

LINEAR CLIQUE-WIDTH FOR HEREDITARY CLASSES OF COGRAPHS

Robert Brignall*

Department of Mathematics and Statistics
The Open University
Milton Keynes, UK

Nicholas Korpelainen*

Mathematics Department
University of Derby
Derby, UK

Vincent Vatter*[†]

Department of Mathematics
University of Florida
Gainesville, Florida USA

The class of cographs is known to have unbounded linear clique-width. We prove that a hereditary class of cographs has bounded linear clique-width if and only if it does not contain all quasi-threshold graphs or their complements. The proof borrows ideas from the enumeration of permutation classes.

1. INTRODUCTION

A variety of measures of the complexity of graph classes have been introduced and have proved useful for algorithmic problems [3–5, 7, 8, 11, 17, 18, 22, 23]. We are concerned with *clique-width*, introduced by Courcelle, Engelfriet, and Rozenberg [6], and more pertinently, the linear version of this parameter, due to Lozin and Rautenbach [20], and studied further by Gurski and Wanke [14].

The *linear clique-width* of a graph G , $lcw(G)$, is the size of the smallest alphabet Σ such that G can be constructed by a sequence of the following three operations:

- add a new vertex labeled by a letter in Σ ,
- add edges between all vertices labeled i and all vertices labeled j (for $i \neq j$), and
- give all vertices labeled i the label j .

*All three authors were partially supported by EPSRC via the grant EP/J006130/1.

[†]Vatter's research was also partially supported by the National Security Agency under Grant Number H98230-12-1-0207 and the National Science Foundation under Grant Number DMS-1301692. The United States Government is authorized to reproduce and distribute reprints not-withstanding any copyright notation herein.

Date: January 25, 2016

AMS 2000 Subject Classification. 05C78, 68R10, 05C85

Such a sequence of operations is referred to as an *lcw expression* for G . An *efficient* lcw expression for G is one using precisely $\text{lcw}(G)$ labels. The distinction between linear clique-width and clique-width is that clique-width expressions also allow one to take the disjoint union of two previously constructed graphs (along with their labels, which need not be distinct).

The *join* of two vertex-disjoint graphs G and H , denoted $G * H$, is formed from the disjoint union $G \cup H$ by adding all possible edges with one endpoint in G and the other in H . In this context, G and H are referred to as the *join components* of the resulting graph. We consider the following families of graphs in this work. The characterisations we give below can be found in numerous places, for instance in the book *Graph Classes: A Survey* [2] by Brandstädt, Le, and Spinrad.

- A graph G is a *threshold graph* if, starting with K_1 , it can be built by repeatedly taking the disjoint union or join of a threshold graph with K_1 . This class consists of precisely those graphs which avoid P_4 , C_4 , and $2K_2$ as induced subgraphs.
- A graph G is a *quasi-threshold graph* if, starting with K_1 , it can be built by repeatedly taking the disjoint union of two quasi-threshold graphs, or the join of a quasi-threshold graph with K_1 . This class consists of precisely those graphs which avoid P_4 and C_4 as induced subgraphs, and these graphs are also called *trivially perfect*.
- A graph G is a *cograph* if, starting with K_1 , it can be built by repeatedly taking the disjoint union or join of two cographs. This class consists of precisely those graphs which avoid P_4 as an induced subgraph.

Thus,

$$\text{threshold graphs} \subsetneq \text{quasi-threshold graphs} \subsetneq \text{cographs}.$$

For a class of graphs \mathcal{C} , the linear clique-width of \mathcal{C} , $\text{lcw}(\mathcal{C})$, is defined as the maximum value of $\text{lcw}(G)$ over all graphs in \mathcal{C} , if it exists. The class of cographs is precisely the class of graphs with clique-width at most 2 (see [8]), but Gurski and Wanke [14] showed that it has unbounded linear clique-width.¹ On the other hand, threshold graphs have linear clique-width at most 2. (Gurski [13] showed that a graph has linear clique-width at most 2 if and only if it avoids P_4 , $2K_2$, and $\overline{2P_3}$.)

These facts are often cited in the literature (e.g., in [15,16]) to demonstrate that linear clique-width is a significant restriction of clique-width. We sharpen these results by demarcating the precise boundary between subclasses of cographs with bounded and unbounded linear clique-width:

Theorem 1.1. *A hereditary class of cographs has bounded linear clique-width if and only if it contains neither all quasi-threshold graphs nor the complements of all quasi-threshold graphs.*

In other words, Theorem 1.1 shows that the quasi-threshold graphs are a minimal class of unbounded linear clique-width, paralleling Lozin’s constructions of minimal classes of unbounded clique-width [21].

We prove that the class of quasi-threshold graphs has unbounded linear clique-width in the next section. That the class of their complements also has unbounded linear clique-width follows from the following observation, mentioned in Gurski and Wanke [14].

¹In fact, they showed that cographs have unbounded clique-tree-width, a parameter which lies between clique-width and linear clique-width. It should be noted that the quasi-threshold graphs in fact have clique-tree-width 2.

Proposition 1.2. *For every graph G we have $\text{lcw}(G) - 1 \leq \text{lcw}(\overline{G}) \leq \text{lcw}(G) + 1$.*

Proposition 1.2 follows from the fact that $\text{lcw}(\overline{G}) \leq \text{lcw}(G) + 1$. One way to establish this is to note that every graph G has an lcw expression with at most $\text{lcw}(G) + 1$ labels in which every vertex is inserted with a particular label reserved for this purpose (which we will call an *insertion label*), and then immediately connected to all of its neighbors which have already been inserted. Thus by changing these edge operations we can build \overline{G} in the same manner, also with $\text{lcw}(G) + 1$ labels.

Throughout this paper, a set of graphs closed under isomorphism is referred to as a *property*, and a property which is closed downward under the induced subgraph partial order is called *hereditary*. As all properties of graphs considered here are hereditary, we use the term *class* to refer to a hereditary property. We assume that the reader is familiar with basic graph notions and terminology as presented in, for example, Diestel’s text [10].

2. QUASI-THRESHOLD GRAPHS HAVE UNBOUNDED LINEAR CLIQUE-WIDTH

Our proof that the quasi-threshold graphs have unbounded linear clique-width follows from a construction inspired by the work of Gurski and Wanke [14], and requires several preliminary results. The crucial difference between our construction and Gurski and Wanke’s is the notion of a sink-free lcw expression, defined shortly.

First note that we can, without loss of generality, restrict our attention to lcw expressions in which labels can only be introduced by adding a new vertex with that label. We need two further basic observations about lcw expressions. In the following, we say that a label is in *use* (when constructing a graph from an lcw expression) if, at that point in time, there is a vertex with that label.

Proposition 2.1. *In the process of building the graph G via any efficient lcw expression, there is a point in the process when all $\text{lcw}(G)$ labels are in use.*

Proof. Suppose that there is some efficient lcw expression where at most $\text{lcw}(G) - 1$ labels are ever in use simultaneously. Among all such lcw expressions for G , consider those which have shortest length, in terms of the number of operations in the expression. Among these shortest expressions, choose the one in which the last label to be introduced is introduced as late in the expression as possible, and denote this label by ℓ . We achieve a contradiction by modifying the expression so that the first use of ℓ occurs even later. To do this, we follow the lcw expression up until the point when the label ℓ is first due to be used. At this point, there must be another label, say k , which is not currently in use (by our hypotheses). In the rest of the lcw expression, we swap all occurrences of ℓ for k , pushing the first use of ℓ later in the expression (but not increasing the number of operations in the expression), establishing the desired contradiction. \square

Proposition 2.2. *Fix an efficient lcw expression for the graph G and a label ℓ used in this expression. Then there is an lcw expression for G in which the vertices are inserted in the same order and where vertices labeled ℓ are never relabeled.*

Proof. Suppose that the given lcw expression does not preserve ℓ , and identify the first instance when vertices labeled ℓ are relabeled by some other label k . Instead, replace this operation by relabeling vertices labeled k to ℓ , and then swap all subsequent instances of ℓ and k . Finally, repeat this process until the desired expression is created. \square

We say that a label in an lcw expression is a *sink label* if it is never used to create edges nor ever relabeled. In other words, it is a label that can only be given to vertices whose neighborhoods have been completed. We further say that an lcw expression is *sink-free* if no label of this expression is a sink label.

Proposition 2.3. *Suppose that every efficient lcw expression for the graph G is sink-free. Then the graph $H = G \cup G$ satisfies $\text{lcw}(H) = \text{lcw}(G) + 1$ and moreover, H has an efficient lcw expression with a sink.*

Proof. First we prove that $\text{lcw}(G) + 1$ is a lower bound on $\text{lcw}(H)$. Suppose to the contrary that H has an lcw expression with $\text{lcw}(G)$ labels. Denote the vertex sets of the two copies of G in H by disjoint sets X and Y . By symmetry we may assume that the first vertex which is inserted lies in X . Denote this vertex by x and let ℓ denote the label of x when it is inserted. By Proposition 2.2, there is an lcw expression for H using $\text{lcw}(G)$ labels in which x is still the first vertex inserted and in which vertices labeled ℓ (specifically, the vertex x) are never relabeled.

Since the restriction of this lcw expression to Y induces an lcw expression for G , there is at least one point during the construction where every label is used to label some vertex in Y , by Proposition 2.1. Since x is labeled ℓ at this point of the construction, however, no edges may ever be added between a vertex of Y labeled ℓ and any other vertex of Y , as this would also create an edge between x and a vertex of Y . Therefore, in the restriction of this lcw expression to the vertices of Y , ℓ is a sink label, contradicting our hypotheses.

It is much easier to show that $\text{lcw}(H) \leq \text{lcw}(G) + 1$: use any lcw expression of G with $\text{lcw}(G)$ labels to create a copy of G , replace all of these labels by a new label, and then repeat the lcw expression of G to create a disjoint copy of G . Note that the lcw expression we have built for H has a sink label, namely the new label we introduced. \square

Proposition 2.4. *Suppose that the graph G is not edgeless and has an efficient lcw expression with a sink. The graph $H = K_1 * (G \cup G)$ satisfies $\text{lcw}(H) = \text{lcw}(G)$ and moreover, every efficient lcw expression for H is sink-free.*

Proof. First we show that $\text{lcw}(H) = \text{lcw}(G)$. We have $\text{lcw}(H) \geq \text{lcw}(G)$ because linear clique-width is hereditary, so we need only show that H has an lcw expression with $\text{lcw}(G)$ labels. We follow an efficient lcw expression with sink ℓ to build one copy of G , before relabeling all vertices of this copy with the label ℓ . We then follow the same lcw expression to build a disjoint copy of G , and again relabel all vertices with ℓ . Since G has an edge, $\text{lcw}(G) \geq 2$, and thus G has at least one non- ℓ label. Insert a final vertex of this label and add edges between it and every vertex constructed before.

Now suppose for the sake of contradiction that H has an efficient lcw expression with sink label ℓ . Denote the vertices of the two copies of G in H by disjoint sets X and Y and let z denote the vertex of H which is adjacent to all of $X \cup Y$. Using Proposition 2.1, suppose that when this lcw expression is restricted to X (resp., Y), the insertion of the vertex x (resp., y) marks the first point at which all labels are in use. By symmetry, we may assume that x is inserted before y . Thus when x is inserted, there is a vertex $x' \in X$ labeled by ℓ . Since we assumed that ℓ is a sink label in the expression for H , x' can never acquire another edge. Therefore z must have been inserted before x' , and hence also before x . At the point x is inserted, there is a vertex $x'' \in X$ with the same label as z . Since x'' and z will always now have the same label, and x'' is inserted before y , it is now impossible to add the edge between y and z without adding an edge between y and x'' , a contradiction. \square

Although not necessary for our upcoming construction, note that for G not satisfying the hypotheses of Proposition 2.4, it is still possible to compute the linear clique-width of $H = K_1 * (G \cup G)$. If G does not have an edge, then $H = K_1 * \bar{K}_n$ for some n so $\text{lcw}(H) = 2$. In the other case, if all efficient lcw expressions for G are sink-free, then Proposition 2.3 shows that $\text{lcw}(H) \geq \text{lcw}(G \cup G) = \text{lcw}(G) + 1$. In fact, $\text{lcw}(H) = \text{lcw}(G) + 1$, as can be seen by following the lcw expression given in the proof of Proposition 2.3 to build $G \cup G$, then relabeling every vertex with the same label, inserting a new vertex with a different label, and adding all edges between this vertex and the rest of the graph.

We now have the necessary tools to prove one half of Theorem 1.1.

Corollary 2.5. *The class of quasi-threshold graphs has unbounded linear clique-width.*

Proof. Set $G_1 = K_2$, and for $k \geq 1$ define G_{k+1} by

$$G_{k+1} = K_1 * ((G_k \cup G_k) \cup (G_k \cup G_k)).$$

Clearly these are all quasi-threshold graphs. By inductively applying Proposition 2.3 to G_k , and then inserting the outcome for $G_k \cup G_k$ into Proposition 2.4, we conclude that $\text{lcw}(G_k) = k + 1$ for all $k \geq 1$, proving the result. \square

In the proof above, note that we took four copies of G_k to form G_{k+1} merely for convenience; there may well be a construction to prove Corollary 2.5 which adds fewer vertices at each step.

As already observed, Proposition 1.2 implies that the class of complements of quasi-threshold graphs also has unbounded linear clique-width.

3. STRUCTURAL NOTIONS

Our proof of the other half of Theorem 1.1 requires the notions of the substitution (or modular) decomposition, well-quasi-ordering, and atomicity. Here we briefly review these notions and some other facts required in our proof.

Let G be a graph on the labeled vertices $\{1, \dots, n\}$ and (H_1, \dots, H_n) a sequence of graphs. The *inflation* of G by H_1, \dots, H_n is the graph on the disjoint union $H_1 \cup \dots \cup H_n$ with additional edges added between all vertices of H_i and all vertices of H_j if and only if i and j are adjacent in G . We denote this graph by $G[H_1, \dots, H_n]$.

Given classes \mathcal{C} and \mathcal{D} , the inflation of \mathcal{C} by \mathcal{D} , which we denote $\mathcal{C}[\mathcal{D}]$, is the set of all inflations $G[H_1, \dots, H_n]$ where G is a labeling of a graph from \mathcal{C} and $H_1, \dots, H_n \in \mathcal{D}$. The relationship between the substitution decomposition and the (not necessarily linear) clique-width parameter was thoroughly studied by Courcelle and Olariu [8], who established that $\text{cw}(\mathcal{C}[\mathcal{D}]) = \max\{\text{cw}(\mathcal{C}), \text{cw}(\mathcal{D})\}$. For linear clique-width, we observe the following.

Proposition 3.1. *For classes \mathcal{C} and \mathcal{D} of graphs with bounded linear clique-width, $\text{lcw}(\mathcal{C}[\mathcal{D}]) \leq \text{lcw}(\mathcal{C}) + \text{lcw}(\mathcal{D})$.*

Proof. Take any graph $G \in \mathcal{C}$ with n vertices labeled by $\{1, \dots, n\}$, and any sequence (H_1, \dots, H_n) of graphs in \mathcal{D} . We want to find an lcw expression for $G[H_1, \dots, H_n]$ with at most $\text{lcw}(G) +$

$\max\{\text{lcw}(H_i)\}$ labels. We build this expression by following an efficient lcw expression for G . Whenever this expression instructs us to insert the vertex $v_i \in G$, we instead use a disjoint set of $\text{lcw}(H_i)$ labels to create a copy of H_i . Once this copy of H_i is completed, we relabel all of its vertices to the label given to v_i in the lcw expression of G and continue. \square

Letting $\overline{\mathcal{K}}$ denote the class of all edgeless graphs, note that $\text{lcw}(\mathcal{C}[\overline{\mathcal{K}}]) = \text{lcw}(\mathcal{C})$; indeed, it is known (see [16]) that graphs of linear clique-width at most 2 are precisely the inflations of threshold graphs by $\overline{\mathcal{K}}$ (as mentioned in the introduction, these are also the graphs which avoid P_4 , $2K_2$, and $\overline{2P_3}$).

A class of finite graphs is said to be *well-quasi-ordered* (*wqo*) under the induced subgraph partial order if given any infinite sequence G_1, G_2, \dots of graphs in the class, we can find indices $i < j$ such that G_i is an induced subgraph of G_j . The folklore result below (see Kruskal [19]) gives several equivalent conditions. In the statement of this result we say that the subclasses of the class \mathcal{C} satisfy the *descending chain condition* if every infinite descending chain of subclasses

$$\mathcal{C} \supseteq \mathcal{D}_1 \supseteq \mathcal{D}_2 \supseteq \dots$$

contains a minimum element, that is, in any such sequence there is an index i such that $\mathcal{D}_j = \mathcal{D}_i$ for all $j \geq i$.

Theorem 3.2 (folklore). *Let \mathcal{C} be a class of finite graphs. The following are equivalent:*

- (a) \mathcal{C} is wqo under the induced subgraph order,
- (b) \mathcal{C} does not contain an infinite antichain,
- (c) given any infinite sequence G_1, G_2, \dots of graphs of \mathcal{C} , there is an infinite ascending subsequence, i.e., a sequence G_{i_1}, G_{i_2}, \dots such that each graph in the sequence is an induced subgraph of the next graph in the subsequence,
- (d) the subclasses of \mathcal{C} satisfy the descending chain condition.

Cographs are known to be wqo (see Damaschke [9]), and we make use of both parts (c) and (d) of Theorem 3.2 in our proof.

Note that there is no general relation between linear clique-width and well-quasi-ordering: in one direction, cographs are wqo but have unbounded linear clique-width, while the set of all cycles forms an infinite antichain in the induced subgraph partial order, but each cycle has linear clique-width at most 4.

Our final concept is atomicity. We say that the class \mathcal{C} is *atomic* if it cannot be written as the union of two proper subclasses. This condition is equivalent to the *joint embedding property*: for every pair of graphs in \mathcal{C} there is another graph in \mathcal{C} which contains both. (Fraïssé [12] seems to have been the first to study these notions.) We will need to make use of the combination of atomicity and well-quasi-order via the proposition, which appears to be folklore. As it is relatively unknown, for completeness we give a proof.

Proposition 3.3. *Every wqo class of graphs is a finite union of atomic classes.*

Proof. Suppose that \mathcal{C} is a wqo class. We recursively build a binary tree whose root is labeled by \mathcal{C} . Consider a node labeled by \mathcal{D} . If $\mathcal{D} = \mathcal{D}_1 \cup \mathcal{D}_2$ for proper subclasses \mathcal{D}_1 and \mathcal{D}_2 then we add

children labeled by \mathcal{D}_1 and \mathcal{D}_2 to this node. Otherwise \mathcal{D} is atomic and nodes labeled \mathcal{D} in the binary tree are leaves. Because \mathcal{C} is wqo it does not have an infinite descending chain of subclasses by Theorem 3.2 (d), so this tree contains no infinite paths and thus is finite by König's Lemma. Its leaves give the desired atomic classes. \square

4. THE OTHER HALF OF THEOREM 1.1

Having shown that the quasi-threshold graphs have unbounded linear clique-width, this section concerns the proof of the other direction of Theorem 1.1. Our proof, which is inspired by the work of Albert, Atkinson, and Vatter [1] on permutation classes, begins by proving that every infinite subclass \mathcal{C} of cographs either contains all quasi-threshold graphs or their complements, or is contained in the inflation of threshold graphs by a proper subclass $\mathcal{D} \subsetneq \mathcal{C}$.

If \mathcal{C} were closed under both disjoint unions and joins then it would contain all cographs. Thus \mathcal{C} is either closed under one of these operations or under neither of them. If it is closed under disjoint unions or joins, the following lemma gives what we need. Note that both the class of edgeless graphs and the class of complete graphs are subclasses of threshold graphs.

Lemma 4.1. *Let \mathcal{C} be a nonempty subclass of cographs that is closed under either disjoint unions or joins. Either \mathcal{C} is contained in the inflation of the edgeless or complete graphs by a proper subclass $\mathcal{D} \subsetneq \mathcal{C}$, or \mathcal{C} contains all quasi-threshold graphs or all of their complements.*

Proof. Pick a subclass \mathcal{C} of cographs, which by symmetry we may assume to be closed under disjoint unions. Let \mathcal{C}_c denote the downward closure of the connected graphs in \mathcal{C} . If \mathcal{C}_c is a proper subclass of \mathcal{C} then $\mathcal{C} \subseteq \overline{\mathcal{K}}[\mathcal{C}_c]$, where $\overline{\mathcal{K}}$ denotes the class of edgeless graphs (all of which are threshold graphs).

If $\mathcal{C}_c = \mathcal{C}$, then every graph in \mathcal{C} is contained in a connected graph in \mathcal{C} . In this case, we prove by induction on $|V(G)|$ that every quasi-threshold graph G is contained in \mathcal{C} . The base case $|V(G)| = 1$ is trivial, as \mathcal{C} is nonempty. Now suppose that $|V(G)| \geq 2$. If G is a disjoint union then we are done because \mathcal{C} is closed under disjoint unions. Otherwise, $G = H * K_1$ for a graph H which lies in \mathcal{C} by induction. Again because \mathcal{C} is closed under disjoint unions, $H \cup K_1 \in \mathcal{C}$, and because $\mathcal{C} = \mathcal{C}_c$, $H \cup K_1$ is contained as an induced subgraph in a connected graph in \mathcal{C} . Finally, because every connected cograph is a join, we see that \mathcal{C} contains $(H \cup K_1) * K_1$, and thus also G , as desired. \square

The remaining case is that where \mathcal{C} is closed under neither disjoint unions nor joins. Our proof of Theorem 1.1 will seek a minimal counterexample, and since cographs are wqo, we can appeal to Theorem 3.2 (d) to establish that a minimal counterexample must be atomic. Consequently, we can restrict the following lemma to atomic subclasses of cographs.

Lemma 4.2. *Let \mathcal{C} be an infinite atomic subclass of cographs. If \mathcal{C} is closed under neither disjoint unions nor joins then it is contained in the inflation of the threshold graphs by a proper subclass $\mathcal{D} \subsetneq \mathcal{C}$.*

Proof. Let \mathcal{C} be an atomic subclass of cographs which is closed under neither disjoint unions nor joins. For every graph $G \in \mathcal{C}$, let \mathcal{U}_G^+ denote the set

$$\mathcal{U}_G^+ = \{H \in \mathcal{C} : G \cup H \in \mathcal{C}\}$$

of those graphs whose disjoint unions with G lie in \mathcal{C} . We further define the set

$$\begin{aligned} \mathcal{U}_G^- &= \{H \in \mathcal{C} : \text{for every } H' \in \mathcal{C}, \text{ if } G \cup H' \in \mathcal{C} \text{ then } H \cup H' \in \mathcal{C}\}, \\ &= \{H \in \mathcal{C} : H \cup H' \in \mathcal{C} \text{ for all } H' \in \mathcal{U}_G^+\} \end{aligned}$$

of all graphs which are “at least as unionable” as G .

First note that \mathcal{U}_G^+ and \mathcal{U}_G^- are classes. Moreover, if for some graph $G \in \mathcal{C}$ we were to have $\mathcal{U}_G^+ = \mathcal{U}_G^- = \mathcal{C}$, then that would mean that for every graph $H \in \mathcal{U}_G^- = \mathcal{C}$ and every graph $H' \in \mathcal{U}_G^+ = \mathcal{C}$ their disjoint union $H \cup H'$ lies in \mathcal{C} , contradicting our assumption that \mathcal{C} is not closed under disjoint unions. Thus for every graph $G \in \mathcal{C}$, we define the proper subclass

$$\mathcal{U}_G = \begin{cases} \mathcal{U}_G^+ & \text{if } \mathcal{U}_G^+ \neq \mathcal{C}, \text{ or} \\ \mathcal{U}_G^- & \text{otherwise (in which case } \mathcal{U}_G^- \neq \mathcal{C} \text{ by the above).} \end{cases}$$

Observe that passing from G to \mathcal{U}_G reverses inclusions for \mathcal{U}_G^+ , but not for \mathcal{U}_G^- : i.e., if H is an induced subgraph of G then $\mathcal{U}_G^+ \subseteq \mathcal{U}_H^+$, but $\mathcal{U}_H^- \subseteq \mathcal{U}_G^-$. Note also that if $\mathcal{U}_G^+ = \mathcal{U}_H^+$, then we also have $\mathcal{U}_G^- = \mathcal{U}_H^-$.

We first claim that the number of *distinct* classes of the form \mathcal{U}_G^+ is finite. Suppose to the contrary that there are infinitely many distinct classes $\mathcal{U}_{G_1}^+, \mathcal{U}_{G_2}^+, \dots$. By Theorem 3.2 (c), the sequence G_1, G_2, \dots contains an infinite ascending sequence G_{i_1}, G_{i_2}, \dots . Our observation above then shows that

$$\mathcal{U}_{G_{i_1}}^+ \supsetneq \mathcal{U}_{G_{i_2}}^+ \supsetneq \dots$$

This, however, contradicts the descending chain condition (Theorem 3.2 (d)). Thus there are only finitely many distinct classes \mathcal{U}_G^+ , and from this we can conclude that there are also only finitely many distinct classes \mathcal{U}_G^- (since there is at most one such class for every class \mathcal{U}_G^+). Thus we have established that there are only finitely many distinct classes \mathcal{U}_G , so there is a finite set $S \subseteq \mathcal{C}$ such that every class \mathcal{U}_G (with $G \in \mathcal{C}$) is of the form \mathcal{U}_H for some $H \in S$.

Define

$$\mathcal{U} = \bigcup_{H \in S} \mathcal{U}_H.$$

As each \mathcal{U}_G is a proper subclass of \mathcal{C} and \mathcal{C} is atomic, we know that \mathcal{U} is also a proper subclass of \mathcal{C} . We claim that every disconnected graph in \mathcal{C} has a connected component contained in \mathcal{U} . To see this, consider a disconnected graph $A \cup B \in \mathcal{C}$. By definition, $A \in \mathcal{U}_B^+$ and $B \in \mathcal{U}_A^+$, so we are done unless both \mathcal{U}_A^+ and \mathcal{U}_B^+ are equal to \mathcal{C} . In this case we see that $\mathcal{U}_A = \mathcal{U}_A^-$ and the disjoint union of B with every graph in \mathcal{C} lies in \mathcal{C} , so $B \in \mathcal{U}_A$, as desired.

Recall that every cograph is either a disjoint union or a join. Thus by an analogous argument (or by applying the above argument to the complement class $\overline{\mathcal{C}}$), we see that there is a proper subclass $\mathcal{J} \subsetneq \mathcal{C}$ such that every connected graph in \mathcal{C} contains a join component in \mathcal{J} . Moreover, because \mathcal{C} is atomic, we see that $\mathcal{U} \cup \mathcal{J} \subsetneq \mathcal{C}$.

It is now easy to establish by induction on $|V(G)|$ that every graph $G \in \mathcal{C}$ is contained in the inflation of a threshold graph by graphs in $\mathcal{U} \cup \mathcal{J}$. This fact is clear for $|V(G)| = 1$ because K_1 must be contained in $\mathcal{U} \cup \mathcal{J}$, and all larger graphs have either a component or a join component in $\mathcal{U} \cup \mathcal{J}$. \square

We conclude with a proof of our main theorem.

Proof of Theorem 1.1. Suppose that the theorem was false and choose a minimal counterexample, say \mathcal{C} , by Theorem 3.2 (d). Note that \mathcal{C} must be an infinite class, it cannot contain all quasi-threshold graphs or their complements, and (by the comments before Lemma 4.2) \mathcal{C} is atomic.

\mathcal{C} cannot be closed under both disjoint unions and joins, else it contains all cographs. In this case, \mathcal{C} must be contained in the inflation of the threshold graphs by a proper subclass $\mathcal{D} \subsetneq \mathcal{C}$: this follows by Lemma 4.1 if \mathcal{C} is closed under disjoint unions or joins (since \mathcal{C} cannot contain all quasi-threshold graphs or their complements), and by Lemma 4.2 if not.

Finally, Proposition 3.1 shows that $\text{lcw}(\mathcal{C}) \leq \text{lcw}(\mathcal{D}) + 2$, which is finite by our choice of \mathcal{C} . \square

Acknowledgements. We are grateful to the anonymous referees, whose careful and detailed reports greatly improved the presentation of this paper, and also to Michael Albert, Jay Pantone, Sang-il Oum, Jisu Jeong and Hojin Choi for their conversations and corrections.

REFERENCES

- [1] ALBERT, M. H., ATKINSON, M. D., AND VATTER, V. Subclasses of the separable permutations. *Bull. Lond. Math. Soc.* 43 (2011), 859–870.
- [2] BRANDSTÄDT, A., LE, V. B., AND SPINRAD, J. P. *Graph classes: a survey*. SIAM Monographs on Discrete Mathematics and Applications. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1999.
- [3] BUI-XUAN, B.-M., TELLE, J. A., AND VATSHELLE, M. Boolean-width of graphs. *Theoret. Comput. Sci.* 412, 39 (2011), 5187–5204.
- [4] CORNEIL, D. G., AND ROTICS, U. On the relationship between clique-width and treewidth. *SIAM J. Comput.* 34, 4 (2005), 825–847.
- [5] COURCELLE, B. The expression of graph properties and graph transformations in monadic second-order logic. In *Handbook of graph grammars and computing by graph transformation, Vol. 1*. World Sci. Publ., River Edge, NJ, 1997, pp. 313–400.
- [6] COURCELLE, B., ENGELFRIET, J., AND ROZENBERG, G. Handle-rewriting hypergraph grammars. *J. Comput. System Sci.* 46, 2 (1993), 218–270.
- [7] COURCELLE, B., MAKOWSKY, J. A., AND ROTICS, U. Linear time solvable optimization problems on graphs of bounded clique-width. *Theory Comput. Syst.* 33, 2 (2000), 125–150.
- [8] COURCELLE, B., AND OLARIU, S. Upper bounds to the clique width of graphs. *Discrete Appl. Math.* 101, 1-3 (2000), 77–114.
- [9] DAMASCHKE, P. Induced subgraphs and well-quasi-ordering. *J. Graph Theory* 14, 4 (1990), 427–435.
- [10] DIESTEL, R. *Graph theory*, third ed., vol. 173 of *Graduate Texts in Mathematics*. Springer-Verlag, Berlin, 2005.

- [11] ESPELAGE, W., GURSKI, F., AND WANKE, E. How to solve NP-hard graph problems on clique-width bounded graphs in polynomial time. In *Graph-theoretic concepts in computer science (Boltenhagen, 2001)*, vol. 2204 of *Lecture Notes in Comput. Sci.* Springer, Berlin, 2001, pp. 117–128.
- [12] FRAÏSSÉ, R. Sur l'extension aux relations de quelques propriétés des ordres. *Ann. Sci. Ecole Norm. Sup. (3)* 71 (1954), 363–388.
- [13] GURSKI, F. Characterizations for co-graphs defined by restricted NLC-width or clique-width operations. *Discrete Math.* 306, 2 (2006), 271–277.
- [14] GURSKI, F., AND WANKE, E. On the relationship between NLC-width and linear NLC-width. *Theoret. Comput. Sci.* 347, 1-2 (2005), 76–89.
- [15] HEGGERNES, P., MEISTER, D., AND PAPADOPOULOS, C. A complete characterisation of the linear clique-width of path powers. In *Theory and applications of models of computation*, vol. 5532 of *Lecture Notes in Comput. Sci.* Springer, Berlin, 2009, pp. 241–250.
- [16] HEGGERNES, P., MEISTER, D., AND PAPADOPOULOS, C. Graphs of linear clique-width at most 3. *Theoret. Comput. Sci.* 412, 39 (2011), 5466–5486.
- [17] KAMIŃSKI, M., LOZIN, V. V., AND MILANIČ, M. Recent developments on graphs of bounded clique-width. *Discrete Appl. Math.* 157, 12 (2009), 2747–2761.
- [18] KOBLER, D., AND ROTICS, U. Edge dominating set and colorings on graphs with fixed clique-width. *Discrete Appl. Math.* 126, 2-3 (2003), 197–221.
- [19] KRUSKAL, J. B. The theory of well-quasi-ordering: A frequently discovered concept. *J. Combin. Theory Ser. A* 13 (1972), 297–305.
- [20] LOZIN, V., AND RAUTENBACH, D. The relative clique-width of a graph. Tech. Rep. 16, RUTCOR Research Reports, http://rutcor.rutgers.edu/pub/rrr/reports2004/16_2004.ps, May 2004.
- [21] LOZIN, V. V. Minimal classes of graphs of unbounded clique-width. *Ann. Comb.* 15, 4 (2011), 707–722.
- [22] OUM, S.-I., AND SEYMOUR, P. Approximating clique-width and branch-width. *J. Combin. Theory Ser. B* 96, 4 (2006), 514–528.
- [23] WANKE, E. k -NLC graphs and polynomial algorithms. *Discrete Appl. Math.* 54, 2-3 (1994), 251–266. Efficient algorithms and partial k -trees.