Exploiting restricted linear structure to cope with the hardness of clique-width

Pinar Heggernes* Daniel Meister† Udi Rotics‡

Abstract

Clique-width is an important graph parameter whose computation is NP-hard. In fact we do not know of any other algorithm than brute force for the exact computation of clique-width on any non-trivial graph class. Results so far indicate that proper interval graphs constitute the first interesting graph class on which we might have hope to compute clique-width, or at least its linear variant linear clique-width, in polynomial time. In TAMC 2009, a polynomial-time algorithm for computing linear clique-width on a subclass of proper interval graphs was given. In this paper, we present a polynomial-time algorithm for a larger subclass of proper interval graphs that approximates the clique-width within an *additive* factor 3. Previously known upper bounds on clique-width result in arbitrarily large difference from the actual clique-width when applied on this class. Our results contribute toward the goal of eventually obtaining a polynomial-time exact algorithm for clique-width on proper interval graphs.

1 Introduction

Clique-width is a graph parameter that has many algorithmic applications [4]. NP-hard problems that are expressible in a certain type of monadic second-order logic admit algorithms with running time $f(k) \cdot n$ on input graphs of clique-width k with n vertices, where function f depends only on k [5]. Unfortunately it is NP-hard to compute the clique-width of a given graph even for cobipartite graphs [8]. This hardness result is true even for the linear variant of clique-width, linear clique-width, which gives an upper bound on clique-width. 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.

Clique-width has received a lot of attention recently [1, 3, 7, 8, 13, 14, 15, 16, 17]. Nevertheless, positive results known on the computation of clique-width so far are very restricted. There exist efficient algorithms that, for fixed integer k, decide whether the clique-width of a given graph is more than k or bounded above by some exponential function in k [12, 18, 19]. 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]. Graphs of bounded treewidth have also bounded clique-width, and their clique-width can be computed in polynomial time [7]. 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].

Proper interval graphs have unbounded clique-width and the best approximation of their clique-width so far is maximum clique size plus 1, which can be arbitrarily larger than the actual clique-width. Still this is the most promising class of graphs of unbounded clique-width

*Department of Informatics, University of Bergen, Norway. Email: `pinar.heggernes@ii.uib.no`
†Theoretical Computer Science, RWTH Aachen University, Germany. Email: `meister@cs.rwth-aachen.de`
‡Netanya Academic College, Netanya, Israel. Email: `rotics@netanya.ac.il`

with respect to whether or not we will be able to obtain a polynomial-time algorithm for the exact computation of their clique-width. There are additive approximation algorithms for grids and for very regular-structured subclasses of proper interval graphs and permutation graphs [9]. The first polynomial-time algorithm to compute linear clique-width on a graph class of unbounded clique-width was given by Heggernes et al. in a TAMC 2009 paper for path powers, which form a subclass of proper interval graphs [10].

In this paper, we continue this line of research and attack the hardness of clique-width by exploiting the linear structure of proper interval graphs. This time we study a significantly larger subclass of proper interval graphs than path powers. We give a polynomial-time approximation algorithm for computing the clique-width and linear clique-width of these graphs within an additive factor of at most 3. Previously known upper bound results do not give an additive approximation when applied to this graph class. Furthermore, the studied graphs constitute the largest graph class on which an algorithm for computing or additively approximating the clique-width is given. The main difference between previously considered graph classes and the graph class that we study is that our graphs have a much more irregular structure.

Two proofs are omitted due to space restrictions and they can be found in the appendix.

2 Definitions, notation and clique-width

We consider simple finite undirected graphs. For a graph $G = (V, E)$, $V = V(G)$ denotes the *vertex set* of G and $E = E(G)$ denotes the *edge set* of G. Edges of G are denoted as uv, which means that u and v are *adjacent* in G. For a vertex u of G, $N_G(u)$ denotes the *neighbourhood* of u in G, which is the set of vertices of G that are adjacent to u. For a vertex pair u, v of G, u is a *true twin* of v if $N_G(u) \cup \{u\} = N_G(v) \cup \{v\}$. Note that a vertex can have several true twins. A graph H is a *subgraph* of G if $V(H) \subseteq V(G)$ and $E(H) \subseteq E(G)$. For a set $S \subseteq V(G)$, the subgraph of G *induced* by S, denoted as $G[S]$, has vertex set S and contains exactly the edges uv of G where $u, v \in S$. For a vertex x of G, $G-x$ denotes that subgraph of G that is induced by $V(G) \setminus \{x\}$. We also say that $G-x$ is obtained from G by *deleting* x. The *disjoint union* of two graphs G and H is $(V(G) \cup V(H), E(G) \cup E(H))$. When applying the disjoint union operation, we always assume that the two involved graphs have disjoint vertex sets.

Clique-width is defined based on a set of operations. Let $k \geq 1$ be an integer. A *k-labelled graph*, *k-graph* for short, is a graph each of whose vertices is labelled with an integer from the set $\{1, \ldots, k\}$. We define four sets of operations on k-labelled graphs:

- $i(u)$ creates a k-graph on vertex set $\{u\}$ where $i \in \{1, \ldots, k\}$ and u has label i
- $\eta_{i,j}(G)$ adds edges between all vertices with label i and all vertices with label j of G where G is a k-graph, $i, j \in \{1, \ldots, k\}$ and $i \neq j$
- $\rho_{i \to j}(G)$ changes all labels i into label j in G where G is a k-graph and $i, j \in \{1, \ldots, k\}$
- $G \oplus H$ is the disjoint union of G and H where G and H are k-graphs.

A *k-expression* is a properly formed expression using the four types of operations. We say that a graph G *has* a k-expression if there exists a k-expression α such that G is equal to the graph defined by α without the labels. The *clique-width* of a graph G, denoted by $\mathrm{cwd}(G)$, is the smallest integer k such that G has a k-expression. An example for a 3-expression for an induced path on five vertices, $(a_1, a_2, a_3, a_4, a_5)$, is this:

$$\rho_{3\to 1}\left(\eta_{2,3}\left(3(a_3) \oplus \Big((\eta_{1,2}(1(a_1) \oplus 2(a_2))) \oplus (\eta_{1,2}(1(a_5) \oplus 2(a_4)))\Big)\right)\right).$$

Linear clique-width is a restriction of clique-width that allows the disjoint union to operate on at most one labelled graph with at least two vertices. An expression that respects the restriction on

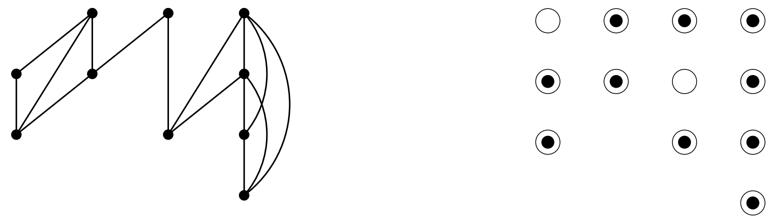


Figure 1: On the left hand side, find a proper interval graph, and on the right hand side, find a bubble model for the depicted graph. The bubble model contains two empty and ten non-empty bubbles.

the disjoint union operation is called a *linear expression*. The *linear clique-width* of a graph G, denoted by $\mathrm{lcwd}(G)$, is the smallest integer k such that G has a linear k-expression. Since linear k-expressions are k-expressions, it immediately follows that $\mathrm{cwd}(G) \leq \mathrm{lcwd}(G)$.

3 Proper interval graphs and the bubble model

A graph G is called *proper interval graph* if each vertex of G can be assigned a closed interval of the real line such that the following two conditions are satisfied: (1) no interval is properly contained in another, and (2) two vertices of G are adjacent if and only if the corresponding assigned intervals have a non-empty intersection. The class of proper interval graphs is equal to indifference graphs [20].

The *clique number* of a graph G, denoted as $\omega(G)$, is the largest number of vertices in a clique of G. The *pathwidth* of G is the smallest clique number of an interval graph that contains G as a subgraph. The linear clique-width of a graph G is bounded from above by its pathwidth $\mathrm{pw}(G)$; precisely, $\mathrm{lcwd}(G) \leq \mathrm{pw}(G) + 2$ [8]. Since proper interval graphs are interval graphs, it holds for every proper interval graph G that $\mathrm{pw}(G) = \omega - 1$. Together with the general pathwidth upper bound on the linear clique-width, we obtain for every proper interval graph G that $\mathrm{cwd}(G) \leq \mathrm{lcwd}(G) \leq \omega(G) + 1$ [8]. Note that this upper bound can be much larger than the actual linear clique-width. A simple example is a complete graph, that has clique-width and linear clique-width 2 and clique number equal to n.

For the study of the clique-width on proper interval graphs, we use a 2-dimensional model for proper interval graphs. Let G be a graph. A *bubble model* for G is a 2-dimensional structure $\mathcal{B} = \langle B_{i,j} \rangle_{1 \leq j \leq s, 1 \leq i \leq r_j}$ such that the following conditions are satisfied:

- for $1 \leq j \leq k$ and $1 \leq i \leq r_j$, $B_{i,j}$ is a (possibly empty) set of vertices of G
- the sets $B_{1,1}, \ldots, B_{r_k,k}$ are pairwise disjoint and cover $V(G)$
- two vertices u, v of G are adjacent if and only if there are $1 \leq j \leq j' \leq s$ and $1 \leq i \leq r_j$ and $1 \leq i' \leq r_{j'}$ such that $u, v \in B_{i,j} \cup B_{i',j'}$ and (a) $j = j'$ or (b) $j + 1 = j'$ and $i > i'$.

A graph is a proper interval graph if and only if it has a bubble model [10]. Let G be a proper interval graph with bubble model $\mathcal{B} = \langle B_{i,j} \rangle_{1 \leq j \leq s, 1 \leq i \leq r_j}$. A *column* of $\mathcal{B}$ is the collection of all bubbles $B_{i,j}$ for a fixed $1 \leq j \leq s$. The *jth column* of $\mathcal{B}$ is the collection of $B_{1,j}, \ldots, B_{r_j,j}$ and is denoted as $\mathcal{B}_j$. We assume throughout the paper that there is at least one non-empty bubble in the first and last column of $\mathcal{B}$, which means that for $j \in \{1, s\}$ there is $1 \leq i \leq r_j$ such that $B_{i,j} \neq \emptyset$. The *column number* of $\mathcal{B}$, denoted as $\#\mathrm{col}(\mathcal{B})$, is the number of columns of $\mathcal{B}$; in our case, $\#\mathrm{col}(\mathcal{B}) = s$. A simple example of a proper interval graph and a bubble model for it is given in Figure 1. As in the figure, a bubble model may contain empty and non-empty bubbles, and the number of non-empty bubbles in the single columns can be different. As a convention throughout the paper, if the indices of a bubble $B_{i,j}$ exceed the values of $\mathcal{B}$, i.e., if $j < 1$ or $j > s$ or $i < 1$ or $i > r_j$ (for $1 \leq j \leq s$), we assume that $B_{i,j}$ still exists and is empty. The interior of $\mathcal{B}$ is the set of bubbles of $\mathcal{B}$ that are above a non-empty bubble. Formally, the *interior* of $\mathcal{B}$, denoted as $\mathrm{in}(\mathcal{B})$, is the set $\{(i,j) : 1 \leq j \leq s \text{ and } 1 \leq i \text{ and } \exists i'(i \leq i' \text{ and } B_{i',j} \neq \emptyset)\}$. We say that $\mathcal{B}$ is *full* if for every $(i,j) \in \mathrm{in}(\mathcal{B})$, $B_{i,j}$ contains at least one vertex. Proper interval graphs with full bubble models are efficiently recognisable.

Proposition 3.1 *There is a linear-time algorithm that on input a connected proper interval graph G decides whether G has a full bubble model, and if so, outputs a full bubble model for G where true twins appear in the same bubble.*

4 An upper bound on the linear clique-width

In this section, we give an efficient algorithm for computing a linear expression for proper interval graphs with full bubble models. This algorithm will provide an upper bound on the clique-width as well as the linear clique-width of the graph.

Let G be a proper interval graph with bubble model $\mathcal{B} = \langle B_{i,j} \rangle_{1 \leq j \leq s, 1 \leq i \leq r_j}$. For $k \geq 0$, a *k-box* in $\mathcal{B}$ is the collection of bubbles $B_{a,b}, \ldots, B_{a+k,b}, B_{a,b+1}, \ldots, B_{a+k,b+k-1}$ for $1 \leq b \leq s-k+1$ such that $a + k \leq r_j$ for all $b \leq j \leq b+k-1$. Intuitively, a k-box is a rectangle of height $k+1$ and width k that can be placed in $\mathcal{B}$ in the range of $\mathrm{in}(\mathcal{B})$. The pair (a,b) is called the *origin* of the box. It is immediately clear that if (a,b) is the origin of a k-box then also $(1,b)$ is the origin of a k-box. The *box number* of $\mathcal{B}$ is the largest k such that $\mathcal{B}$ has a k-box.

Lemma 4.1 *Let G be a connected proper interval graph that has a full bubble model. Let $\mathcal{B} = \langle B_{i,j} \rangle_{1 \leq j \leq s, 1 \leq i \leq r_j}$ be a full bubble model for G where true twins appear in the same bubble.*

1) Let κ be the box number of $\mathcal{B}$ plus 1.

2) Let $\alpha =_{\mathrm{def}} 0$ if $\mathcal{B}$ has at most one vertex per bubble; otherwise, let $\alpha =_{\mathrm{def}} 1$.

3) Let $b \leq s$ be smallest such that $\mathcal{B}$ contains a $(\kappa - 1)$-box with origin $(1,b)$. If $\mathcal{B}_{b+\kappa-1}$ contains at least two vertices then let $\beta =_{\mathrm{def}} 0$; otherwise, let $\beta =_{\mathrm{def}} -1$.

Then, $\mathrm{lcwd}(G) \leq (\kappa+2) + \alpha + \beta$ and a linear $(\kappa+2+\alpha+\beta)$-expression for G can be computed in time $\mathcal{O}(n^2)$.

Proof. We define a linear $(\kappa+2+\alpha+\beta)$-expression for G. Let $\mathscr{H}$ be the set of indices j with $1 \leq j \leq s$ and $r_j \leq \kappa$. Informally, $\mathscr{H}$ represents the set of "short" columns of $\mathcal{B}$; the other columns are called "long". Let $\mathscr{H} = \{j_1, \ldots, j_p\}$, where we assume without loss of generality that $j_1 < \cdots < j_p$. By definition of $\mathscr{H}$, it holds that $r_j \geq \kappa+1$ for every $1 \leq j \leq s$ with $j \notin \mathscr{H}$. Let $j_0 =_{\mathrm{def}} 0$ and $j_{p+1} =_{\mathrm{def}} s+1$. It follows that $j_{i+1} - j_i \leq \kappa$ for all $0 \leq i \leq p$. Otherwise, $j_{i+1} - j_i \geq \kappa + 1$ for some $0 \leq i \leq p$ implies the existence of a κ-box with origin $(1, j_i + 1)$ in $\mathcal{B}$, which contradicts the definition of κ as being larger than the box number of $\mathcal{B}$. So, there are at most $\kappa - 1$ long columns between each pair of consecutive short columns in $\mathcal{B}$. For the construction of our expression for G, we partition $\mathcal{B}$ into small parts of long columns, separated by short columns, and process $\mathcal{B}$ from right to left. Due to space restrictions, we concentrate on the case when $\alpha = 0$ and $\beta = 0$.

We construct the linear expression for G inductively. For convenience reasons, we assume that $\mathcal{B}$ also has columns $\mathcal{B}_0$ and $\mathcal{B}_{s+1}$, and $r_0 =_{\mathrm{def}} r_{s+1} =_{\mathrm{def}} 0$. Let $1 \leq a \leq p+1$. Denote by G_a the subgraph of G that is induced by the vertices in the columns $\mathcal{B}_{j_a}, \ldots, \mathcal{B}_{s+1}$. We assume that we already have a linear expression $\mathscr{E}_a$ for G_a that defines a labelled graph with the following labels: the vertices in the columns $\mathcal{B}_{j_a+1}, \ldots, \mathcal{B}_{s+1}$ have label 1, and the vertices in B_{i,j_a} have

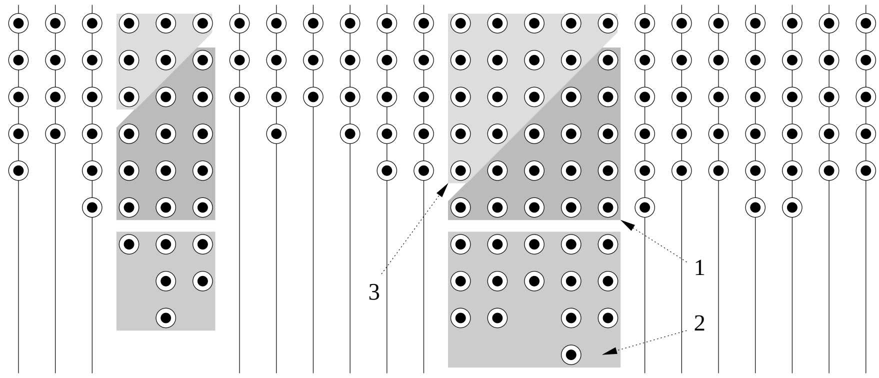


Figure 2: Depicted is a bubble model for a proper interval graph. The box number is 5, as the largest box has width 5 and height 6. The vertical line segments identify "short columns", that have no vertex in row 7. The other, "long", columns, which have a vertex in row 7, form areas that are partitioned into subareas. There are two areas of long columns in the depicted bubble model. Each of the areas of long columns is partitioned into three subareas, that are identified by the shaded backgrounds and numbered in the right hand side occurrence. The proof of Lemma 4.1 gives an algorithm for computing a linear expression and treats long columns particularly different from short columns.

label $i+1$ for every $1 \leq i \leq r_{j_a} \leq \kappa$ (remember that $\mathcal{B}_{j_a}$ is a short column). It is important to note that we can assume this particularly for the induction base case of $a = p+1$. For the later arguments, observe the following facts:

– (since $\mathscr{E}_a$ defines a labelled graph) all vertices of G_a have a label

– only labels from the set $\{1, \ldots, \kappa+1\}$ are used

– vertices with the same label have the same neighbours outside of G_a and no vertex with label 1 has a neighbour outside of G_a.

We show that we can define a linear expression $\mathscr{E}_{a-1}$ for G_{a-1} that uses at most $(\kappa+2)+\alpha$ labels.

We construct $\mathscr{E}_{a-1}$ in three phases. We partition the vertices in columns $\mathcal{B}_{j_{a-1}+1}, \ldots, \mathcal{B}_{j_a-1}$ into three areas, as indicated by the three different area backgrounds in Figure 2:

– area 1: bubbles $B_{j_a-j+1,j}, \ldots, B_{\kappa,j}$ for all $j_{a-1} < j < j_a$

– area 2: bubbles $B_{\kappa+1,j}, \ldots, B_{r_j,j}$ for all $j_{a-1} < j < j_a$

– area 3: bubbles $B_{1,j}, \ldots, B_{j_a-j,j}$ for all $j_{a-1} < j < j_a$.

We first add the vertices from area 1, then from area 2 and finally from area 3. Let $W =_{\text{def}} j_a - j_{a-1} - 1$, which is the number of columns that are properly between $\mathcal{B}_{j_{a-1}}$ and $\mathcal{B}_{j_a}$. In other words, these are the W consecutive long columns that we will add during this phase. Note that area 3 has the shape of an isosceles triangle, whereas area 1 has the ideal shape as in Figure 2 only in case when W has its maximum possible value, which is $W = \kappa - 1$.

Throughout the following construction, we will use label $\kappa+2$ for introducing a new vertex; we will refer to it as "creating the vertices in a bubble". For obtaining $\mathscr{E}_{a-1}$ by extending $\mathscr{E}_a$, we consider the vertices in areas 1, 2, 3. This is of particular interest, if $W \geq 1$. Otherwise, the three areas are empty. Due to space restrictions, we concentrate on the case when $W \geq 2$.

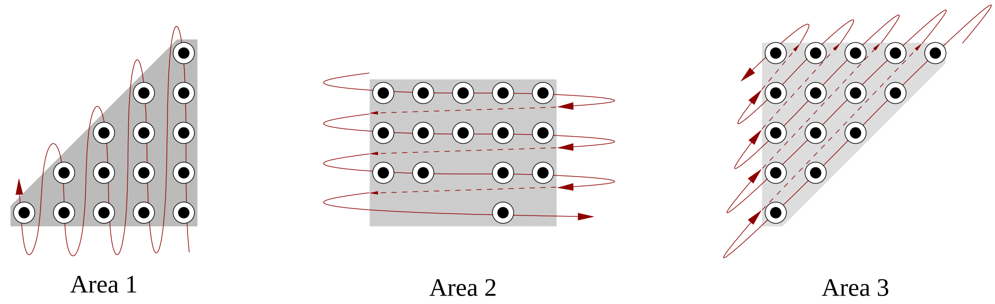


Figure 3: In the proof of Lemma 4.1: the three areas in Figure 2 are processed separately and according to the scheme, indicated by the curves.

We partition the construction for this case into three smaller parts according to the description in Figure 2. The vertices in each area are processed according to special pattern. The three patterns are sketched in Figure 3.

Area 1

This area is marked with number 1 in Figure 2. We process W column intervals of $\mathcal{B}$, starting with the longest one. The first interval consists of the bubbles $B_{2,j_a-1}, \ldots, B_{\kappa,j_a-1}$; we process these bubbles in a bottom-to-top manner. We create the vertices in B_{κ,j_a-1}, then we make all vertices with label $\kappa+2$ adjacent to all vertices with labels $2, \ldots, \kappa$. After this, the vertices in $B_{\kappa-1,j_a}$ and B_{κ,j_a} are not distinguishable anymore. So, we change label κ to label $\kappa+1$, and then, we change label $\kappa+2$ to label κ.

If $\kappa \geq 3$, we proceed with the vertices in $B_{\kappa-1,j_a-1}$. We create the vertices in $B_{\kappa-1,j_a-1}$, then we make the vertices with label $\kappa+2$ adjacent to the vertices with labels $2, \ldots, \kappa$. We change label $\kappa-1$ to label $\kappa+1$, and then, we change label $\kappa+2$ to label $\kappa-1$. This process is continued analogously until the vertices in B_{2,j_a-1} have been processed. Then, it holds that all vertices in $\mathcal{B}_{j_a}$ have label $\kappa+1$, and the vertices in B_{i,j_a-1} have label i for all $2 \leq i \leq \kappa$.

We analogously continue with the next intervals, until all bubbles of area 1 have been processed. At the end of the process, the followings holds for the vertices in area 1: the vertices in column $\mathcal{B}_{j_{a-1}+j}$ have label $\kappa - W + j$ for all $2 \leq j \leq W$, the vertices in bubble $B_{i,j_{a-1}+1}$ have label $i - W + 1$ for all $W+1 \leq i \leq \kappa$, and the vertices in column $\mathcal{B}_{j_a}$ have label $\kappa+1$.

Area 2

This area is marked with number 2 in Figure 2. We process the rows in a top-to-bottom manner and within a row from left to right. The first vertices to process are from bubble $B_{\kappa+1,j_{a-1}+1}$. We create the vertices in $B_{\kappa+1,j_{a-1}+1}$. The vertices with label $\kappa+2$ have to be made adjacent to all already created vertices in columns $\mathcal{B}_{j_{a-1}+1}$ and $\mathcal{B}_{j_{a-1}+2}$, which are exactly the vertices with labels $2, \ldots, \kappa - W + 2$. So, we make all vertices with label $\kappa+2$ adjacent to all vertices with labels $2, \ldots, \kappa - W + 2$, and then, we change label $\kappa+2$ to label $\kappa - W + 1$. Note here that the vertices with label $\kappa - W + 1$ are exactly the vertices in bubbles $B_{\kappa,j_{a-1}+1}$ and $B_{\kappa+1,j_{a-1}+1}$, that have no neighbours in column $\mathcal{B}_{j_{a-1}}$.

We continue and create the vertices in $B_{\kappa+1,j_{a-1}+2}$, make the vertices with label $\kappa+2$ adjacent to the vertices with labels $\kappa - W + 2$ and $\kappa - W + 3$, and then change label $\kappa+2$ to $\kappa - W + 2$. This process continues until bubble $B_{\kappa+1,j_a-1}$ has been executed.

If there is $r_j > \kappa + 1$ for some $j_{a-1} < j < j_a$, we repeat the procedure with the bubbles $B_{\kappa+2,j_{a-1}+1}, \ldots, B_{\kappa+2,j_a-1}$, and so on until all vertices from area 2 have been created. When all rows of area 2 have been processed, all neighbours of the vertices in column $\mathcal{B}_{j_a}$ have been created and made adjacent, so that we can change label $\kappa+1$ to 1. It holds that after completion of this area 2, the already created vertices in column $\mathcal{B}_{j_{a-1}+j}$ have label $\kappa - W + j$ for all $2 \leq j \leq W$. For $\mathcal{B}_{j_{a-1}+1}$, it holds that the vertices in $B_{i,j_{a-1}+1}$ have label $i - W + 1$ for all

$W+1 \leq i < \kappa$ and the other vertices have label $\kappa - W + 1$. The vertices in columns $\mathcal{B}_{j_a}, \ldots, \mathcal{B}_{j_{p+1}}$ have label 1. It is important to note here that label $\kappa + 1$ is not used.

Area 3

This area is marked with number 3 in Figure 2. We process diagonals. Remember that the shape of this area is an isosceles triangle. This also means that there are exactly W diagonals to process. We begin with the longest diagonal and end with the shortest diagonal. Bubbles on diagonals are processed from upper right to lower left. We create the vertices in B_{1,j_a-1}, make all vertices with label $\kappa+2$ adjacent to all vertices with label κ, and change label $\kappa+2$ to label $\kappa+1$. Next, we create the vertices in B_{2,j_a-2}, make all vertices with label $\kappa+2$ adjacent to the already created vertices from $\mathcal{B}_{j_a-2}$ and to the vertices with label $\kappa+1$, change label $\kappa+1$ to κ, and then change label $\kappa+2$ to $\kappa+1$. We continue this until we completed bubble $B_{W,j_a-W} = B_{W,j_{a-1}+1}$. Note that, similar to the case of area 2, special care has to be taken of the different labels of the vertices in $\mathcal{B}_{j_{a-1}+1}$. All vertices in column $\mathcal{B}_{j_a-1}$ have now been created and made adjacent to all their neighbours. So, we change label κ to label 1. Then, we change label $\kappa+1$ to label κ.

We continue and repeat the process analogously with the next smaller diagonal, and repeat until all bubbles from area 3 except for $B_{1,j_{a-1}+1}$ have been processed. The situation now is the following. The vertices from columns $\mathcal{B}_{j_{a-1}+2}, \ldots, \mathcal{B}_{j_{p+1}}$ have label 1. The vertices in bubble $B_{i,j_{a-1}+1}$ have label $\kappa - W + i$ for all $2 \leq i \leq W$, the vertices in bubble $B_{i,j_{a-1}+1}$ have label $i - W + 1$ for all $W+1 \leq i < \kappa$, and finally, the vertices in bubble $B_{i,j_{a-1}+1}$ have label $\kappa - W + 1$ for all $\kappa \leq i \leq r_{j_{a-1}+1}$. Note again that label $\kappa+1$ is not used.

We complete area 3 by finally creating the vertices in $B_{1,j_{a-1}+1}$ and making them adjacent to all other vertices in $\mathcal{B}_{j_{a-1}+1}$. Then, all vertices from $\mathcal{B}_{j_{a-1}+1}$ have been created, and all neighbours of the vertices in $B_{\kappa,j_{a-1}+1}, \ldots, B_{r_{j_{a-1}+1},j_{a-1}+1}$ have been created and made adjacent. So, the label of these vertices, that are exactly the vertices with label $\kappa - W + 1$, can be changed to label 1. Remember that $\mathcal{B}_{j_{a-1}}$ is a short column. Using the free label $\kappa+1$, we can change the labels of the other vertices in $\mathcal{B}_{j_{a-1}+1}$ so that it holds for all $1 \leq i < \kappa$ that the vertices in $B_{i,j_{a-1}+1}$ have label $i+1$. This completes the construction for the vertices in area 3.

For completing the definition of $\mathscr{E}_{a-1}$, it remains to add the vertices from column $\mathcal{B}_{j_{a-1}}$. This follows the procedure laid out for the vertices in area 1. If $W = 0$ then the vertices in B_{κ,j_a} have no neighbour in $\mathcal{B}_{j_{a-1}}$, and so we change label $\kappa+1$ to label 1. We create the vertices in $B_{\kappa,j_{a-1}}$, make all vertices with label $\kappa+2$ adjacent to all vertices with labels $2, \ldots, \kappa$, change label κ to label 1, and then change label $\kappa+2$ to label $\kappa+1$. We continue with $B_{\kappa-1,j_{a-1}}$ and proceed until the vertices in $B_{1,j_{a-1}}$ have been created and finally received label 2. This completes the definition of $\mathscr{E}_{a-1}$ and satisfies the conditions. By induction, we obtain the claimed result. ∎

5 The approximation result

The previous section has established an upper bound on the linear clique-width, and therefore also clique-width, of proper interval graphs with full bubble models. The upper bound is dependent on the box number parameter. In this section, we complete this result by giving lower bounds. The upper and lower bounds together will provide an approximation on the clique-width and linear clique-width of proper interval graphs with full bubble models.

We begin by showing a strong connection between the box number of a proper interval graph with full bubble model and the existence of a well-structured induced subgraph. An *induced path* on n vertices is a graph P that admits a vertex sequence $(x_1, \ldots, x_n)$ such that $E(P) = \{x_1x_2, \ldots, x_{n-1}x_n\}$ are the edges of P. For $k \geq 1$, the *kth power* of P is the graph G on vertex set $\{x_1, \ldots, x_n\}$, and two vertices x_i, x_j of G are adjacent if and only if $|i-j| \leq k$. For $k \geq 1$, a *k-path power* is a graph that is the kth power of some induced path. Every k-path power is a proper interval graph.

Lemma 5.1 *Let G be a proper interval graph with full bubble model $\mathcal{B} = \langle B_{i,j} \rangle_{1 \le j \le s, 1 \le i \le r_j}$. Let $\mathcal{B}$ contain a k-box with origin $(1,b)$, where $k \ge 1$. Then, G contains a k-path power on $k(k+1)$ vertices as induced subgraph. If $b \le s-k$ and $r_{b+k} \ge 2$ then G contains a k-path power on $k(k+1)+2$ vertices as induced subgraph.*

Conversely, if G contains a k-path power on $k(k+1)+2$ vertices as induced subgraph where $k \ge 1$ then $\mathcal{B}$ contains a k-box with origin $(1,b)$ and where $r_{b+k} \ge 2$.

The linear clique-width of k-path powers is completely characterised [11], and there exists a very good lower bound on the clique-width of k-path powers [9]. The following proposition summarises the cases that are important for our results.

Proposition 5.2 ([11], [9]) *Let G be a k-path power on n vertices, where $k \ge 1$.*

- *If $n = k(k+1)+2$ then $\mathrm{lcwd}(G) = k+2$ and $\mathrm{cwd}(G) \ge k$.*
- *If $n = k(k+1)$ then $\mathrm{lcwd}(G) = k+1$ and $\mathrm{cwd}(G) \ge k$.*

Combining the previous two results, we obtain the following lower bounds on the clique-width and linear clique-width of proper interval graphs with full bubble models.

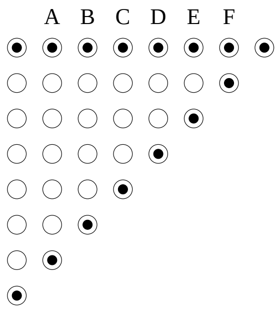
Corollary 5.3 *Let G be a proper interval graph with full bubble model $\mathcal{B} = \langle B_{i,j} \rangle_{1 \le j \le s, 1 \le i \le r_j}$. Let $\mathcal{B}$ contain a k-box with origin $(1,b)$, where $k \ge 1$. Then, $\mathrm{lcwd}(G) \ge k+1$ and $\mathrm{cwd}(G) \ge k$. If $b \le s-k$ and $r_{b+k} \ge 2$ then $\mathrm{lcwd}(G) \ge k+2$.*

Proof. Due to Lemma 5.1, G contains a k-path power on $k(k+1)$ vertices as induced subgraph. Then, due to Proposition 5.2, G contains an induced subgraph H with $\mathrm{lcwd}(H) = k+1$ and $\mathrm{cwd}(H) \ge k$, and thus, $\mathrm{lcwd}(G) \ge k+1$ and $\mathrm{cwd}(G) \ge k$. Now, let $b \le s-k$ and $r_{b+k} \ge 2$. Due to Lemma 5.1, G contains a k-path power on $k(k+1)+2$ vertices as induced subgraph. Analogous to the above case and due to Proposition 5.2, G contains an induced subgraph H with $\mathrm{lcwd}(H) = k+2$, and therefore, $\mathrm{lcwd}(G) \ge k+2$. ∎

Let G be a graph. If G contains no pair of adjacent vertices then $\mathrm{cwd}(G) \le \mathrm{lcwd}(G) \le 1$. Otherwise, $\mathrm{lcwd}(G) \ge \mathrm{cwd}(G) \ge 2$. It is easy to check that $\mathrm{lcwd}(G) \ge \mathrm{cwd}(G) \ge 3$ if G contains an induced path on four vertices as induced subgraph. With these additional and easy bounds on clique-width, we are ready to prove the main result of the paper.

Theorem 5.4 *There is an $\mathcal{O}(n^2)$-time algorithm that on input a connected proper interval graph G with full bubble model, approximates the clique-width of G within $\mathrm{cwd}(G)+3$ and approximates the linear clique-width of G within $\mathrm{lcwd}(G)+3$. If G contains no true twins then the algorithm approximates the linear clique-width of G within $\mathrm{lcwd}(G)+1$.*

Proof. Let G be a connected proper interval graph with full bubble model. If u and v are true twins of G then $\mathrm{cwd}(G) = \mathrm{cwd}(G-v)$. With iterated application of this argument, it holds that $\mathrm{cwd}(G) = \mathrm{cwd}(G')$ where G' is a maximal induced subgraph of G without true twins. Let $\mathcal{B} = \langle B_{i,j} \rangle_{1 \le j \le s, 1 \le i \le r_j}$ be a full bubble model for G where true twins appear in the same bubble. Remember that $\mathcal{B}$ exists according to Proposition 3.1. Let k be the box number of $\mathcal{B}$. If G is a complete graph then $\mathrm{cwd}(G) \le \mathrm{lcwd}(G) \le 2$. Henceforth, let G not be complete. This particularly means that $s \ge 2$ and $r_1, \ldots, r_{s-1} \ge 2$. Thus, $k \ge 1$, since $\mathcal{B}$ has a 1-box with origin $(1,1)$. We distinguish between two cases according to the value of k. As the first case, let $k = 1$. If $s = 2$ and $r_s = 1$ then $2 \le \mathrm{cwd}(G) \le \mathrm{lcwd}(G) \le 3$. Otherwise, if $r_2 \ge 2$, G contains an induced path on four vertices as induced subgraph, by choosing a vertex from each of the four bubbles $B_{1,1}, B_{2,1}, B_{1,2}, B_{2,2}$. This means that $\mathrm{lcwd}(G) \ge \mathrm{cwd}(G) \ge 3$. We apply the algorithm of Lemma 4.1. It holds that $\kappa = 2$ and $\beta \le 0$ and $\alpha \le 1$. This gives



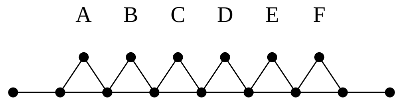


Figure 4: The left hand side shows a bubble model of the right hand side depicted graph. The graph has clique-width 3 and linear clique-width 3 and the box number of the bubble model is 4.

$3 \leq \mathrm{cwd}(G) \leq \mathrm{lcwd}(G) \leq (\kappa + 2) + \alpha \leq \mathrm{cwd}(G) + 2$. If G has no true twins then $\alpha = 0$, and thus, $(\kappa + 2) + \alpha \leq \mathrm{lcwd}(G) + 1$.

Now, let $k \geq 2$. We first consider linear clique-width. We apply the algorithm of Lemma 4.1. It holds that $\kappa = k + 1$ and $\beta \leq 0$ and $\alpha \leq 1$. Hence, $\mathrm{cwd}(G) \leq \mathrm{lcwd}(G) \leq (k + 3) + \alpha + \beta$. We first consider linear clique-width and the case when G has no true twins; in particular, $\alpha = 0$. Let b be smallest such that $\mathcal{B}$ has a k-box with origin $(1, b)$. Due to definition, $\beta = 0$ if and only if $r_{b+k} \geq 2$, where we assume $r_{s+1} = 0$. Due to Corollary 5.3, it holds that $\mathrm{lcwd}(G) \geq k+2+\beta$. We combine the upper and lower bound results and obtain that $\mathrm{lcwd}(G) \leq (k+3)+\beta \leq \mathrm{lcwd}(G)+1$. If G has true twins then $\alpha = 1$. Due to Corollary 5.3, $\mathrm{lcwd}(G) \geq k + 1$, which shows that $\mathrm{lcwd}(G) \leq (k + 3) + 1 \leq \mathrm{lcwd}(G) + 3$. In case of clique-width, we consider $\mathrm{cwd}(G')$. We obtain a bubble model $\mathcal{B}'$ for G' by deleting the vertices in $V(G) \setminus V(G')$ from $\mathcal{B}$. Due to the assumption about the true twins being in the same bubble, $\mathcal{B}'$ is a full bubble model for G', and the box number of $\mathcal{B}'$ is k. Due to Corollary 5.3, $\mathrm{cwd}(G') \geq k$, and applying Lemma 4.1, where $\kappa = k + 1$, $\beta \leq 0$ and $\alpha = 0$, we obtain together with the above lower bound that $\mathrm{cwd}(G) \leq k + 3 \leq \mathrm{cwd}(G) + 3$. This completes the proof of the theorem. ∎

6 Final remarks

We gave an efficient approximation algorithm for computing the clique-width and the linear clique-width of proper interval graphs with a full bubble model. The correctness and quality of the result relies on almost tight lower and upper bounds. The lower bound on the clique-width and linear clique-width is obtained from the results about path powers and the close relationship between the box number and the existence of large path powers as induced subgraphs in these proper interval graphs. In Figure 4, we give an example showing that this close relationship does not extend to arbitrary proper interval graphs.

Can our results help to understand and solve the general case for proper interval graphs? We think that our results are indeed very helpful for the continuation of the work with the aim of computing the clique-width and linear clique-width of proper interval graphs exactly. A closer study of our results shows that the approximation result can be extended to a much larger class of proper interval graphs, namely to all proper interval graphs for which the box number gives a good approximation on the size of an induced subgraph that is a large path power. Another possible extension is by identifying another class of well-structured proper interval graphs, similar to path powers, that can provide a lower bound on the clique-width and linear clique-width of proper interval graphs. Up to now, path powers are the only proper interval graphs for which such lower bound results exist.

Finally, we want to mention that our approximation result gives hope even for an algorithm that computes the clique-width and linear clique-width of proper interval graphs with full bubble models exactly. We believe that our upper bounds for graphs without true twins are tight. So, the main challenge is a matching lower bound. The currently most promising approach is again the identification of small and well-structured induced subgraphs that already require larger clique-width.

Acknowledgements

The authors are grateful to Charis Papadopoulos and Daniel Lokshtanov for fruitful discussions on the subject and valuable hints.

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 (Extended Abstract). *Proceedings of LATIN 2000*, Lecture Notes in Computer Science, 1776:126–134, 2000.

[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] P. Heggernes, D. Meister, C. Papadopoulos. A new representation of proper interval graphs with an application to clique-width. *Electronic Notes in Discrete Mathematics*, 32:27–34, 2009.

[11] P. Heggernes, D. Meister, C. Papadopoulos. A Complete Characterisation of the Linear Clique-Width of Path Powers *Proceedings of TAMC 2009*, Lecture Notes in Computer Science, 5532:241–250, 2009.

[12] P. Hlinený and S.-I. Oum. Finding branch-decompositions and rank-decompositions. *SIAM Journal on Computing*, 38:1012–1032, 2008.

[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] 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.

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

[16] 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.

[17] H. Müller and R. Urner. On a disparity between relative cliquewidth and relative NLC-width. *Discrete Applied Mathematics*, to appear.

[18] S.-I. Oum. Approximating rank-width and clique-width quickly. *ACM Transactions on Algorithms*, 5, 2008.

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

[20] F. S. Roberts. Indifference graphs. In: F. Harary (Ed.), *Proof techniques in graph theory*, pp. 139–146, Academic Press, New York, 1969.

Appendix

A graph G is a proper interval graph if and only if there is a vertex ordering $\langle x_1, \ldots, x_n \rangle$ for G such that for every triple i, j, k with $1 \leq i < j < k \leq n$, if $x_i x_k \in E(G)$ then $x_i x_j \in E(G)$ and $x_j x_k \in E(G)$. Such an ordering is called *proper interval ordering*, and it can be computed in linear time.

Let G be a proper interval graph without true twins. Let $\mathcal{B} = \langle B_{i,j} \rangle_{1 \leq j \leq s, 1 \leq i \leq r_j}$ be a bubble model for G. Then, no bubble of $\mathcal{B}$ contains more than one vertex. Every bubble model defines a vertex ordering in the following way. We lay out the bubbles of $\mathcal{B}$ as

$$B_{1,1}, \ldots, B_{r_1,1}, B_{1,2}, \ldots, B_{r_2,2}, B_{1,3}, \ldots, B_{r_s,s}\,.$$

The vertex ordering defined by $\mathcal{B}$ orders the vertices of G as they appear in the layout of the bubbles. It holds that there are at most two vertex orderings for proper interval graphs without true twins that are defined by bubble models.

Proof of Proposition 3.1. The main idea for the algorithm is the tight relationship between bubble models and proper interval orderings. Let G be a proper interval graph, and assume that G has a full bubble model $\mathcal{B} = \langle B_{i,j} \rangle_{1 \leq j \leq s, 1 \leq i \leq r_j}$. Assume that G has true twins, for instance vertices u and v where $u \in B_{i,j}$ and $v \in B_{i',j'}$ for some $1 \leq j, j' \leq s$ and $1 \leq i \leq r_j$ and $1 \leq i' \leq r_{j'}$. Without loss of generality, we can assume that $j \leq j'$. Since u and v have the same neighbours, it holds that $B_{1,j} \subseteq N_G[u] \cap N_G[v]$. The properties of bubble models imply that $j = j'$. We can assume without loss of generality that $i \leq i'$. Assume that $i < i'$. Since v is non-adjacent to the vertices in $B_{i+1,j-1}$ and since u is non-adjacent to the vertices in $B_{i,j+1}$, it must hold that $B_{i+1,j-1} \cup B_{i,j+1} = \emptyset$. Since $\mathcal{B}$ is a full bubble model, it follows that $B_{i+1,j-1} \cup \cdots \cup B_{r_j,j-1} = \emptyset$ and $B_{i,j+1} \cup \cdots \cup B_{r_j,j+1} = \emptyset$. Then, the bubble model $\mathcal{B}'$ that is equal to $\mathcal{B}$ with the two exceptions $B'_{i,j} = B_{i,j} \cup \cdots \cup B_{r_j,j}$ and $r'_j = i$ is a full bubble model for G. Iterating this process, we conclude that G has a full bubble model if and only if G has a full bubble model where true twins appear in the same bubble. Let G' be obtained from G by iteratively deleting a vertex that has a true twin. Note that the result is independent of the actual chosen vertex up to isomorphism. It follows with the above equivalence that G has a full bubble model if and only if G' has a full bubble model.

With the above result, we can assume in the following that G has no true twins. This particularly means that every bubble of $\mathcal{B}$ contains at most one vertex. It holds that $B_{1,1} \cup B_{1,2} \cup \cdots \cup B_{1,s}$, which is the set of vertices from the first row of $\mathcal{B}$, is an independent set of G. Let $\langle x_1, \ldots, x_n \rangle$ be the vertex ordering defined by $\mathcal{B}$. Then, $x_1 \in B_{1,1}$, and the vertex in $B_{1,2}$ is x_i where i is smallest possible such that $x_1 x_i \notin E(G)$. We can say that x_i is the closest non-neighbour of x_1. This continues in the sense that the vertex in $B_{1,j+1}$ is the closest non-neighbour of the vertex in $B_{1,j}$ for every $1 \leq j < s$. We can view this set as a greedy maximal independent set of G.

For the algorithm, let G be a connected proper interval graph without true twins and let $\sigma = \langle x_1, \ldots, x_n \rangle$ be a proper interval ordering for G. Remember that σ can be computed in linear time. If G has a full bubble model then G has a full bubble model that defines σ or its reverse. So, it suffices to check whether one of the two vertex orderings admits a full bubble model. The procedure is easy, that we describe informally here. It heavily relies on the fact that we are attempting to obtain a full bubble model. We construct bubble model $\mathcal{B}$. So, place x_1 in bubble $B_{1,1}$, then x_2 in $B_{2,1}$, then x_3 in $B_{3,1}$ and so on until we encounter a non-neighbour of x_1; let it be x_i. Then, x_i is placed in $B_{1,2}$ and we continue in column $\mathcal{B}_2$ with the remaining neighbours of x_i and move on to column $\mathcal{B}_3$, and so on. After finishing the construction, it remains to check whether $\mathcal{B}$ is a bubble model for G. It is clear that $\mathcal{B}$ is full. If the test fails here, we try the reversed vertex ordering. If that test also fails then G has no full bubble model due to the arguments in the above paragraphs. Each test requires linear running time, which gives linear running time for the whole algorithm. ■

Proof of Lemma 5.1. For the first statement, choose a vertex from each bubble in the k-box with origin $(1, b)$, and let H be the subgraph of G that is induced by the chosen vertices. Due to the definition of k-box, H contains exactly $k+1$ vertices from each of k consecutive columns of $\mathcal{B}$. It is not difficult to verify with the properties of bubble models that these $k(k+1)$ vertices induce a k-path power. If column $\mathcal{B}_{b+k}$ contains vertices in at least two bubbles then H can be extended by two vertices from G to obtain H', that is a k-path power on $k(k+1)+2$ vertices. Note that this is only true since $\mathcal{B}$ is a full bubble model. It holds that $\mathcal{B}_{b+k}$ contains vertices in at least two bubbles if and only if $r_{b+k} \geq 2$.

For the second statement, let G contain a k-path power H on $k(k+1)+2$ vertices as induced subgraph. Since H contains no true twins, the vertices of H appear in separate bubbles of $\mathcal{B}$. We look at the maximal cliques of H. Every maximal clique of H contains exactly $k+1$ vertices. A maximal clique of G consists of the vertices of a single column in $\mathcal{B}$ or is formed by two half columns: for some $1 \leq j < s$ and $1 \leq i < r_j$, $B_{1,j+1}, \ldots, B_{i,j+1}, B_{i+1,j}, \ldots, B_{r_j,j}$. Due to the observation above and since every maximal clique of H consists of $k+1$ vertices, every maximal clique requires $k+1$ bubbles of $\mathcal{B}$. Therefore, for every column $\mathcal{B}_j$ that contains a vertex of H, it holds that $r_{j-1} \geq k+1$ or $r_j \geq k+1$. Let c be smallest such that column $\mathcal{B}_c$ contains a vertex of H, and let d be largest such that $\mathcal{B}_d$ contains a vertex of H. Let $c < j \leq d$ and let v be a vertex of H in $\mathcal{B}_j$ that has a neighbour in H in $\mathcal{B}_{j-1}$. Note that, if v has no such neighbour, then H would not be connected. Let u be a neighbour of v in H in $\mathcal{B}_{j-1}$. Then, $\{u, v\}$ is subset of a maximal clique of H, so that $r_{j-1} \geq k+1$. We conclude that $r_c, \ldots, r_{d-1} \geq k+1$. It remains to determine the difference $d-c$. Every column that contains vertices of H can contain at most $k+1$ vertices of H due to the size of a maximal clique. Due to $|V(H)| = k(k+1)+2$, it follows that $d-c \geq k$. Thus, choosing $b =_{\text{def}} c$, $\mathcal{B}$ contains a k-box with origin $(1, b)$. If $d-c \geq k+1$ then $c+k < d$ and thus $r_{b+k} \geq k+1 \geq 2$. Otherwise, if $d-c = k$ then $\mathcal{B}_c, \ldots, \mathcal{B}_{d-1}$ contain at most $k+1$ vertices of H each, so that $\mathcal{B}_d$ contains at least two vertices of H, and thus $r_d = r_{b+k} \geq 2$. ■