of \mathcal{O}_{dh} as the above lemmas cover all possible cases of applying a dh-operation to a thread graph. The set of non-thread graphs that results from these cases is exactly equal to set \mathcal{H} . \square

Proof of Theorem 4. The proof follows from Theorem 11 and Lemma 6. \square

5 Vertex-Minor Obstructions

Let G be a graph and let $v \in V(G)$. The graph obtained from G by performing a *local complementation at v* is the graph $G * v$ with $V(G * v) := V(G)$ and $E(G * v) := E(G) \Delta \{\{x, y\} \subseteq N_G(v) \mid x \neq y\}$ (Δ denoting the symmetric difference of two sets). We say that two graphs G and H are *locally equivalent*, if H can be obtained from G by a sequence of local complementations. Note that this is indeed an equivalence relation. Figure 4 shows all graphs that are locally equivalent to C_5 (up to isomorphism).

A graph H is a *vertex-minor* of a graph G , if H can be obtained from G by a sequence of local complementations and vertex deletions. In particular, every induced subgraph of G is a vertex-minor of G . H is a *proper vertex-minor* of G , if $H \preceq_v G$ and $|V(H)| < |V(G)|$. For a given, non-negative integer k , the class of graphs of rank-width at most k is closed under taking of vertex-minors [14].

A graph G is a *vertex-minor obstruction* for the class of graphs of linear rank-width at most 1, if $\text{lrw}(G) \geq 2$ and every proper vertex-minor H of G satisfies $\text{lrw}(H) \leq 1$.

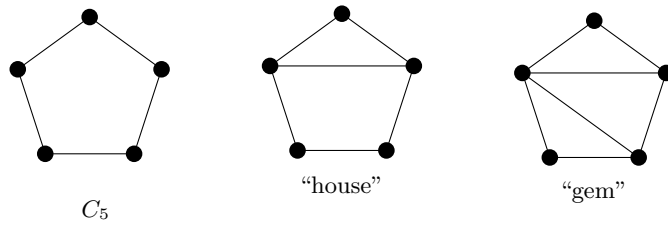


Figure 4. The three graphs that are locally equivalent to C_5 .

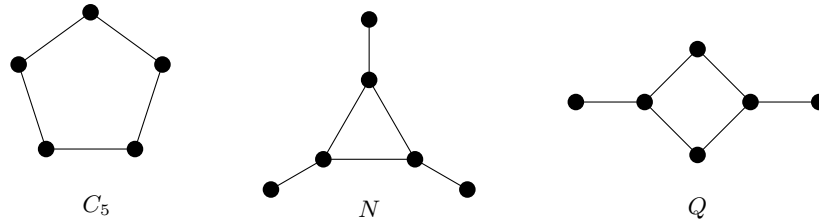


Figure 5. The three vertex-minor obstructions for linear rank-width at most 1: The 5-cycle C_5 , the net graph N , and the half-cube Q .

Theorem 25 *The vertex-minor obstructions for the class of graphs of linear rank-width at most 1 are the three graphs C_5 , N and Q depicted in Figure 5.*

Proof. Since the class of graphs of linear rank-width at most 1 is closed under taking vertex-minors, it suffices to show that the graphs in Figure 5 are a set of pairwise not locally equivalent obstructions, such that every graph with linear rank-width at least 2 contains one of them as a vertex-minor. Indeed, it is not hard to see that the graphs in Figure 5 are pairwise not locally equivalent.

Every graph G with linear rank-width ≥ 2 contains one of the graphs listed in Theorem 4 as an induced subgraph.

If G contains a hole, the house, the gem or the domino graph, then it contains C_5 as a vertex-minor: Any cycle C_k , with $k > 5$, can be reduced in size by repeatedly performing a local complementation and deleting the degree 2 vertex of the resultant 3-cycle. Applying local complementation to a vertex of C_5 yields the house graph. Applying local complementation to one of mutually adjacent vertices of degree 2 of the house graph yields the gem graph. These locally equivalent graphs are shown in Figure 4. Applying local complementation to any vertex v of the domino graph (see Figure 8) and then deleting v , yields a graph locally equivalent to the C_5 . Thus, the only non-distance-hereditary vertex-minor obstruction is the 5-cycle.

To determine the set of distance-hereditary vertex-minor obstructions, first consider the induced subgraph obstructions S_0, S_1, S_2, S_3 that are created by Lemma 22 (shown in Figure 3). All four of these graphs have the net graph N as a vertex-minor. This is realized by first applying local complementation to a vertex of degree 2 in any triangle, reducing all four graphs to S_0 . By applying local complementation to the central vertex v of degree 3 in S_0 and then deleting v , one realizes the net graph N .

Figure 6 and Figure 7 show the remaining ten induced subgraph obstructions postulated by Theorem 11 grouped by local equivalence with graphs N and Q . By applying local complementation to the bottom vertex of graph S , one reaches N . By applying local complementation to the vertices in triangles on the sides of T and U , one reaches S . By applying local complementation to the degree 5 vertex of graph R , one reaches graph U . For the graphs locally equivalent to Q , applying local complementation to a degree 2 vertex of P yields Q . Applying local complementation to the vertex with a pendant neighbor in graph V yields graph X . By applying local complementation to one vertex from each side of X in succession, one reaches Q . Finally, applying local complementation to the top vertex of graph W yields graph X .

Therefore, by deleting zero or more vertices from any graph G with $\text{lrw}(G) \geq 2$, one of the induced subgraph obstructions of Theorem 4 will be encountered. This subgraph can be reduced to one of the three vertex-minor obstructions, as shown above. \square

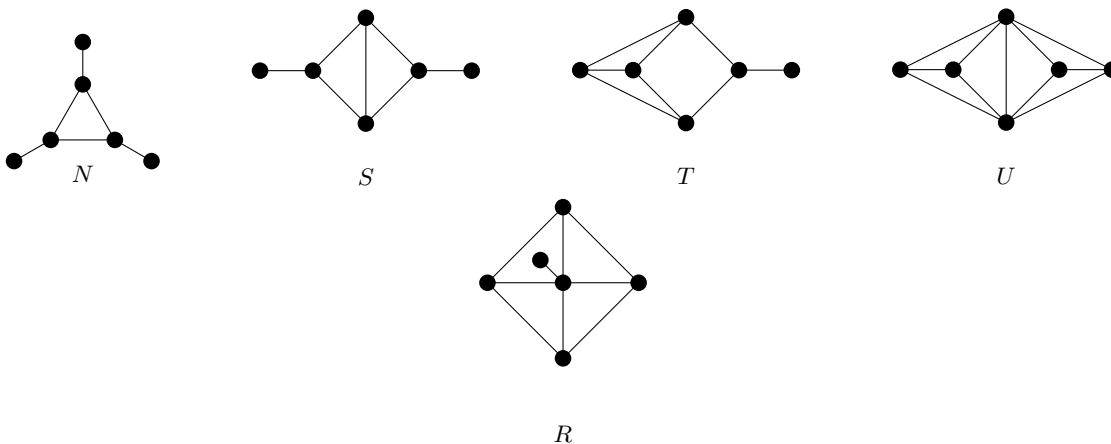


Figure 6. The five induced subgraph obstructions locally equivalent to the net graph N .

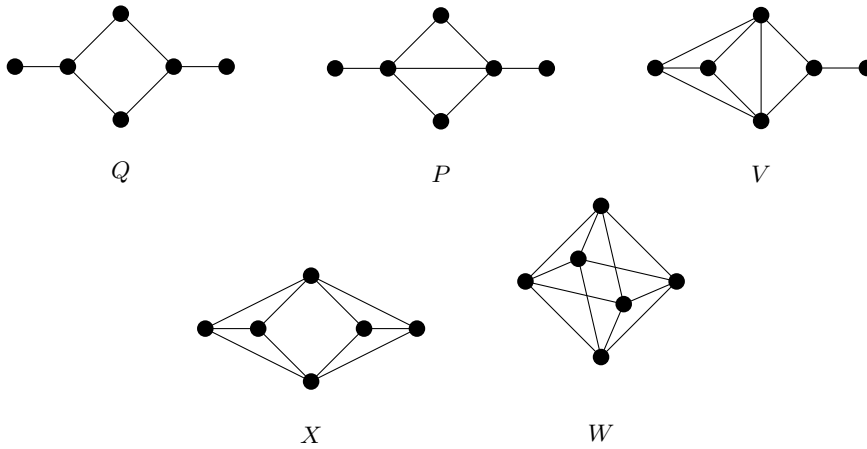


Figure 7. The five induced subgraph obstructions locally equivalent to the half-cube Q .

6 Pivot-Minor Obstructions

Pivot-equivalence is defined with respect to an edge $\{u, v\}$. Define three sets of vertices A, B , and C as $A = \{w | \{u, w\} \in E(G) \text{ and } \{v, w\} \notin E(G)\}$, $B = \{w | \{u, w\} \in E(G) \text{ and } \{v, w\} \in E(G)\}$, $C = \{w | \{u, w\} \notin E(G) \text{ and } \{v, w\} \in E(G)\}$. The result of a *pivot operation* $G \times \{u, v\}$ is defined as the complementation of edges of G between the three sets of vertices A, B , and C , and also between $\{u, v\}$ and $A \cup C$ (“swapping u and v ”).

We say that two graphs, G and H are *pivot-equivalent*, $G \sim_p H$ if H can be obtained from G by a sequence of pivot operations. Pivot-minor containment, $H \preceq_p G$, is defined as H being an induced subgraph of any graph pivot-equivalent to G . H is a *proper pivot-minor* of G , if $H \preceq_p G$ and $|V(H)| < |V(G)|$. It is easy to check that for an edge $\{u, v\}$ in G , $G \times \{u, v\} = G * u * v * u$. Therefore, pivot-minor containment is a special case of vertex-minor containment.

A graph G is a *pivot-minor obstruction* for the class of graphs of linear rank-width at most 1, if $\text{lrw}(G) \geq 2$ and every proper pivot-minor H of G satisfies $\text{lrw}(H) \leq 1$.

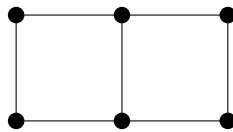


Figure 8. The domino graph

Theorem 26 *The pivot-minor obstructions for the class of graphs of linear rank-width at most 1 are the eight graphs depicted in Figure 9.*

Proof. We define C_k^+ , for $k > 5$, as the cycle on k vertices with a chord between two vertices at distance 3 along the cycle. Figure 8 shows C_6^+ , the domino graph. By applying the pivot operation to an edge between two vertices of degree 2, one sees that $C_k \sim_p C_k^+$ and therefore, for all $k > 6$, $C_{k-2} \preceq_p C_k$. As a result, all non-distance-hereditary graphs contain C_5 or C_6 as a pivot-minor; these graphs are the two pivot-minor obstructions for the class of distance-hereditary graphs [2].

Now consider the induced subgraph obstructions S_i , for $0 \leq i \leq 3$. By applying the pivot operation to an edge that includes the central vertex of one of these graphs, a graph that contains another obstruction as an induced subgraph is generated. For example, applying the pivot operation to such an edge in S_0 and then deleting the non-central vertex of the edge results in obstruction Q . Similarly, S_1 results in obstruction T , S_2 results in X , and S_3 results in U . Therefore, the induced subgraph obstructions S_i are eliminated as pivot-minor obstructions.

To determine the remaining pivot-minor obstructions for the class of graphs with linear rank-width at most 1, we need only identify pivot-equivalent graphs in \mathcal{O}_{dh} . One finds that $N \sim_p R, S \sim_p T, P \sim_p W, X \sim_p V$. There are no further equivalences among these graphs, that would allow reduction of this set of graphs. This can be established readily, by observation, as there are few distinct edges in each proposed obstruction. For example, consider graph N . Applying a pivot operation to a pendant edge results in the identical graph. Applying the pivot operation to the other distinct edge of N , results in graph R . Applying the pivot operation to the pendant edge of R results in R . Applying the pivot operation to an edge between two vertices of degree 3 results in graph N , Applying the pivot operation to the other edge in R results in a graph isomorphic to R . A similar case analysis shows that applying the pivot operation to edges of any of the other pivot-equivalent pairs of graphs in \mathcal{O}_{dh} that were noted above only results in graphs isomorphic to graphs of the pair. \square

7 Conclusion

The celebrated Robertson-Seymour Theorem shows the finiteness of the obstruction sets for classes of graphs that are closed under the taking of minors. However, the cardinality of such an obstruction set can be enormous. It is known that if a class of graphs has bounded rank-width, then the obstruction set is finite [15]. This implies that, for every integer $k \geq 0$, the obstruction set for the class of graphs of linear rank-width at most k is finite. Until now, none of these obstruction sets were known explicitly. In this paper, we have exhibited the exact finite sets of obstructions under taking of induced subgraph, vertex, and pivot minors for the class of graphs having linear rank-width at most 1. A natural next step

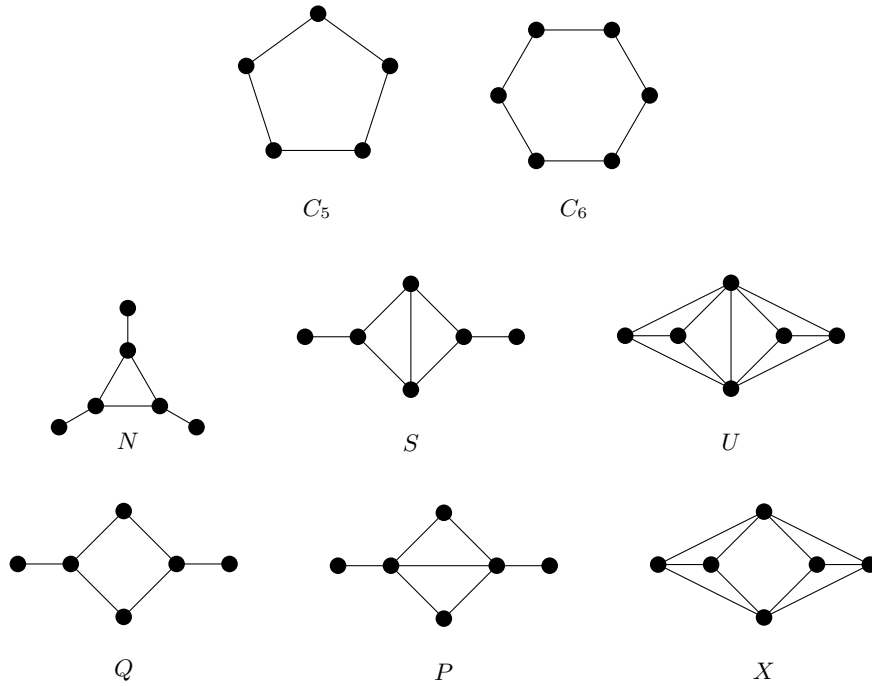


Figure 9. The eight pivot-minor obstructions for linear rank-width at most 1.

would be to determine the obstruction set for the graphs of linear rank-width at most 2. However, we expect the number of obstructions to be large.

8 Acknowledgement

We thank the anonymous referees for their careful scrutiny of our manuscript; their comments helped us to provide a clearer exposition of our results.

References

1. Isolde Adler and Mamadou M. Kanté. Linear rank-width and linear clique-width of trees, 2013. WG'13, Accepted.
2. Hans-Jürgen Bandelt and Henry Martyn Mulder. Distance-hereditary graphs. *J. Comb. Theory, Ser. B*, 41(2):182–208, 1986.
3. Hans L. Bodlaender. A partial k -arboretum of graphs with bounded treewidth. *Theor. Comput. Sci.*, 209(1-2):1–45, 1998.
4. André Bouchet. Circle graph obstructions. *J. Comb. Theory, Ser. B*, 60(1):107–144, 1994.
5. Bruno Courcelle, Johann A. Makowsky, and Udi Rotics. Linear time solvable optimization problems on graphs of bounded clique-width. *Theory Comput. Syst.*, 33(2):125–150, 2000.

6. Bruno Courcelle and Stephan Olariu. Upper bounds to the clique width of graphs. *Discrete Applied Mathematics*, 101(1-3):77–114, 2000.
7. Reinhard Diestel. Graph minors 1: A short proof of the path-width theorem. *Combinatorics, Probability & Computing*, 4:27–30, 1995.
8. 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, *WG*, volume 2204 of *Lecture Notes in Computer Science*, pages 117–128. Springer, 2001.
9. Robert Ganian. Thread graphs, linear rank-width and their algorithmic applications. In Costas S. Iliopoulos and William F. Smyth, editors, *IWOCA*, volume 6460 of *Lecture Notes in Computer Science*, pages 38–42. Springer, 2010.
10. Robert Ganian and Petr Hliněný. Better polynomial algorithms on graphs of bounded rank-width. In Jirí Fiala, Jan Kratochvíl, and Mirka Miller, editors, *IWOCA*, volume 5874 of *Lecture Notes in Computer Science*, pages 266–277. Springer, 2009.
11. Edward Howorka. A characterization of distance-hereditary graphs. *The Quarterly Journal of Mathematics*, 2(28):417–420, 1977.
12. Nancy G. Kinnersley and Michael A. Langston. Obstruction set isolation for the gate matrix layout problem. *Discrete Applied Mathematics*, 54(2-3):169–213, 1994.
13. Daniel Kobler and Udi Rotics. Edge dominating set and colorings on graphs with fixed clique-width. *Discrete Applied Mathematics*, 126(2-3):197–221, 2003.
14. Sang-il Oum. Rank-width and vertex-minors. *J. Comb. Theory, Ser. B*, 95(1):79–100, 2005.
15. Sang-il Oum. Rank-width and well-quasi-ordering. *SIAM J. Discrete Math.*, 22(2):666–682, 2008.
16. Sang-il Oum and Paul D. Seymour. Approximating clique-width and branch-width. *J. Comb. Theory, Ser. B*, 96(4):514–528, 2006.
17. Neil Robertson and Paul D. Seymour. Graph minors. i. excluding a forest. *J. Comb. Theory, Ser. B*, 35(1):39–61, 1983.