Computing the clique-width of large path powers in linear time via a new characterisation of clique-width

Pinar Heggernes* Daniel Meister† Udi Rotics‡

Abstract

Clique-width is one of the most important graph parameters, as many NP-hard graph problems are solvable in linear time on graphs of bounded clique-width. Unfortunately the computation of clique-width is among the hardest problems. In fact we do not know of any other algorithm than brute force for the exact computation of clique-width on any non-trivial large graph class. Another difficulty about clique-width is the lack of alternative characterisations of it that might help in coping with its hardness. In this paper we present two results. The first is a new characterisation of clique-width based on rooted binary trees, completely without the use of labelled graphs. Our second result is the exact computation of the clique-width of large path powers in polynomial time, which has been an open problem since the 1990's. The presented new characterisation is used to achieve this latter result. With our result, large k-path powers constitute the first non-trivial infinite class of graphs of unbounded clique-width whose clique-width can be computed exactly in polynomial time.

1 Introduction

Clique-width is a graph parameter that has many algorithmic applications [4]. In particular, NP-hard graph problems that are expressible in a certain type of monadic second-order logic admit algorithms with linear running time on graphs whose clique-width is bounded by a constant [5, 20]. Unfortunately it is NP-hard to compute the clique-width of a given graph even for cobipartite graphs [8]. Fellows et al. ask whether the computation of clique-width is fixed parameter tractable when parametrised by the clique-width of the input graph [8]. This question is still open. Furthermore, we do not know of an algorithm with running time c^n, where c is a constant. Although clique-width has received a lot of attention recently [1, 3, 7, 8, 13, 15, 16, 17, 19], positive results known on the computation of clique-width so far are very restricted. Graphs of clique-width at most 3 can be recognised in polynomial time, and their exact clique-width can be computed efficiently [6, 2]. Examples of such graph classes are cographs, trees and distance-hereditary graphs [9]. Regarding classes of unbounded clique-width, the class of square grids is the only class for which a polynomial-time clique-width computation algorithm is known [9].

*Department of Informatics, University of Bergen, Norway. Email: `pinar.heggernes@ii.uib.no`
†Theoretical Computer Science, University of Trier, Germany. Email: `daniel.meister@uni-trier.de`
‡Netanya Academic College, Netanya, Israel. Email: `rotics@netanya.ac.il`

Clique-width is often compared to treewidth, as the same set of problems that are efficiently solvable on graphs of bounded clique-width are also efficiently solvable on graphs of bounded treewidth. However, clique-width is more general than treewidth, as graphs of bounded treewidth have bounded clique-width [6], whereas there are graph classes of bounded clique-width whose treewidth is not bounded (for example complete graphs). For such graph classes, the gap between the two parameters may be arbitrarily large [3]. The known results on these two parameters so far make treewidth a much more manageable graph parameter than clique-width. Although treewidth is also NP-hard to compute in general, by Bodlaender's celebrated result graphs of bounded treewidth can be recognised in linear time, and the complexity of computing treewidth is known for most of the well-known graph classes. We can say that treewidth is well understood, whereas the same is not true for clique-width. For example, treewidth has many characterisations via tree decompositions, partial k-trees, embeddings into chordal graphs, graph searching, and forbidden minors. When it comes to clique-width, only little is known about characterisations. Clique-width was originally defined as the smallest number of labels needed for building a graph by application of labelled graph operations. During the last two decades, only one other characterisation has been discovered [6]: For a finite set C of labels, a *C-construction* of a graph G is a sequence $(G_0, \ldots, G_r)$ of C-labelled graphs on the vertex set of G, where each graph G_i is the disjoint union of two C-labelled graphs and G_{i+1} emerges from G_i by changing labels or by adding the edges in one union graph of G_i. The size of set C bounds the clique-width of G from above [6].

In this paper we present a new characterisation of clique-width. It is based on a rooted binary tree that iteratively partitions the vertex set into finally singleton partition sets by obeying certain adjacency conditions. The clique-width of a graph is then determined by the number of partition sets that are active at a time. Using this characterisation, we show that k-path powers with at least $(k+1)^2$ vertices have clique-width $k+2$. This solves a long-standing open problem posed in [9]. A k-path power is the k-power graph of a simple path. The only known result for classes of unbounded clique-width so far has been for square grids: the clique-width of a $k \times k$-grid is k [9]. Note that for each positive integer k, there is only one $k \times k$ grid, whereas there are infinitely many k-path powers.

Some earlier results that emerged from the study of the computational complexity of the relative clique-width and NLC-width problems can be seen as related to our new characterisation. NLC-width is a graph parameter similar to clique-width, and the two parameters differ by at most a factor of 2 [14]. Müller and Urner gave a characterisation of NLC-width through decomposition trees [19]. Our characterisation of clique-width is more than a direct generalisation of previous characterisation results for NLC-width. For NLC-width, the employed precise formulation defines a partition that is unique. The context of the decomposition defined at some other node of the decomposition tree is not required, as it is implicit. This is not the case for clique-width. In particular, the partition at some tree node is *not* unique. This may also explain part of the difficulties that have arisen from understanding and working with clique-width.

2 Definitions and notation

We consider only simple finite undirected graphs. For $G = (V, E)$ a graph, $V = V(G)$ is the *vertex set* of G and $E = E(G)$ is the edge set of G. Edges are denoted as uv, where u and v are *adjacent* in G, or u is a *neighbour* of v in G. For a vertex u of G, the *(open) neighbourhood* of u, denoted as $N_G(u)$, is the set of neighbours of u in G. A graph H is a *subgraph* of G if $V(H) \subseteq V(G)$ and $E(H) \subseteq E(G)$. For a set $X \subseteq V(G)$, the subgraph of G *induced* by X, denoted as $G[X]$, is the subgraph H of G for which for every edge uv of G, $u, v \in V(H)$ implies $uv \in E(H)$.

The definition of clique-width is based on labelled graphs and a set of operations. For an integer $k \geq 1$, a *k-labelled graph* is an ordered triple $G = (V, E, \ell)$ where (V, E) is a graph and $\ell : \{1, \ldots, k\} \to V$. Analogous to graphs, V and E are the vertex and edge set of G, respectively, and ℓ is the *label function* of G. We define four types of operations for k-labelled graphs:

- for $1 \leq i \leq k$ and u a vertex, $i(u)$ creates the k-labelled graph $(\{u\}, \emptyset, \{(u, i)\})$
- for $1 \leq i < j \leq k$ and G a k-labelled graph, $\eta_{i,j}(G)$ is the k-labelled graph that emerges from G by adding all edges between vertices with label i and vertices with label j that are not edges of G
- for $1 \leq i, j \leq k$, $i \neq j$, and G a k-labelled graph, $\rho_{i \to j}(G)$ is the k-labelled graph that emerges from G by changing all labels i into label j
- for G and H k-labelled graphs, $G \oplus H$ is the k-labelled graph on vertex set $V(G) \cup V(H)$ and with edge set $E(G) \cup E(H)$ and each vertex of $G \oplus H$ has the same label as in G or H.

A *k-expression* is built from the four operation types: $i(u)$ is a k-expression, for α and ω k-expressions, $\eta_{i,j}(\alpha)$, $\rho_{i \to j}(\alpha)$ and $(\alpha \oplus \omega)$ are k-expressions. By $\text{val}(\alpha)$, we denote the labelled graph that is defined by α. For a graph G, the *clique-width* of G, denoted as $\text{cwd}(G)$, is the smallest integer k such that there is a k-expression α and a label function ℓ for G with $(V(G), E(G), \ell) = \text{val}(\alpha)$. We say that α is a k-expression for G.

3 Supergroup partitions characterise clique-width

We aim at a simple characterisation of clique-width. We will see that clique-width can be seen as iteratively refining a partition. Previously, clique-width and its variant linear clique-width were investigated by using the graph notion of "group". Let G be a graph and let H be an induced subgraph of G. A set A of vertices of H is called *group* if for every vertex pair u, v from A, the two vertices are indistinguishable with respect to the vertices of G that are not in H, formally, $N_G(u) \setminus V(H) = N_G(v) \setminus V(H)$. The group notion is very useful for understanding linear clique-width [18, 10, 11]. However, it is insufficient for studying clique-width. Such observations were first published by Müller and Urner in their work about the complexity of computing the relative clique-width [19]. For illustrating the situation, consider the graph, G, in Figure 1. The vertex set of the depicted graph is partitioned into $\{a, b, c, d\}$ and $\{e, f\}$, which is indicated by the two rectangles. The groups in the left hand side partition set are $\{a\}$, $\{b\}$ and $\{c, d\}$. Spending one

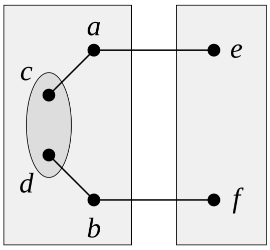

Figure 1: The vertex set of the depicted graph is partitioned into the sets $\{a, b, c, d\}$ and $\{e, f\}$. The maximal groups of $\{a, b, c, d\}$ are $\{a\}, \{b\}, \{c, d\}$.

label per group and then computing the disjoint union with $G[\{e, f\}]$ requires a label for each of e and f, and the two labels must be different from the labels used in $G[\{a, b, c, d\}]$. Hence, for computing G from the disjoint union of $G[\{a, b, c, d\}]$ and $G[\{e, f\}]$ requires five labels. However, if we allow four labels in $G[\{a, b, c, d\}]$, e and f can have the same label as respectively c and d. This requires a total of four labels for computing G from the disjoint union of $G[\{a, b, c, d\}]$ and $G[\{e, f\}]$, and this value is optimal for this partition.

In this section, we will give a characterisation of clique-width by applying the ideas and observations made in the above paragraph. The main idea is to generalise the notion of "group" in induced subgraphs. Let G be a graph and let H be a subgraph of G. Denote by H^{-1} the graph $G \setminus E(H)$.

Definition 3.1 *Let G be a graph and let H be a subgraph of G.*

1) *A set A of vertices of H is called* group *of H if for every vertex pair u, v from A, $N_{H^{-1}}(u) = N_{H^{-1}}(v)$.*

2) *A set A of vertices of H is called* supergroup *of H if for every vertex pair u, v from A, $N_{H^{-1}}(u) \subseteq N_G(v)$.*

3) *A* supergroup partition *for H is a partition $(A_1, \ldots, A_r)$ of $V(H)$ such that $A_1, \ldots, A_r$ are supergroups of H. The* size *of a supergroup partition is the number r of partition classes.*

The definition of group and supergroup requires that vertices that are adjacent in H^{-1} do not belong to the same group or supergroup. Note that *group* could be defined equally by requiring only one inclusion between the two neighbourhood sets. Equality would follow with the pair where the roles of u and v are exchanged. Observe that the notions of group and supergroup are indeed different. Reconsider graph G depicted in Figure 1. Let H be the subgraph of G obtained from deleting edges ae and bf. The set $\{c, e\}$ is not a group of H, since c is not adjacent to a in H^{-1} but e is. In contrast, $\{c, e\}$ is a supergroup of H, since both c and e are adjacent to a in G. The following lemma determines the relationship between the notions of group and supergroup.

Lemma 3.2 *Let G be a graph and let H be a subgraph of G. Then, every group of H is a supergroup of H. If H is an induced subgraph of G then every supergroup of H is a group of H.*

Proof. For every vertex x of G, it holds that $N_{H^{-1}}(x) \subseteq N_G(x)$. Let A be a group of H. For every vertex pair u, v from A, it holds that $N_{H^{-1}}(u) = N_{H^{-1}}(v) \subseteq N_G(v)$, so that A is a supergroup of H indeed.

Now, let H be an induced subgraph of G and let A be a supergroup of H. Thus, for every vertex pair u, v from A, it holds that $N_{H^{-1}}(u) \subseteq N_G(v)$. Let $w \in N_G(v)$ and assume that that $w \notin N_{H^{-1}}(v)$. This means that $w \in N_H(v)$. In particular, w is a vertex of H. Since H is an induce subgraph of G, it follows that w is not adjacent to u in G. Hence, $N_{H^{-1}}(u) \subseteq N_G(v)$ implies $N_{H^{-1}}(u) \subseteq N_{H^{-1}}(v)$, which implies that A is a group of H. ∎

Let G be a graph and let H be a subgraph of G. According to the definition of groups by means of an equivalence relation, the set of maximal groups of H uniquely partitions $V(H)$. In particular, two maximal groups of H are either equal or disjoint. Let $\mathscr{M} = (M_1, \ldots, M_r)$ and $\mathscr{N} = (N_1, \ldots, N_s)$ be partitions of a ground set X. We say that $\mathscr{N}$ is a *refinement* of $\mathscr{M}$, if $N_1 \cup \cdots \cup N_s = M_1 \cup \cdots \cup M_r$ and for every $1 \leq i \leq s$ there is $1 \leq j \leq r$ such that $N_i \subseteq M_j$.

We use the supergroup notion to define an iterated partition of the vertex set of a graph.

Definition 3.3 *Let G be a graph. A* supergroup tree *for G is a rooted binary tree T whose nodes are labelled with partitions of subsets of $V(G)$ such that the following conditions are satisfied:*

1) *every leaf of T is labelled with partition $(\{x\})$ where x is a vertex of G*

2) *let a be an inner node of T with sons b and c, and let a, b, c be labelled with the partitions $(A_1, \ldots, A_p)$, $(B_1, \ldots, B_q)$ and $(C_1, \ldots, C_r)$, respectively; then,*

   - *$(B_1, \ldots, B_q, C_1, \ldots, C_r)$ is a refinement of $(A_1, \ldots, A_p)$*
   - *$(A_1, \ldots, A_p)$ is a supergroup partition for $G[B_1 \cup \cdots \cup B_q] \oplus G[C_1 \cup \cdots \cup C_r]$.*

The size *of T is the largest size of a supergroup partition a node of T is labelled with.*

Note that the definition of supergroup tree ensures that each vertex be the label of exactly one leaf of the tree. We show a tight correspondence between clique-width expressions and supergroup trees. This correspondence is developed in two steps. For the first step, which also speaks about an efficient transformation algorithm, we assume that a given clique-width expression is "reasonable" in the sense that it contains only useful operations. Let T be a rooted tree and let a be a node of T. By T_a, we denote the subtree of T that is rooted at a. Let G be a graph and let α be a clique-width expression for G. Let $T = T[\alpha]$ be clique-width tree that is defined by α. For a node a of T and a vertex x of G, we say that x *occurs* at a in T, if $i(x)$ for some label i is the label of some leaf in T_a. Alternatively, x occurs at a if x is vertex of the graph that is defined by the clique-width expression represented by T_a.

Lemma 3.4 *There is a function that, given a graph G and a k-expression α for G, computes a supergroup tree for G of size at most k in $\mathcal{O}(n^2)$ time.*

Proof. We give an algorithm that computes a tree and assigns to each of its nodes a partition of some subset of the vertex set of the given graph. So, let G be a graph and let α be a k-expression for G, where $k \geq 1$. We will describe the construction of a supergroup tree for G of size at most

k. Denote by $T[\alpha]$ the clique-width tree of α. Note that there is a 1-to-1 correspondence between the leaves of $T[\alpha]$ and the vertices of G. Obtain the rooted binary tree T from $T[\alpha]$ as follows: there is a 1-to-1 correspondence between the leaves of $T[\alpha]$ and the leaves of T and there is a 1-to-1 correspondence between the $\oplus$-labelled nodes of $T[\alpha]$ and the nodes of T such that for every inner node a of T, if B and C are the sets of vertices that appear in the two subtrees of T rooted at a, B and C are the sets of vertices appearing in the two subtrees of $T[\alpha]$ rooted at the image of a in $T[\alpha]$. Informally spoken, T is obtained from $T[\alpha]$ by shrinking vertices that are not labelled with $\oplus$ and keeping the structure. In other words, $T[\alpha]$ can be derived from T by subdividing edges and adding labels. We add labels to the nodes of T in the following way. Let a be a node of T, and let a' be the node of $T[\alpha]$ that corresponds to a. Let X be the set of vertices that occur at a' in $T[\alpha]$ and let $(X_1, \dots, X_r)$ be the partition of X that is defined by the labels at a'. Note that $r \leq k$. Then, label a in T with $(X_1, \dots, X_r)$. We show that T with the defined labels is a supergroup tree for G.

For showing that T is a supergroup tree for G, we verify the conditions of Definition 3.3 for each node of T. Let a be a node of T. Let a' be the node of $T[\alpha]$ that corresponds to a, and let A be the set of vertices of G that occur at a' in $T[\alpha]$. We distinguish between two cases for a. As the first case, let a be a leaf of T. Then, a' is a leaf of $T[\alpha]$, and there is exactly one vertex that occurs at a' in $T[\alpha]$. This, the label of a in T is (X), and this satisfies the conditions of Definition 3.3. As the second case, let a be an inner node of T. Since T is a binary tree, a has two sons, b and c, in T. Let b' and c' be the vertices of $T[\alpha]$ that correspond to b and c, respectively. Let B and C be the sets of vertices of G that occur at respectively b' and c' in $T[\alpha]$. Due to the properties of clique-width expressions and thus $T[\alpha]$, $A = B \cup C$ and $B \cap C = \emptyset$. Let $(A_1, \dots, A_p)$, $(B_1, \dots, B_q)$ and $(C_1, \dots, C_r)$ be the partitions the nodes a, b, c are labelled with in T. Following the conditions in Definition 3.3, we first show that $(B_1, \dots, B_q, C_1, \dots, C_r)$ is a refinement of $(A_1, \dots, A_p)$. Since $(B_1, \dots, B_q)$ is a partition of B and $(C_1, \dots, C_r)$ is a partition of C, $(B_1, \dots, B_q, C_1, \dots, C_r)$ is a partition of A. And since $(A_1, \dots, A_p)$ is a partition of the vertices that occur at a' in $T[\alpha]$ due to the definition of T, $(A_1, \dots, A_p)$ is also a partition of A. Let $X \in \{B_1, \dots, B_q\}$. Due to the definition of T, the vertices in X have the same label at b' in $T[\alpha]$, and so, by the properties of clique-width expressions, the vertices in X have the same label at a' in $T[\alpha]$. The definition of $(A_1, \dots, A_p)$ implies that there exists a (unique) number $1 \leq i \leq p$ with $X \subseteq A_i$. Since the same holds for every $X \in \{C_1, \dots, C_r\}$ by analogous arguments, it follows that $(B_1, \dots, B_q, C_1, \dots, C_r)$ is indeed a refinement of $(A_1, \dots, A_p)$.

For the other condition in Definition 3.3, we have to show that $(A_1, \dots, A_p)$ is a supergroup partition for $G[B] \oplus G[C]$. Let $1 \leq i \leq r$ and let v be a vertex in A_i. By a symmetry argument, we can assume that $v \in B$. Let u be a vertex of G such that $uv \in E(G)$. Let $uv \notin E(G[B]) \cup E(G[C])$. Due to the definition of $G[B] \oplus G[C]$, it holds that $u \notin B$. We look into the situation in $T[\alpha]$. Edge uv is created in $T[\alpha]$ in a node above a', say in node d'. When uv is created in $T[\alpha]$, u and v have different labels; in particular, $u \notin A_i$. Furthermore, all vertices of G that occur at d' in $T[\alpha]$ and have the same label as v at d' are made adjacent to u. Due to the properties of clique-width expressions, all vertices of G that occur at a' in $T[\alpha]$ and have the same label as v at a' occur at d' and have the same label as v at d'. Thus, all vertices in A_i are adjacent to u in G. We conclude that $(A_1, \dots, A_p)$ is a supergroup partition for $G[B] \oplus G[C]$. And by induction, we conclude that T is a supergroup tree for G. Furthermore, since the size of each assigned supergroup partition is at most k, it follows that T is a supergroup tree for G

of size at most k.

To complete the proof of the lemma, it remains to consider the running time of the described algorithm constructing T. Constructing $T[\alpha]$ from α takes linear time in the number of operations in α. From $T[\alpha]$, T without labels can be obtained in linear time. The labels of the leaves of T are easily found in the leaves of $T[\alpha]$. In a bottom-top manner simultaneously in T and $T[\alpha]$, the labels of each inner node of T can be computed from the labels of the sons and by determining the development of vertex labels in α. Since T contains only $2n-1$ nodes and each node is labelled with a vertex partition of a subset of $V(G)$, and since α contains at most $\mathcal{O}(n^2)$ many operations, we conclude the claimed running time. ∎

Lemma 3.5 *Let G be a graph. Let T be a supergroup tree for G of size t. Then, G has a k-expression α for some $k \leq t$.*

Proof. Let a be a node of T and let $(A_1, \ldots, A_p)$ be the supergroup partition that a is labelled with in T. Let $A =_{\text{def}} A_1 \cup \cdots \cup A_p$. We show by induction that there exists a clique-width expression for $G[A]$ that uses at most as many labels as the largest size of a labelling supergroup partition in T_a and such that the partition of A induced by the used labels is equal to $(A_1, \ldots, A_p)$. If a is a leaf of T then $A = \{x\}$ for some vertex x of G, and $1(x)$ is a clique-width expression that satisfies the claims. Now, let a be an inner node of T. Then, a has two sons, b and c, in T. Let b and c be labelled with $(B_1, \ldots, B_q)$ and $(C_1, \ldots, C_r)$, respectively. Let $B =_{\text{def}} B_1 \cup \cdots \cup B_q$ and $C =_{\text{def}} C_1 \cup \cdots \cup C_r$. Due to induction hypothesis, there are clique-width expressions β and γ for respectively $G[B]$ and $G[C]$ that use at most as many labels as the largest size of a labelling supergroup partition in respectively T_b and T_c, and the partitions of B and C induced by the labels at the end of β and γ correspond to $(B_1, \ldots, B_q)$ and $(C_1, \ldots, C_r)$. We show that we can obtain a similar clique-width expression for $G[A] = G[B \cup C]$. Note that $(\beta \oplus \gamma)$ is a clique-width expression for $G[B] \oplus G[C]$. The expression to be constructed is obtained in two steps: first, we change labels, and second, we add edges. For the first step, we change labels for vertices from B and C such that the label of each vertex corresponds to the index of its partition class in $(A_1, \ldots, A_p)$. We consider β. As a preparation step, assume that there are $1 \leq i < i' \leq q$ and $1 \leq j \leq p$ such that $B_i \cup B_{i'} \subseteq A_j$. Let k and l be the labels of the vertices from respectively B_i and $B_{i'}$ in β. We construct $\beta' =_{\text{def}} \rho_{k \to l}(\beta)$. Repeating this construction, we obtain an expression β^+ for $G[B]$ such that for every $1 \leq i \leq p$, A_i contains at most one label class of $\text{val}(\beta^+)$. Let $\varphi : B \to \{1, \ldots, p\}$ be such that $x \in A_{\varphi(x)}$ for every $x \in B$. We obtain expression β^* from β^+ by changing the occurrence of each label such that the label of x in $\text{val}(\beta^*)$ is equal to $\varphi(x)$. It is important to note that this exchange is indeed possible, since labels are changed in the expression itself and not by applying a sequence of relabel operations at the end. Applying the same construction to γ, we obtain γ^* for $G[C]$. It follows that the label partition of $\text{val}(\beta^* \oplus \gamma^*)$ is exactly $(A_1, \ldots, A_p)$, where i is the label of the vertices in A_i, $1 \leq i \leq p$.

For the second step, we consider the edges of $G[A]$ that are not contained in $G[B] \oplus G[C]$ and thus are not contained in $\text{val}(\beta^* \oplus \gamma^*)$. Let $uv \in E(G[A])$ with $uv \notin E(G[B] \oplus G[C])$. Without loss of generality, we can assume $u \in B$ and $v \in C$. Let $1 \leq i \leq p$ be such that $u \in A_i$. Suppose that $v \in A_i$. The definition of supergroup partition for $G[B] \oplus G[C]$ implies that each vertex in A_i is adjacent to v in G, in particular, v itself. This is not possible, since $vv \notin E(G)$, so that $v \notin A_i$. Let $1 \leq j \leq p$ such that $v \in A_j$. We have seen that $i \neq j$. Due to the definition of supergroup partitions, $xy \in E(G)$ for every vertex pair x, y of G with $x \in A_i$ and $y \in A_j$. Exactly these edges are created by the operation $\eta_{i,j}$ applied to $(\beta^* \oplus \gamma^*)$. Remember that the vertices in A_i are the vertices with label i in $\text{val}(\beta^* \oplus \gamma^*)$, and the vertices in A_j are the vertices with label j. Repeating this construction, we obtain a clique-width expression α for $G[A]$ that uses at most $\max\{p, ||\beta^*||, ||\gamma^*||\} = \max\{p, ||\beta||, ||\gamma||\}$ labels. Applying the induction hypothesis, it follows that $||\alpha||$ is at most the largest size of a labelling supergroup partition in T_a. Note that the order in which the edge creation operations are added is not important for the definition of α.

By inductively applying the construction, we obtain a clique-width expression for the root vertex of T and thus for G. The number of used labels is at most the size of T, which proves the claim of the lemma. ■

Theorem 3.6 *Let G be a graph. The smallest size of a supergroup tree for G is equal to the clique-width of G.*

Proof. Let $k =_{\text{def}} \text{cwd}(G)$ and let α be a k-expression for G. Note that α exists due to definition. Let t be the smallest size of a supergroup tree for G and let T be a supergroup tree for G of size at most t. Due to Lemma 3.4, $t \leq k$, and due to Lemma 3.5, $k \leq t$. Hence, $t = k$. ■

How can we show lower bounds on the clique-width of graphs by using the supergroup tree? Two steps: we show lower bounds on the size of supergroup partitions of special situations $(G[B] \oplus G[C])$, and we show that every interesting supergroup tree must contain a node where a special situation occurs.

Supergroup trees are useful for proving results of clique-width, as we will show in the next sections. They seem also useful for algorithmic purposes. In this context, it may be desirable to have small supergroup trees, that can be represented in space linear in the size of the graph. The representation that is computed by the algorithm in Lemma 3.4 constructs a tree of size $\mathcal{O}(n)$ but labels every node of the tree by an explicit description of the supergroup partition. This requires total $\mathcal{O}(n^2)$ space, which also determines the running time of the given algorithm. One can think of implicit representations and hereby reducing the space required for the tree and also the running time of the algorithm.

Lemma 3.7 *Let G be a graph and let $A, B \subseteq V(G)$ where $A \cap B = \emptyset$.*

1) *Let X be a maximal group of $G[A]$. There is a maximal group C of $G[A] \oplus G[B]$ such that $X \subseteq C$.*

2) *Let X and Y be maximal groups of $G[A]$ where $X \neq Y$. There is no supergroup Z of $G[A] \oplus G[B]$ such that $X \cap Z \neq \emptyset$ and $Y \cap Z \neq \emptyset$.*

Proof. Let $H =_{\text{def}} G[A] \oplus G[B]$. For proving the first statement, it suffices to show that X is a group of H. For a contradiction, suppose that there are vertices $x, y \in X$ and $w \in V(G)$ such that $xw \in E(H^{-1})$ and $yw \notin E(H^{-1})$. Note that $w \notin A$. Furthermore, due to the definition of group, $xw \in E(H^{-1})$ implies $xw, yw \in E(G)$. Thus, $yw \in E(H)$, and therefore, $w \in B$.

However, H contains no edge between vertices from A and vertices from B, so this yields a contradiction.

For the second statement, let X and Y be maximal groups of $G[A]$. Due to the definition of H, it holds that $N_{H^{-1}}(u) = N_{(G[A])^{-1}}(u)$ for every vertex $u \in A$. Let $U \subseteq X$ and $V \subseteq Y$, and assume that $U \cup V$ is a supergroup of H, which means $N_{H^{-1}}(x) \subseteq N_G(y)$ for every vertex pair $x, y \in U \cup V$. Thus, $N_{(G[A])^{-1}}(x) = N_{H^{-1}}(x) \subseteq N_G(y)$ for every vertex pair $x, y \in U \cup V$, so that $U \cup V$ is a supergroup of $G[A]$. We apply Lemma 3.2 and see that $U \cup V$ is a group of $G[A]$. So, $U \cup V$ is contained in a unique maximal group of $G[A]$. It follows that $X \cap Y \neq \emptyset$, which implies $X = Y$, and this proves the statement. ∎

The result of the second statement of Lemma 3.7 shows that the number of maximal groups of $G[A]$ is a lower bound on the size of any supergroup partition for $G[A] \oplus G[B]$. This corollary will be useful in later results.

4 The clique-width of large path powers

We want to determine the exact clique-width of a class of proper interval graphs. Let $\langle v_1, \ldots, v_n \rangle$ be an ordering of the vertices $v_1, \ldots, v_n$. Let $k \geq 1$. Graph G on vertex set $\{v_1, \ldots, v_n\}$ and with v_i and v_j adjacent if and only if $0 < |i - j| \leq k$ is the *k-path power* on n vertices. A *path power* is a graph that is isomorphic to a k-path power for some $k \geq 1$. Every path power is a proper interval graph. The linear clique-width of arbitrary path powers is completely characterised.

Theorem 4.1 ([12]) *Let G be a k-path power on n vertices, with $k \geq 1$ and $n \geq k + 2$.*

- *If $n \geq k(k + 1) + 2$ then* $\mathrm{lcwd}(G) = k + 2$.
- *If $k + 2 \leq n \leq k(k + 1) + 1$ then* $\mathrm{lcwd}(G) = \lceil \frac{n-1}{k+1} \rceil + 1$.

Note that k-path powers on at most $k + 1$ vertices are complete graphs and therefore have linear clique-width at most 2. The result of Theorem 4.1 provides an upper bound on the clique-width of path powers. A lower bound of k for k-path powers on at least $(k+1)^2$ vertices is known [9]. In this section, we want to close the gap for k-path powers on at least $(k + 1)^2$ vertices by showing a lower bound on the clique-width that matches the upper bound of $k + 2$ provided by Theorem 4.1. The graphs that we will study for obtaining the result, we will call *proper interval squares*, *squares* for short. Let $n \geq 2$. We denote by Q_n the $(n - 1)$-path power on n^2 vertices. The vertices of Q_n are $v_{1,1}, \ldots, v_{n,1}, v_{1,2}, \ldots, v_{n,n}$, and the edges of Q_n are determined by vertex ordering $\langle v_{1,1}, \ldots, v_{n,1}, v_{1,2}, \ldots, v_{n,n} \rangle$ in the sense of the above definition of k-path powers. This means that for each pair i, j:

$$N_{Q_n}[v_{i,j}] = \left\{ v_{i+1,j-1}, \ldots, v_{n,j-1}, v_{1,j}, \ldots, v_{n,j}, v_{1,j+1}, \ldots, v_{i-1,j+1} \right\};$$

in border cases, some of the listed vertices may not exist, that we simply exclude in such cases. We partition the vertices of Q_n: a *column* of Q_n is the set of the vertices $v_{1,j}, \ldots, v_{n,j}$. We often speak of "column j", which means exactly the vertices $v_{1,j}, \ldots, v_{n,j}$. Small examples of proper interval squares are depicted in Figure 2, where the vertices are arranged analogous to this representation.

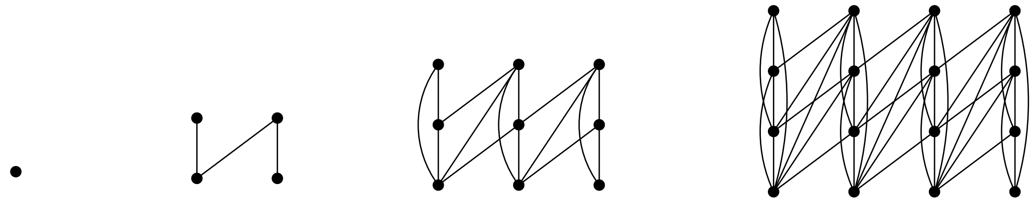

Figure 2: Depicted are four proper interval squares, Q_1, Q_2, Q_3 and Q_4. For each graph Q_n, the upper left and lower right vertex are respectively $v_{1,1}$ and $v_{n,n}$.

For showing the lower bound on the clique-width of Q_n, we will apply Theorem 3.6, which means we will determine a lower bound on the size of supergroup trees for Q_n. The proof is partitioned into two major parts. The first part will determine lower bounds on the size of supergroup partitions for particular subgraphs of Q_n. The second part will show the structure of subgraphs in supergroup trees for Q_n. According to these two proof parts, this section is partitioned into two subsections, each covering one of the two parts.

4.1 Lower bounds on the size of supergroup partitions

We want to determine lower bounds on the size of supergroup partitions of particular subgraphs of Q_n, that are the disjoint union of two induced subgraphs of Q_n. For obtaining the desired lower bounds, we will consider two types of situations: induced subgraphs or disjoint union of two induced subgraphs. For induced subgraphs, the notions of group and supergroup coincide due to Lemma 3.2. In this case, we will determine a lower bound on the number of maximal groups. We will see in the proof of Lemma 4.4 that this lower bound also provides a lower bound for the second case, the disjoint union of two induced subgraphs. Let $A \subseteq V(Q_n)$ and let $1 \leq s \leq n$. The *s-boundary* of A is the set $\{v_{p_1,q_1}, \ldots, v_{p_r,q_r}\}$ of vertices from A such that for every $1 \leq i \leq r$, $q_i < s$ and $v_{p_i,q_i+1}, \ldots, v_{p_i,s-1} \notin A$. We say that column s is *full* in A if $\{v_{1,s}, \ldots, v_{n,s}\} \subseteq A$, and we say that column s is *empty* in A if $v_{1,s}, \ldots, v_{n,s} \notin A$.

Lemma 4.2 ([12]) *Let $A \subseteq V(Q_n)$, and let e be an empty column of A. The vertices of the e-boundary of A are in pairwise different maximal groups of $Q_n[A]$.*

Let $A \subseteq V(Q_n)$. We say that A has the *filled row* property if for all $v_{i,j}, v_{i,j'} \in A$ where $j < j'$, $\{v_{i,j}, \ldots, v_{i,j'}\} \subseteq A$. Informally, for two vertices in A from the same row, all vertices between them in that row must be contained in A.

Lemma 4.3 *Let $A \subseteq V(Q_n)$. Let f be smallest such that $\{v_{1,f}, \ldots, v_{n,f}\} \subseteq A$, and let there be $e > f$ such that $v_{1,e}, \ldots, v_{n,e} \notin A$. Then, A satisfies one of the following conditions:*

- *$|A| = n$*
- *$f \leq 2$ and $v_{1,1} \in A$ and A has the filled row property*
- *$Q_n[A]$ has more than n maximal groups.*

Proof. Without loss of generality, we can choose e such that for every $f < j < e$, $\mathcal{B}_j$ is a non-empty column. Let $\Phi = \{v_{1,j_1}, \ldots, v_{n,j_n}\}$ be the e-boundary of A. Note that Φ is well-defined due to the assumptions about $\mathcal{B}_f$ and $\mathcal{B}_e$. Due to Lemma 4.2, the vertices in Φ are in pairwise different maximal groups, so that A has at least n maximal groups. We assume that $|A| \geq n+1$. We show that $Q_n[A]$ has at least $n+1$ maximal groups if A does not have the filled row property, if $f \geq 3$ or if $v_{1,1} \notin A$.

Assume that there are $1 \leq i \leq n$ and $e < j \leq n$ such that $v_{i,j} \in A$. We choose i and j such that first j is smallest possible and then i is smallest possible. This particularly means that $\mathcal{B}_{j-1}$ is an empty column in A, and $e \leq j-1$. If $i < n$ then $v_{i,j}$ is adjacent to $v_{n,j-1}$ and no vertex in Φ is adjacent to $v_{n,j-1}$. Thus, the vertices in $\Phi \cup \{v_{n,j-1}\}$ are in pairwise different maximal groups of $Q_n[A]$. If $i = n$ then, due to the choice of i, $v_{1,j} \notin A$. Since $v_{1,j}$ is not adjacent to any vertex in Φ, it follows that the vertices in $\Phi \cup \{v_{i,j}\}$ are in pairwise different maximal groups of $Q_n[A]$. In both cases, $Q_n[A]$ has more than n maximal groups.

As the next case, we consider vertices that are in columns $\mathcal{B}_f, \ldots, \mathcal{B}_{e-1}$. Suppose that $Q_n[A]$ has (exactly) n maximal groups. With the considerations from the first paragraph of the proof, it follows that every maximal group contains a vertex from Φ. Let a, b be such that there is c with $f < b < c < e$ and $v_{a,b} \notin A$ and $v_{a,c} \in A$. We choose a and b such that $a + b$ is smallest possible. Note that $b \geq 2$ and $v_{a,b-1} \in A$ and $v_{a,b-1} \notin \Phi$ because of $v_{a,c}$. Let X be the maximal group of $Q_n[A]$ which contains $v_{a,b-1}$. Let u be a vertex in $X \cap \Phi$; let $u = v_{i,j}$. If $j < b-1$ then $a \neq i$ and $v_{i,b-1}$ is not in A and distinguishes u and $v_{a,b-1}$, in contradiction to the definition of group. If $j = b-1$ and $i < a$ then u and $v_{a,b-1}$ are distinguished by $v_{i,b}$, if $j = b-1$ and $i > a$ or if $j = b$ then u and $v_{a,b-1}$ are distinguished by $v_{a,b}$. All the cases lead to a contradiction, so that $b+1 \leq j \leq e-1$. Since $v_{a,b-1}$ is not adjacent to any vertex from column $\mathcal{B}_j$, the definition of group implies that $\mathcal{B}_j$ is full in A and all the neighbours of u in $\mathcal{B}_{j+1}$ are in A. Let y be a vertex from $\mathcal{B}_b$ in A; such a vertex exists, since $\mathcal{B}_b$ is non-empty in A. Let Y be the maximal group of $Q_n[A]$ which contains y. Since $v_{a,b}$ distinguishes $v_{a,b-1}$ and y, it holds that $Y \neq X$. Let $v_{i',j'}$ be a vertex in $Y \cap \Phi$. Since $v_{i',j'}$ must be adjacent to $v_{a,b}$ and since $j' \geq j$, it follows that $j' = b+1$. Note that $X \neq Y$ implies $v_{i,j} \neq v_{i',j'}$, and since $v_{i,j}, v_{i',j'} \in \Phi$, $i \neq i'$. If $i < i'$ then neighbour $v_{i,b+2}$ of $v_{i',b+1}$ is a vertex in A, if $i' < i$ then neighbour $v_{i',j+1}$ of $v_{i,j}$ is a vertex in A. Both cases yield a contradiction to the definition of Φ as the set of the n-boundary vertices of A. This completes this case.

Assume that $f \geq 3$. Due to the choice of f, the columns $\mathcal{B}_1, \ldots, \mathcal{B}_{f-1}$ are not full. Let $1 \leq i \leq n$ and $1 \leq j < f$ be such that $v_{i,j} \in A$, where we first choose j smallest possible and then i smallest possible. If $j = 1$ then $v_{i,j}$ is not in group with any vertex from Φ in $Q_n[A]$ due to a vertex from $\mathcal{B}_1$ that is not contained in A. If $j \geq 2$ then $v_{i,j}$ is not in group with any vertex from Φ in $Q_n[A]$ because of $v_{1,j}$ or $v_{n,j-1}$. It follows that $Q_n[A]$ has at least $n+1$ maximal groups or the columns $\mathcal{B}_1, \ldots, \mathcal{B}_{f-1}$ are empty in A. So, let $\mathcal{B}_1, \ldots, \mathcal{B}_{f-1}$ be empty in A. The vertices $v_{1,f}, \ldots, v_{n,f}$ are in pairwise different maximal groups of $Q_n[A]$, and each of $v_{1,f}, \ldots, v_{n-1,f}$ is in a singleton maximal group of $Q_n[A]$. Due to our assumptions, $e \geq f+2$. If $v_{n,e-1} \in A$ then $v_{n,e-1}$ is in a singleton maximal group of $Q_n[A]$, if $v_{n,e-1} \notin A$, no vertex from $\mathcal{B}_{e-1}$ is in group with $v_{n,f}$ in $Q_n[A]$. It follows in both cases that $Q_n[A]$ must contain at least $n+1$ maximal groups.

For the last case, assume $f \leq 2$. Assume that $\mathcal{B}_1$ is empty in A. Note that the vertices $v_{1,2}, \ldots, v_{n,2}$ are in pairwise different maximal groups of $Q_n[A]$; they are distinguished from each other by the vertices from $\mathcal{B}_1$. Suppose that $Q_n[A]$ has at most n maximal groups. Then, all vertices from $A \setminus \{v_{1,2}, \ldots, v_{n,2}\}$ must be in the same maximal group as a vertex from $\mathcal{B}_2$. No vertex from the columns $\mathcal{B}_3, \ldots, \mathcal{B}_n$ is adjacent to a vertex from $\mathcal{B}_1$. Hence, all vertices from $A \setminus \{v_{1,2}, \ldots, v_{n,2}\}$ must be in the same maximal group as $v_{n,2}$. Similar to the previous case, if $v_{n,e-1} \in A$ then $v_{n,2}$ and $v_{n,e-1}$ are different and cannot be in the same maximal group in $Q_n[A]$, and if $v_{n,e-1} \notin A$ then no vertex from $\mathcal{B}_{e-1}$ can be in the same maximal group as $v_{n,2}$ in $Q_n[A]$. We obtain a contradiction. Assume that $\mathcal{B}_1$ is not empty in A. If $v_{1,1} \notin A$ then no vertex from $\mathcal{B}_1$ is in the same maximal group as any vertex from Φ, so that $Q_n[A]$ contains at least $n+1$ maximal groups. This completes the proof of the lemma. ∎

The result of Lemma 4.3 gives a strong characterisation of induced subgraphs of Q_n with only few maximal groups. One of the main applications of this result will be that induced subgraphs of Q_n with few (namely at most n) maximal groups do not have "holes". For an induced subgraph H of Q_n, a hole here means the situation that there are $1 \leq i \leq n$ and $1 \leq j < j' < j'' \leq n$ such that $v_{i,j}, v_{i,j''} \in V(H)$ and $v_{i,j'} \notin V(H)$.

We want to determine a lower bound on the size of supergroup partitions for specific subgraphs of Q_n. For $1 \leq i \leq n$ and $1 < j \leq n$, we call (A, B) a *partial $[i,j]$-partition* of $V(Q_n)$ if $A \subseteq \{v_{1,1}, \ldots, v_{n,j-1}\} \cup \{v_{1,j}, \ldots, v_{i-1,j}\}$ and $B \subseteq \{v_{i,j}, \ldots, v_{n,j}\} \cup \{v_{1,j+1}, \ldots, v_{n,n}\}$. We use partial partitions for defining subgraphs of Q_n.

Lemma 4.4 *Let $1 \leq i \leq n$ and $1 < j \leq n$ and let (A, B) be a partial $[i,j]$-partition of $V(Q_n)$ such that A has a full column and B is non-empty. Furthermore, let $j = n$ imply $i = 1$ and let $j = n$ and $v_{n,n} \in B$ imply $|B| \geq 2$. Then, every supergroup partition for $Q_n[A] \oplus Q_n[B]$ has size at least $n+1$.*

Proof. If B contains no vertex from column j then (A, B) is also a partial $[1, j+1]$-partition of $V(Q_n)$. Note that $B \neq \emptyset$ implies $j < n$. Iterating this argument, we can henceforth assume that B contains a vertex from column j. Let $(X_1, \ldots, X_k)$ be an arbitrary supergroup partition for $Q_n[A] \oplus Q_n[B]$. By assumption about j, column n is empty in A. Thus, the vertices of the n-boundary of A are in pairwise different maximal groups of $Q_n[A]$ due to Lemma 4.2. Since there is a full column in A by our assumption, the n-boundary of A consists of n vertices. Applying Lemma 3.7, it follows that there are (exactly) n supergroups of $(X_1, \ldots, X_k)$ that contain an n-boundary vertex of A. If $k \geq n+1$ then $(X_1, \ldots, X_k)$ "satisfies" the lemma. Otherwise, $k = n$, and each X_i contains an n-boundary vertex of A. Let s be smallest such that $v_{s,j} \in B$. Note that (A, B) is a partial $[s,j]$-partition of $V(Q_n)$. We distinguish between $s = n$ and $s < n$.

Suppose that $s = n$. Note that our assumptions directly imply $j < n$. Let $1 \leq i \leq k$ be such that $v_{n,j} \in X_i$. Let v be the vertex from the n-boundary of A with $v \in X_i$. Remember that v exists and is unique due to the results of the first paragraph. Since $v_{n,j}$ is adjacent to all vertices from column j in Q_n, it follows that v is not from column j. Since $v_{n-1,j} \notin B$ due to assumption $s = n$, it follows that $v_{n-1,j} \notin X_i$ and thus v must be adjacent to $v_{n-1,j}$. This directly implies that $v = v_{n,j-1}$. Since $v_{1,j+1}$ would distinguish $v_{n,j}$ and v, it holds that $v_{1,j+1} \in B$. Let $1 \leq i' \leq k$ be such that $v_{1,j+1} \in X_{i'}$. Since $v_{1,j} \notin B$, $v_{n,j}$ and $v_{1,j+1}$ are distinguished by $v_{1,j}$, and thus, $i \neq i'$. Let v' be the vertex from the n-boundary of A with $v' \in X_{i'}$. Since $v_{n-1,j}$ is not in B and adjacent to $v_{1,j+1}$, it follows that v' is adjacent to $v_{n-1,j}$. And since $v_{1,j+1}$ and v' must be non-adjacent in Q_n, v' is either $v_{n,j-1}$ or $v_{1,j}$. The former case would contradict $i \neq i'$. The latter case implies that the vertices from X_i and $X_{i'}$ are pairwise adjacent in Q_n because of $v_{1,j}v_{n,j} \in E(Q_n) \setminus E(Q_n[A] \oplus Q_n[B])$, which yields a contradiction because of $v_{n,j-1}v_{1,j+1} \notin E(Q_n)$. So, $s < n$ must hold.

Let $1 \leq t \leq k$ be such that $v_{s,j} \in X_t$. Suppose that $v_{n,j} \notin B$. Then, $v_{s,j}$ is distinguished by $v_{n,j}$ from all vertices $v_{1,1}, \ldots, v_{n,j-1}$, which means that X_t does not contain any vertex from the columns $1, \ldots, j-1$. And since $v_{s,j}$ is adjacent to every other vertex from column j, X_t cannot contain any vertex from column j that is in A. Thus, X_t does not contain any vertex from A. In particular, X_t does not contain any vertex from the n-boundary of A, which contradicts the assumptions from the first paragraph. We conclude $v_{n,j} \in B$. Let $1 \leq p \leq k$ be such that $v_{n,j} \in X_p$. Since $v_{s,j}$ is distinguished from $v_{n,j}$ by $v_{n,j-1}$, it follows that $p \neq t$. Let v_p and v_t be the vertices from the n-boundary of A that are contained in X_p and X_t, respectively. Since $v_t \in A$ and $v_{s,j} \in B$ and v_t and $v_{s,j}$ are in the same supergroup X_t, it follows that v_t and $v_{s,j}$ are non-adjacent in Q_n. Consider $v_{s-1,j}$ or $v_{n,j-1}$, depending on whether $s \geq 2$ or $s = 1$. Since $v_{s-1,j} \notin B$ in case of $s > 1$ or $v_{n,j-1} \notin B$ in case of $s = 1$, and in each case the considered vertex is adjacent to $v_{s,j}$ in Q_n, it follows from the definition of supergroup that the vertex is adjacent also to v_t, and since v_t is non-adjacent to $v_{s,j}$, we conclude that $v_t = v_{s,j-1}$. We distinguish between $s > 1$ and $s = 1$. Suppose $s > 1$. If v_p is adjacent to $v_{s,j}$ then, due to the definition of supergroups, the vertices in X_t and X_p are pairwise adjacent, in particular, $v_{n,j}$ is adjacent to $v_{s,j-1}$. This yields a contradiction to the definition of Q_n. Thus, v_p is non-adjacent to $v_{s,j}$, and therefore, v_p is non-adjacent to v_t in Q_n. Now, observe that $v_{1,j} \notin B$ due to $s \geq 2$, so that v_p and $v_{n,j}$ are adjacent to $v_{1,j}$ (the latter by definition, the former due to the properties of supergroups). This means that v_p is a vertex from column $j-1$ or j of Q_n. However, since v_p is not adjacent to neither $v_{s,j}$ nor $v_{s,j-1}$ in Q_n, we conclude a contradiction. Hence, $s = 1$, which means that $v_t = v_{1,j-1}$ and $v_{s,j} = v_{1,j}$. If there is $1 < i < n$ such that $v_{i,j} \notin B$, then $v_{i,j}$ distinguishes $v_{1,j-1}$ and $v_{1,j}$. Thus, $\{v_{1,j}, \ldots, v_{n,j}\} \subseteq B$. Consider $v_{2,j}$. Note that $v_{1,j}$ and $v_{2,j}$ are distinguished by $v_{2,j-1}$ and that $v_{2,j-1} \notin B$. So, $v_{2,j} \notin X_t$. Let $1 \leq q \leq k$ be such that $v_{2,j} \in X_q$. Let v_q be the vertex from the n-boundary of A which is contained in X_q. Since v_q is non-adjacent to $v_{2,j}$ due to being in the same supergroup and is adjacent to $v_{n,j-1}$, it follows that $v_q = v_{2,j-1}$. It remains to observe that $v_{2,j-1}$ and $v_{1,j}$ are adjacent in Q_n and that $v_{1,j-1}$ and $v_{2,j}$ are non-adjacent in Q_n, which yields a contradiction. We conclude that $k \leq n$ is not possible, and thus $k \geq n+1$. ■

4.2 Bounds on the size of supergroup trees

In the first subsection, we have established lower bounds on the size of supergroup partitions of particular subgraphs of Q_n. In this subsection, we will show that any supergroup tree for Q_n either produces such a subgraph as intermediate graph or produces a situation that is easy to analyse. Thereby, we will obtain a lower bound on the size of supergroup trees for Q_n, that matches the upper bound provided by Theorem 4.1. As a corollary, we will be able to completely characterise the clique-width of a subclass of proper interval graphs. We partition the lower bound proof into a series of results of special situations.

Let T be a supergroup tree for a graph G. Let a be a node of T, and let $(A_1, \ldots, A_r)$ be the supergroup partition that a is labelled with in T. By M_a^T, we denote the union $A_1 \cup \cdots \cup A_r$. In other words, M_a^T denotes the set of vertices that occur in T_a. Throughout this subsection, the context T will always be clear, so that we will usually omit the superscript and simply write M_a.

Lemma 4.5 *Let $n \geq 3$, and let T be a supergroup tree for Q_n. Assume that T has a node b such that $\{v_{1,n-1}, \ldots, v_{n,n-1}, v_{n,n}\} \subseteq M_b$ and $v_{1,n}, \ldots, v_{n-1,n} \notin M_b$. Then, the size of T is at least $n + 1$.*

Proof. Let a be the lowest node of T above b such that there is $1 \leq p < n$ with $v_{p,n} \in M_a$. Note that b is node in T_a. Let a' and a'' be the two sons of a in T. Without loss of generality, we may assume that b is a node in $T_{a'}$. Then, $v_{p,n} \in M_{a''}$. Let $(A_1, \ldots, A_r)$ be the supergroup partition a is labelled with. Let $1 \leq i, i' \leq r$ be such that $v_{p,n} \in A_i$ and $v_{n,n} \in A_{i'}$. From $v_{p,n}$ being adjacent to $v_{n,n}$, it follows that $i \neq i'$, and due to the supergroup property, the vertices in A_i and $A_{i'}$ are pairwise adjacent in G. Since no vertex from $M_{a'} \setminus \{v_{n,n}\}$ is adjacent to $v_{n,n}$ due to the choice of a, A_i cannot contain any vertex from $M_{a'}$. By the assumptions about M_b and the choice of a, $Q_n[M_{a'}]$ contains at least n maximal groups, and Lemma 3.7 shows that $(A_1, \ldots, A_r)$ contains at least n supergroups with vertices from $M_{a'}$. Since A_i is not among them, $(A_1, \ldots, A_r)$ contains at least $n + 1$ supergroups, which shows $r \geq n + 1$, and thus, the claim follows. ∎

Lemma 4.6 *Let $n \geq 3$, and let T be a supergroup tree for Q_n. Assume that T has an inner node a with b and c its sons such that $M_b = \{v_{1,f}, \ldots, v_{n,f}\}$ for some $1 \leq f \leq n$, there is no full column in M_c and there is no empty column in M_a. Then, the size of T is at least $n + 1$.*

Proof. Let $(A_1, \ldots, A_r)$ be the supergroup partition that a is labelled with in T. For every pair p, p' where $1 \leq p < p' \leq n$, $v_{p,f}$ and $v_{p',f}$ are distinguished by $v_{p,f+1}$ or $v_{p',f-1}$, depending on whether $f \leq n - 1$ or $f \geq 2$. Thus, the vertices from M_b appear in n pairwise different supergroups of $(A_1, \ldots, A_r)$. Assume that $f \leq n - 1$. Let $1 \leq p \leq n$ be smallest such that $v_{p,n} \in M_c$. Note that p exists, since column n is not empty in M_a, and $v_{1,n}, \ldots, v_{p-1,n} \notin M_a$. Let $1 \leq i \leq r$ be such that $v_{p,n} \in A_i$. If A_i contains no vertex from M_b then $r \geq n + 1$. Otherwise, A_i contains vertices from M_b; let $1 \leq q \leq n$ be such that $v_{q,f} \in A_i$. If $|n - f| \geq 2$, i.e., if $f \leq n - 2$, then column n must be full in M_a, which implies that M_c contains a full column and thus contradicts the assumptions. Hence, $f = n - 1$. Since $v_{q,f}$ and $v_{p,n}$ are not adjacent, $q \leq p$. If $q < n$ then $v_{n,n-2}$ distinguishes $v_{q,f}$ and $v_{p,n}$, which yields a contradiction. So, $q = n$, and thus, $p = n$. Due to the choice of p, it follows that $\{v_{1,n-1}, \ldots, v_{n,n-1}, v_{n,n}\} \subseteq M_a$ and $v_{1,n}, \ldots, v_{n-1,n} \notin M_a$, which satisfies the assumptions of Lemma 4.5, and we conclude the claimed lower bound on the size of T.

Now, assume that $f = n$. Then, there is $1 \leq p \leq n$ such that $v_{p,1} \in M_c$. Due to our assumptions about M_a, there is $1 \leq p' \leq n$ such that $v_{p',1} \notin M_a$ and therefore distinguishes $v_{p,1}$ and every vertex from column f. Thus, there is no $1 \leq i \leq r$ such that A_i contains $v_{p,1}$ and a vertex from column f, so that $r \geq n + 1$. ∎

Lemma 4.7 *Let $n \geq 3$, and let T be a supergroup tree for Q_n. Assume that T has an inner node a with b and c its sons such that M_a has a full column, M_a has no empty column, M_b and M_c have no full columns. Then, the size of T is at least $n + 1$.*

Proof. Let $(A_1, \ldots, A_r)$ be the supergroup partition that a is labelled with in T. Denote by U_b the set of vertices from M_b that are "highest" in their column. Formally, $v_{i,j} \in U_b$ if $v_{i,j} \in M_b$ and for every $1 \leq i' < i$, $v_{i',j} \notin M_b$. Analogously, define U_c. We first show that vertices from U_b and U_c cannot appear in the same supergroup. So, for a contradiction, assume that there are $1 \leq i \leq r$ and $v_{p,q} \in U_b$ and $v_{p',q'} \in U_c$ such that $v_{p,q}, v_{p',q'} \in A_i$. Without loss of generality, we may assume $q \leq q'$. Since $v_{p,q}$ and $v_{p',q'}$ must be non-adjacent in Q_n, it directly follows that $q < q'$. Then, the properties of supergroups imply that $\{v_{1,q}, \ldots, v_{p-1,q}\} \subseteq M_b$. The definition of U_b therefore implies $p = 1$. Since $v_{1,q}$ is non-adjacent to every vertex from column q', column q' must be full in M_c due to the properties of vertices in the same supergroup, which contradicts the assumptions about c. Thus, no vertex from U_b appears in the same supergroup as a vertex from U_c.

Next, we show that the vertices from U_b and from U_c appear in pairwise different supergroups. By analogy, it suffices to show the result for U_b. Let $v = v_{p,q}$ and $v' = v_{p',q'}$ be (different) vertices from U_b, where we can assume without loss of generality that $q < q'$. Remember that $q = q'$ is not possible due to the definition of U_b. If $p > 1$ then $v_{1,q} \notin M_b$ and therefore distinguishes v and v'. Analogously, if $p = 1$ and $p' > 1$ then $v_{1,q'} \notin M_b$ and distinguishes v and v'. Assume that $p = p' = 1$. Then, there exists a vertex $w = v_{i,q'}$ such that $w \notin M_b$, since M_b has no full column. Observe that w distinguishes v and v'. We conclude that v and v' are not contained in the same supergroup. It follows that the vertices from $U_b \cup U_c$ appear in pairwise different supergroups, which implies $|U_b| + |U_c| \leq r$. Due to the assumptions about M_a, M_b, M_c, particularly since M_a has a full and no empty column, we conclude that $|U_b| + |U_c| \geq n + 1$. ∎

Lemma 4.8 *Let $n \geq 3$, and let T be a supergroup tree for Q_n. Assume that T has an inner node a with b and c its sons such that M_b has a full column and an empty column and $|M_b| \geq n + 1$. Then, the size of T is at least $n + 1$.*

Proof. As the first case, assume that there are $1 \leq f < e \leq n$ such that column f is full in M_b and column e is empty in M_b and column j is not full for every $1 \leq j < f$ and column j is not empty for every $f < j < e$. For a contradiction, suppose that the size of T is at most n. We apply Lemma 4.3 and directly conclude that $f \leq 2$ and $v_{1,1} \in M_b$ and M_b has the filled row property. Let $(A_1, \ldots, A_r)$ be the supergroup partition that a is labelled with in T. Consider the e-boundary of M_b; it contains n vertices. We apply Lemmas 4.2 and 3.7 and see that the e-boundary vertices of M_b appear in pairwise different supergroups of the partition. Due to the assumption, it follows that every A_i contains exactly one e-boundary vertex of M_b. We show that (M_b, M_c) is a partial $[i, j]$-partition of $V(Q_n)$ for appropriate i, j. Let $1 \leq q \leq n$ be smallest such that there is $1 \leq p \leq n$ with $v_{p,q} \in M_c$; we choose p smallest possible. Remember that M_c is non-empty. Let $1 \leq l \leq r$ be such that $v_{p,q} \in A_l$. If $q = 1$ then $f = 2$, and since $v_{p,q}$ is adjacent to $v_{1,1}$ and no vertex from the e-boundary of M_b is adjacent to $v_{1,1}$, A_l cannot contain an e-boundary vertex of M_b, a contradiction. So, let $q \geq 2$, which means $q > f$. If $q \geq e$ then (M_b, M_c) is a partial $[1, q]$-partition of $V(Q_n)$. This directly follows from the filled row property of M_b. Let $q < e$. Assume that A_l contains an e-boundary vertex $v_{p',q'}$ of M_b. If $q < q'$ then, due to the filled row property of M_b, $v_{p,q'} \notin M_b$, which yields a contradiction to A_l being a supergroup of $Q_n[M_b] \oplus Q_n[M_c]$. So, $q' < q$. If $q' \leq q - 2$ then $\{v_{1,q}, \ldots, v_{n,q}\} \subseteq M_c$, so that column q is empty in M_b. Since $f < q < e$, we obtain a contradiction to the assumption about

e. Thus, $q' = q-1$, and since $v_{p,q}$ and $v_{p',q'}$ are non-adjacent in Q_n, $p' \leq p$ due to the adjacency definitions. It follows that $\{v_{p,q}, \ldots, v_{n,q}, v_{1,q+1}, \ldots, v_{p-1,q+1}\} \subseteq M_c$. Since M_b has the filled row property, we conclude that (M_b, M_c) must be a partial $[p,q]$-partition of $V(Q_n)$.

So, we have seen that (M_b, M_c) is a partial $[i,j]$-partition of $V(Q_n)$, and M_b has a full column and M_c is non-empty. If (M_b, M_c) is a partial $[i,j]$-partition for some $1 < j < n$ then Lemma 4.4 implies $r \geq n+1$, and so the claim follows. Otherwise, (M_b, M_c) is a partial $[1,n]$-partition, and $e = n$. If $\{v_{1,n}, \ldots, v_{n-1,n}\} \cap M_c \neq \emptyset$ then, again, we apply Lemma 4.4 and conclude $r \geq n+1$. Otherwise, $M_c = \{v_{n,n}\}$, which implies that $p = n$ and $q = n$. Since the neighbours of $v_{n,n}$ in Q_n are exactly $v_{1,n}, \ldots, v_{n-1,n}$, $q' = n-1$ and $\{v_{1,n-1}, \ldots, v_{n,n-1}\} \subseteq M_b$. It follows that $\{v_{1,n-1}, \ldots, v_{n,n-1}, v_{n,n}\} \subseteq M_a$ and $v_{1,n}, \ldots, v_{n-1,n} \notin M_a$, and due to Lemma 4.5, the size of T is at least $n+1$. This completes the proof of the first case.

As the second case, assume that for every $1 \leq f \leq n$, if column f is full in M_b then column j is non-empty for every $f < j \leq n$. After performing a 180° rotation to the representation of Q_n as in Figure 2, which mainly results in a re-indexing of the vertices, the conditions of the first case are satisfied, and we conclude the result also for this case. ∎

Now, we are ready to complete the lower bound proof. We particularly show that one of the situations in the already given lemmas must occur in a supergroup tree for Q_n.

Lemma 4.9 *For every $n \geq 3$, $\mathrm{cwd}(Q_n) \geq n+1$.*

Proof. Let $n \geq 3$, and let T be an arbitrary supergroup tree for Q_n. Let F be the set of nodes x of T such that M_x has a full column. Note that F is non-empty, and all nodes in F are inner nodes of T. We call a node in F *minimal* if its sons do not belong to F. We distinguish between two cases for the minimal nodes in F. As the first case, assume that there is a minimal node a in F such that M_a has no empty column. Then, due to minimality, a and its two sons satisfy the conditions of Lemma 4.7, and we conclude that the size of T is at least $n+1$. As the second case, assume the contrary, which means that for every minimal node x in F, M_x has an empty column. Let a be an inner node of T with its sons b and c such that b is a minimal node in F. Then, M_b has a full and an empty column. If we can choose a, b, c so that $|M_b| \geq n+1$ then the three nodes satisfy the conditions of Lemma 4.8, and we conclude that the size of T is at least $n+1$. If a, b, c cannot be chosen to satisfy the conditions of Lemma 4.8 then $|M_x| = n$ for every minimal node in F. So, let a, b, c be an arbitrary choice such that b and c are the sons of a and b is a minimal node from F. Let a' be the parent of a in T, if it exists. If c is a minimal node from F then $|M_a| = |M_b| + |M_c| = 2n$ and M_a has a full and an empty column, and therefore, a' satisfies the conditions of Lemma 4.8. Otherwise, if c is not a minimal node from F, then T_c may or may not contain a node from F. If the former then, due to the assumptions about the choice of c as not being minimal, $|M_c| \geq n+1$, $|M_c|$ contains a full column and, since M_b contains a full column, M_c contains an empty column. Thus, node a satisfies the conditions of Lemma 4.8. If the latter then, if M_a contains no empty column, the nodes a, b, c satisfy the conditions of Lemma 4.6, if M_a contains an empty column, a' satisfies the conditions of Lemma 4.8.

We have shown for every possible case that T has a supergroup partition label of size at least $n+1$, and thus, the claim of the lemma follows by application of Theorem 3.6. ∎

Theorem 4.10 *Let $k \geq 1$, and let G be a k-path power on at least $(k+1)^2$ vertices. Then,* $\text{cwd}(G) = k + 2$.

Proof. Due to Theorem 4.1, $\text{cwd}(G) \leq \text{lcwd}(G) \leq k + 2$. For the lower bound, note that G contains Q_{k+1} as induced subgraph. If $k \geq 2$ then $\text{cwd}(Q_{k+1}) \geq k+2$ due to Lemma 4.9, if $k = 1$ then Q_{k+1} is an induced path of length 3, and thus $\text{cwd}(Q_{k+1}) \geq k+2$ since $\text{cwd}(Q_2) \geq 3$. Then, the monotonicity of clique-width for induced subgraphs shows $\text{cwd}(G) \geq \text{cwd}(Q_{k+1}) \geq k + 2$, which proves the claimed result. ■

5 Final remarks

In this paper, we have shown two main results. First, we have given a purely graph-theoretic characterisation of clique-width by using partition trees. We believe that this provides a new view on clique-width and may lead to interesting theoretic and algorithmic results.

The second main result of this paper is the characterisation of the clique-width of a class of proper interval graphs. Except for the class of square grids, no other graph class of unbounded clique-width is known for which such a characterisation result exists. The main technical results for achieving the characterisation provided lower bounds for particular subgraphs and showed in the proof of Lemma 4.9 that it suffices to consider only such subgraphs. The proof of Lemma 4.9 is also interesting from a broader perspective. All arguments considered only very local properties. It was sufficient to determine a lower bound on the size of supergroup partitions for subgraphs of the type $Q_n[A] \oplus Q_n[B]$, independent of constraints that would be imposed by situations at other nodes of the supergroup tree. A similar property could be observed in the proof of Theorem 4.1. It seems that such a property makes is comparably "easy" to analyse the clique-width. Is there a general scheme behind? Is this true for graphs of specific structural properties?

Theorems 4.1 and 4.10 together show that clique-width and linear clique-width coincide on larger path powers. We obtained this result by explicitly proving the bounds. Can this result be shown also by using only structural arguments?

References

[1] A. Brandstädt, F. Dragan, H.-O. Le, R. Mosca. New Graph Classes of Bounded Clique-Width. *Theory of Computing Systems*, 38:623–645, 2005.

[2] D. G. Corneil, M. Habib, J.-M. Lanlignel, B. A. Reed, U. Rotics. Polynomial time recognition of clique-width ≤ 3 graphs. *LATIN 2000, Springer LNCS*, 1776:126–134.

[3] D. G. Corneil and U. Rotics. On the Relationship between Clique-width and Treewidth. *SIAM Journal on Computing*, 34:825–847, 2005.

[4] B. Courcelle, J. Engelfriet, G. Rozenberg. Handle-rewriting hypergraph grammars. *Journal of Computer and System Sciences*, 46:218–270, 1993.

[5] B. Courcelle, J. A. Makowsky, U. Rotics. Linear time solvable optimization problems on graphs of bounded clique-width. *Theory of Computing Systems*, 33:125–150, 2000.

[6] B. Courcelle and S. Olariu. Upper bounds to the clique width of graphs. *Discrete Applied Mathematics*, 101:77–114, 2000.

[7] W. Espelage, F. Gurski, E. Wanke. Deciding clique-width for graphs of bounded tree-width. *Journal of Graph Algorithms and Applications*, 7:141–180, 2003.

[8] M. R. Fellows, F. A. Rosamond, U. Rotics, S. Szeider. Clique-Width is NP-Complete. *SIAM Journal on Discrete Mathematics*, 23:909–939, 2009.

[9] M. C. Golumbic and U. Rotics. On the Clique-Width of Some Perfect Graph Classes. *International Journal of Foundations of Computer Science*, 11: 423-443, 2000.

[10] F. Gurski. Linear layouts measuring neighbourhoods in graphs. *Discrete Mathematics*, 306:1637–1650, 2006.

[11] P. Heggernes, D. Meister, C. Papadopoulos. Graphs of small bounded linear clique-width. Technical report 362 in Informatics, University of Bergen, 2007.

[12] P. Heggernes, D. Meister, C. Papadopoulos. A Complete Characterisation of the Linear Clique-Width of Path Powers *TAMC 2009*, Springer LNCS, 5532:241–250, 2009.

[13] P. Hlinený, S.-I. Oum, D. Seese, G. Gottlob. Width parameters beyond tree-widthand their applications. *The Computer Journal*, 51:326–362, 2008.

[14] Ö. Johansson. Clique-decomposition, NLC-decomposition, and modular decomposition – relationships and results for random graphs. *Congressus Numerantium*, 132:39–60, 1998.

[15] V. Lozin. From tree-width to clique-width: excluding a unit interval graph. *Proceedings of ISAAC 2008*, Lecture Notes in Computer Science, 5369:872–883, 2008.

[16] V. Lozin and D. Rautenbach. Chordal bipartite graphs of bounded tree- and clique-width. *Discrete Mathematics*, 283:151–158, 2004.

[17] V. Lozin and D. Rautenbach. On the Band-, Tree-, and Clique-Width of Graphs with Bounded Vertex Degree. *SIAM Journal on Discrete Mathematics*, 18:195–206, 2004.

[18] V. Lozin and D. Rautenbach. The relative clique-width of a graph. *Journal of Combinatorial Theory, Series B*, 97:846–858, 2007.

[19] H. Müller and R. Urner. On a disparity between relative cliquewidth and relative NLC-width. *Discrete Applied Mathematics*, 158:828–840, 2010.

[20] S.-il Oum and P. Seymour. Approximating clique-width and branch-width. *Journal of Combinatorial Theory, Series B*, 96:514–528, 2006.