



Bandwidth Parameterized by Cluster Vertex Deletion Number

Tatsuya Gima¹ · Eun Jung Kim^{2,3,4} · Noleen Köhler⁵ · Nikolaos Melissinos⁶ · Manolis Vasilakis⁷

Received: 12 March 2024 / Accepted: 9 April 2025
© The Author(s) 2025

Abstract

Given a graph G and an integer b , BANDWIDTH asks whether there exists a bijection π from $V(G)$ to $\{1, \dots, |V(G)|\}$ such that $\max_{\{u,v\} \in E(G)} |\pi(u) - \pi(v)| \leq b$. This is a classical NP-complete problem, known to remain NP-complete even on very restricted classes of graphs, such as trees of maximum degree 3 and caterpillars of hair length 3. In the realm of parameterized complexity, these results imply that the problem remains NP-hard on graphs of bounded pathwidth, while it is additionally known to be W[1]-hard when parameterized by the tree-depth of the input graph. In contrast, the problem does become FPT when parameterized by the vertex cover number. In this paper we make progress in understanding the parameterized (in)tractability of BANDWIDTH. We first show that it is FPT when parameterized by the cluster vertex deletion number cvd plus the clique number ω , thus significantly strengthening the previously mentioned result for vertex cover number. On the other hand, we show that BANDWIDTH is W[1]-hard when parameterized only by cvd . Our results develop and generalize some of the methods of argumentation of the previous results and narrow some of the complexity gaps.

Keywords Bandwidth · Clique number · Cluster vertex deletion number · Parameterized complexity

1 Introduction

Given an undirected graph G and an integer b , BANDWIDTH asks whether there exists a bijection $\pi: V(G) \rightarrow \{1, \dots, |V(G)|\}$ of the vertices of G (called an *ordering*) such that $\max_{\{u,v\} \in E(G)} |\pi(u) - \pi(v)| \leq b$. The main motivation behind its study dates back to over half a century; a closely related problem in the field of matrix theory was first studied in the 1950's, while in the 1960's it was formulated as a graph problem, finding applications in minimizing (average) absolute error in codes, and has been extensively studied ever since [7, 12–14, 19, 20, 25, 33, 35].

An extended abstract of this work was presented at the 18th International Symposium on Parameterized and Exact Computation (IPEC 2023) [32].

Extended author information available on the last page of the article

BANDWIDTH is long known to be NP-complete [7, 49]; as a matter of fact, that is the case even on very restricted classes of graphs, such as trees of maximum degree 3 [30], caterpillars of hair length 3 [46], and cyclic caterpillars of hair length 1 [47]. Considering these NP-hardness results, in this paper we focus on the problem's parameterized complexity. When parameterized by the natural parameter b , BANDWIDTH is known to be in XP [34, 50], whilst it is XNLP-complete even when the input graph is a tree [3, 4], which implies $W[t]$ -hardness for all positive integers t . In fact, BANDWIDTH cannot be solved in time $f(b)n^{o(b)}$ even on trees of pathwidth at most two unless the Exponential Time Hypothesis fails [18], rendering the known $2^{\mathcal{O}(b)}n^{b+1}$ algorithm essentially optimal [50]. Regarding structural parameterizations, the previously mentioned results imply that BANDWIDTH is para-NP-complete when parameterized by the pathwidth or the treewidth plus the maximum degree of the input graph; the latter implies NP-completeness also on graphs of constant tree-cut width [29]. Moreover, it is known to be $W[1]$ -hard parameterized by the tree-depth of the input graph [31]. In contrast, the problem does become fixed-parameter tractable (FPT) when parameterized by the vertex cover number [23], the neighborhood diversity [1], or the max-leaf number [22].

In the last few years, a plethora of structural parameters have been introduced in an attempt to precisely determine the limits of tractability of algorithmic problems that are FPT parameterized by vertex cover number, yet become $W[1]$ -hard when parameterized by more general parameters, such as treewidth or clique-width. Some of the most well-studied such parameters are vertex integrity [31], tree-depth [48], twin cover number [27], cluster vertex deletion number [17], shrub-depth [28], neighborhood diversity [41], and modular-width [26]. The tractability of BANDWIDTH with respect to those parameters has remained largely unexplored, with the exception of neighborhood diversity [1] and tree-depth [31].

Cluster vertex deletion number lies between clique-width and vertex cover number (more precisely twin cover number) and is defined as the minimum size of a set of vertices whose removal induces a cluster graph, i.e., a disjoint union of cliques. Computing the cluster vertex deletion number of a graph is known to lie in FPT [5, 36, 51] and it was first considered as a structural parameter in [17], while it has been used to show parameterized (in)tractability results in multiple occasions ever since [2, 6, 8, 9, 39, 40, 44]. Notice that BANDWIDTH is trivial on cluster graphs; it suffices to check whether the clique number is at most $b + 1$, as there exists an optimal ordering that places the vertices of every clique consecutively, for some ordering of the cliques. Therefore, its tractability when parameterized by the cluster vertex deletion number of the input graph poses a very natural question.

Our contribution. In the current work, we present both tractability and intractability results for BANDWIDTH when cluster vertex deletion number is a parameter of the problem (see Fig. 1 for an overview of our results and the relationships between the structural parameters mentioned). We first prove that BANDWIDTH is FPT when parameterized by $\text{cvd} + \omega$, where cvd and ω denote the cluster vertex deletion number and clique number of the input graph respectively. This significantly strengthens the tractability result for vertex cover number of [23], and follows the same idea of encoding the problem as an *integer linear program* (ILP) of a small number of variables. Solving said ILP, one can verify whether there exists any ordering π of the vertices of

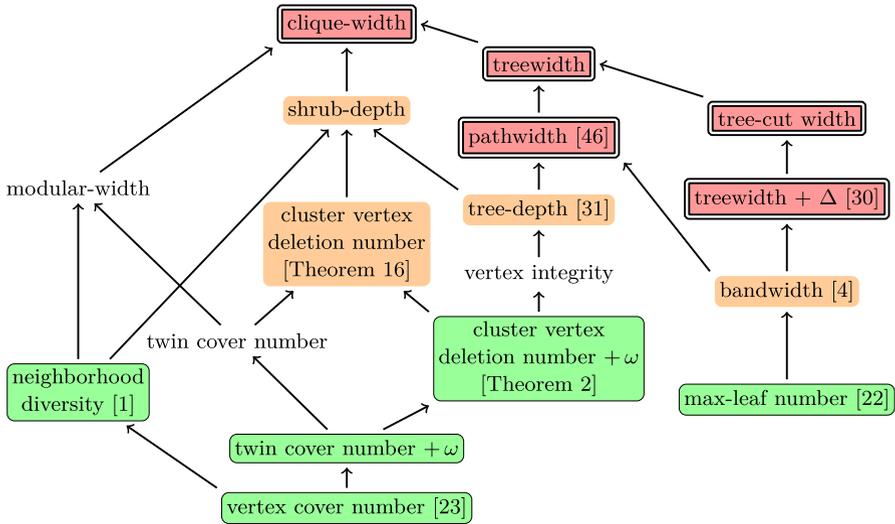


Fig. 1 Our results and hierarchy of some related structural graph parameters, where ω and Δ denote the clique number and the maximum degree of the input graph, respectively. Arrows between parameters indicate generalization relations, that is, for any graph, if the parameter at the tail of an arrow is a constant then the parameter at the head of the arrow is also a constant. The reverse does not hold in this figure. The framed green, frameless orange, and double framed red rectangles indicate fixed-parameter tractable, $W[*]$ -hard, and NP-complete cases, respectively

G such that (a) $|\pi(v) - \pi(u)| \leq b$ for all $\{u, v\} \in E(G)$, and (b) π is “nice”, where an ordering is nice if it has some specific properties. Proving that there exists a nice ordering π that minimizes $\max_{\{u,v\} \in E(G)} |\pi(v) - \pi(u)|$ then yields the stated result.

A natural question that arises from the previous result is whether it is necessary for both cvd and ω to be parameters of the problem in order to assure fixed-parameter tractability. Notice that **BANDWIDTH** is NP-complete even when $\omega \leq 2$, since that is the case for trees. Therefore, we proceed by studying the problem’s tractability when parameterized only by cvd . In this setting, we show that **BANDWIDTH** is $W[1]$ -hard via a reduction from **UNARY BIN PACKING**, thus positively answering the previous question.

Related work. **BANDWIDTH** is one of the so-called *graph layout* problems (see the survey of [15]). As far as the structural parameterized complexity of such problems is concerned, Fellows, Lokshtanov, Misra, Rosamond, and Saurabh [23] were the first to prove FPT results for a multitude of them when parameterized by the vertex cover number of the input graph, making use of ILP formulations. Since then, not much progress has been made on that front, with a notable exception being **IMBALANCE**, which was shown to be FPT when parameterized by twin cover plus ω [45], vertex integrity [31], or tree-cut width [29], while it belongs to XP when parameterized by twin cover [45]. **MINIMUM LINEAR ARRANGEMENT** is known to be FPT parameterized by max-leaf number, or edge clique number of the input graph [21], as well as by the vertex cover number [43]. Lastly, as far as **CUTWIDTH** is concerned, a

$2^{\mathcal{O}(\text{vc})}n^{\mathcal{O}(1)}$ time algorithm was presented in [11], improving over the ILP formulation of [23], where vc denotes the vertex cover number.

Organization. In Sect. 2 we discuss the general preliminaries, followed by the FPT-algorithm in Sect. 3 and the hardness result in Sect. 4. Lastly, in Sect. 5 we present the conclusion as well as some directions for future research.

2 Preliminaries

Throughout the paper we use standard graph notation [16], and we assume familiarity with the basic notions of parameterized complexity [10]. We assume that \mathbb{N} is the set of all non-negative integers. All graphs considered are undirected without loops. The *clique number* of a graph G , denoted by $\omega(G)$, is the size of its largest induced clique. For $x, y \in \mathbb{Z}$, let $[x, y] = \{z \in \mathbb{Z} : x \leq z \leq y\}$, while $[x] = [1, x]$. For $\mathcal{I}_i = [a_i, b_i]$, we say that intervals $\mathcal{I}_1, \dots, \mathcal{I}_k$ *partition* interval $\mathcal{I} = [a, b]$ if $\mathcal{I} = \bigcup_{i \in [k]} \mathcal{I}_i$ and $\mathcal{I}_i \cap \mathcal{I}_j = \emptyset$, for any $1 \leq i < j \leq k$. Additionally, let $\mathcal{I}_i < \mathcal{I}_j$ if $b_i < a_j$. For a function $f : A \rightarrow B$ and $A' \subseteq A$, let $f(A') = \{f(a) \in B : a \in A'\}$. Moreover, let $\max(f(A')) = \max\{f(a) : a \in A'\}$ and $\min(f(A'))$ defined analogously.

Let G be a graph and $\pi : V(G) \rightarrow [n]$ an ordering of its vertices. We define the *stretch* of an edge $e = \{u, v\} \in E(G)$ with regard to π as $\text{stretch}_\pi(e) = |\pi(u) - \pi(v)|$. We define the *stretch* of π to be the maximum stretch of the edges of G with regard to π , i.e., $\text{stretch}(\pi) = \max_{e \in E(G)} \text{stretch}_\pi(e)$. The *bandwidth* of G , denoted $\text{bw}(G)$, is the minimum stretch over all its possible vertex orderings, while BANDWIDTH asks, given a graph G and an integer b , whether $\text{bw}(G) \leq b$.

Remark 1 Note that the stretch of a vertex ordering is invariant under isomorphism, which means in particular that $\text{stretch}(\pi) = \text{stretch}(\pi \circ f)$ for any vertex ordering $\pi : V(G) \rightarrow [n]$ and any automorphism $f : V(G) \rightarrow V(G)$ of G .

A *cluster deletion set* of a graph G is a set $S \subseteq V(G)$ such that every component of $G - S$ is a clique. If S is a cluster deletion set, we call the components of $G - S$ *clusters*. The *cluster vertex deletion number* of G , denoted $\text{cvd}(G)$, is the size of its minimum cluster deletion set.

The feasibility variant of *integer linear programming* (ILP) is to decide, given a set X of variables and a set C of linear constraints (i.e., inequalities) over the variables in X with integer coefficients, whether there is an assignment $\alpha : X \rightarrow \mathbb{Z}$ of the variables satisfying all constraints in C . It is known that the feasibility of an instance of (ILP) can be tested in $\mathcal{O}(p^{2.5p+o(p)} \cdot L)$ time, where p is the number of variables and L is the size of the input [24, 38, 42]. In other words, computing the feasibility of an ILP formula is FPT parameterized by the number of variables. Moreover, a solution can be computed in the same time if it exists.

3 An FPT-Algorithm Parameterized by Cluster Vertex Deletion Number Plus Clique Number

In this section we prove that BANDWIDTH is FPT parameterized by the cluster vertex deletion number plus the clique number of the input graph. We first state the main theorem of this section, followed by a short discussion regarding the considered parameterization and a high-level overview of our proof.

Theorem 2 BANDWIDTH is fixed-parameter tractable when parameterized by $\text{cvd} + \omega$, where cvd and ω denote the cluster vertex deletion number and clique number of the input graph respectively.

Parameterization by $\text{cvd} + \omega$. Given a graph G and an integer k , it is known that one can compute a cluster deletion set of G of size at most k in time $2^{\mathcal{O}(k)} n^{\mathcal{O}(1)}$, or correctly determine that no such set exists [5, 36, 51]. Notice that one can use this algorithm in order to determine, for a graph G and an integer k , whether $\text{cvd}(G) + \omega(G) \leq k$ in time $2^{\mathcal{O}(k)} n^{\mathcal{O}(1)}$: (i) run the previously mentioned algorithm for $i = 1, \dots, \min\{\text{cvd}(G), k\}$ to compute a minimum cluster deletion set S of size $|S| = \text{cvd}(G)$, provided such a set exists, (ii) compute $\omega(G)$ by guessing the subset $S' \subseteq S$ which is the intersection of a maximum clique with S , and then consider for each cluster C of $G - S$ the graph induced by the vertices of C and S' ; the considered maximum clique is then composed of S' as well as the vertices of a cluster C that are adjacent to all the vertices of S' .

Proof overview. Our proof is a generalization of the FPT result for vertex cover number from [23]. The general idea for obtaining an ILP encoding of BANDWIDTH given a vertex cover S is to augment S by a small number (dependent only on the vertex cover number) of representative vertices of every neighborhood-type. It can be easily seen that we can modify any ordering π in such a way that the leftmost and rightmost neighbor of any vertex in S is contained in this augmented set S' without increasing the stretch. For any ordering σ of S' we can decide whether we can extend σ to an ordering of $V(G)$ of stretch at most b by encoding how vertices of certain neighborhood-types are distributed into the gaps between the vertices of S' into an ILP. By ensuring that we distribute the vertices in such a way that the leftmost and rightmost neighbor of any vertex in S is contained in S' we can bound the stretch of every edge by using one linear constraint for every edge in $G[S']$.

In our setting we can use the vertex cover approach to bound the stretch of all edges incident to the deletion set. To gain control over the stretch of edges within clusters, we show that we can convert any ordering into a *nice* one without increasing the stretch. Here niceness intuitively means, that we can order the vertices in between any two vertices of S' in such a way that vertices of the same type appear consecutively, where the type now depends on the isomorphism-type of the cluster union the deletion set. This will allow us to bound the stretch of such edges by a linear constraint as well.

3.1 Types and Buckets

Throughout this subsection we let G be a graph and S a cluster deletion set of G of size k . For any vertex $v \in V(G - S)$, let $N_S(v) = N(v) \cap S$ be its S -neighborhood.

In the following we define types of clusters. Cluster-types capture the isomorphism-type of the union of a cluster with the deletion set S . As clusters of the same size are isomorphic, our notion of type only needs to capture how the vertices of the cluster interact with the deletion set. In particular, we define cluster-types to be vectors such that each entry specifies the number of vertices in the cluster which have a particular S -neighborhood; in that case, two clusters are of the same cluster-type if and only if they agree on the number of their vertices with S -neighborhood N , for all $N \subseteq S$. We define this formally as follows. Let $\mathcal{K} \subseteq \mathbb{N}^{2^k}$ be the set of non-negative integer vectors κ with 2^k entries for which $\|\kappa\|_1 \leq \omega(G)$.¹ We assume that the entries of the vectors in \mathcal{K} are indexed by the subsets of S . We say that a cluster C has *cluster-type* $\kappa \in \mathcal{K}$ if $|\{v \in V(C) : N_S(v) = N\}| = (\kappa)_N$ for every $N \subseteq S$ where $(\kappa)_N$ denotes the entry of κ corresponding to N . We further let $\#\kappa$ denote the number of clusters of cluster-type κ in G . We say that a set \mathcal{C} of clusters is *representative* if it consists of $\min\{2|S|, \#\kappa\}$ distinct clusters of type κ for every cluster-type κ . We further say that a set S' is an *extended deletion set* if $S' = S \cup \bigcup_{C \in \mathcal{C}} V(C)$ for a representative set \mathcal{C} of clusters.

Lemma 3 *For every extended deletion set S' there is an ordering $\pi : V(G) \rightarrow [n]$ such that $\text{stretch}(\pi) = \text{bw}(G)$ and for every $s \in S$, the set S' contains vertices v_{\min}^s and v_{\max}^s , where $\pi(v_{\min}^s) = \min(\pi(N(s)))$ and $\pi(v_{\max}^s) = \max(\pi(N(s)))$, i.e., S' contains the leftmost and rightmost neighbor of s .*

Proof Let S' be an extended deletion set, where $S' = S \cup \bigcup_{C \in \mathcal{C}} V(C)$ for some representative cluster set \mathcal{C} , and π an ordering of $V(G)$ such that $\text{stretch}(\pi) = \text{bw}(G)$. Notice that since $|\{v_{\min}^s, v_{\max}^s : s \in S\}| \leq 2|S|$, it follows that there are at most $2|S|$ vertices among them not belonging to S and belonging to clusters of the same cluster-type.

Let $s \in S$ and suppose that $v^s \in \{v_{\min}^s, v_{\max}^s\}$ does not belong to S' . Let $C \notin \mathcal{C}$ denote the cluster of $G - S'$ that contains v^s , and assume that it is of cluster-type κ . In that case, it follows that $\#\kappa > 2|S|$ and that there exists a cluster $C' \in \mathcal{C}$ of cluster-type κ , such that $V(C') \cap \{v_{\min}^s, v_{\max}^s : s \in S\} = \emptyset$. Exchanging the vertices of C with vertices of C' respecting their S -neighborhood yields an automorphism $f : V(G) \rightarrow V(G)$. For $\pi' = \pi \circ f$, it holds that $\text{stretch}(\pi') = \text{bw}(G)$ due to Remark 1, while $v^s \in S'$. By repeatedly applying the previous argument, the statement follows. \square

We say that an ordering $\pi : V(G) \rightarrow [n]$ is *S' -extremal* if the second property in Lemma 3 is satisfied for π .

Observation 4 *Since there are at most $\omega(G)2^{|S| \cdot \omega(G)}$ different cluster-types, it holds that the size of any extended deletion set S' is at most $|S| + 2|S|\omega(G) \cdot \omega(G)2^{|S| \cdot \omega(G)}$.*

Let \mathcal{C} be a representative set of clusters, $S' = S \cup \bigcup_{C \in \mathcal{C}} V(C)$ the extended deletion set containing vertices from \mathcal{C} and S , and $k' = |S'|$. A *bucket distribution* of S' is a partition $\mathcal{B} = (B_0, \dots, B_{k'})$ of the vertices of $G - S'$. Fix a bucket distribution $\mathcal{B} = (B_0, \dots, B_{k'})$ of S' . We call the subsets B_i *buckets* of \mathcal{B} .

¹ Here $\|\cdot\|_1$ denotes the 1-norm, i.e., the sum of the absolute values of the entries.

Having fixed a bucket distribution, we define a more refined notion of types of clusters. In this refined notion, which we call distribution-type, we specify how many vertices of a cluster with a particular S -neighborhood are contained in a particular bucket. We define this formally in the following. Let $\mathcal{T} \subseteq \mathbb{N}^{2^k \times (k'+1)}$ be the set of matrices τ with $\|\tau\|_1 \leq \omega(G)$. We assume that the rows of matrices are indexed with subsets of S and the columns with $[0, k']$. We say that a cluster $C \notin \mathcal{C}$ has *distribution-type* $\tau \in \mathcal{T}$ in \mathcal{B} if $|\{v \in V(C) \cap B_i : N_S(v) = N\}| = (\tau)_{N,i}$ for every $N \subseteq S$ and every $i \in [0, k']$. For every $\kappa \in \mathcal{K}$, let $\mathcal{T}_\kappa \subseteq \mathcal{T}$ denote the set of distribution-types τ such that $\sum_{i \in [0, k']} (\tau)_{N,i} = (\kappa)_N$ for every $N \subseteq S$, i.e., the set of $\tau \in \mathcal{T}$ such that any cluster of distribution-type τ has cluster-type κ .

Observation 5 *The number of distribution-types is at most $\omega(G)^{2^{|S|} \cdot (|S'|+1)}$ for any extended deletion set S' .*

Let $\sigma : S' \rightarrow [k']$ be an ordering of the vertices of S' . We say that a vertex ordering $\pi : V(G) \rightarrow [n]$ is *compatible* with σ if for any $s_1, s_2 \in S'$ it holds that $\pi(s_1) < \pi(s_2)$ if and only if $\sigma(s_1) < \sigma(s_2)$. We say that a vertex ordering $\pi : V(G) \rightarrow [n]$ is *compatible* with σ and \mathcal{B} if π is compatible with σ and $B_0 = \{v \in V(G) : \pi(v) < \pi(\sigma^{-1}(1))\}$, $B_{k'} = \{v \in V(G) : \pi(v) > \pi(\sigma^{-1}(k'))\}$ and $B_i = \{v \in V(G) : \pi(\sigma^{-1}(i)) < \pi(v) < \pi(\sigma^{-1}(i+1))\}$ for $i \in [k' - 1]$.

3.2 Nice Orderings

Throughout this subsection we let G be a graph and S a cluster deletion set of G . Furthermore, for this subsection we let \mathcal{C} be a representative set of clusters, $S' = S \cup \bigcup_{C \in \mathcal{C}} V(C)$ the extended deletion set containing vertices from \mathcal{C} and S , and $k' = |S'|$. Additionally, we fix a bucket distribution $\mathcal{B} = (B_0, \dots, B_{k'})$ of S' and an ordering $\sigma : S' \rightarrow [k']$.

To obtain our nice ordering we use a series of exchange arguments that will not increase the stretch. We call an ordering *nice* if it has Properties (Π_1) , (Π_2) , and (Π_3) . We will first give some intuition regarding the properties, before defining them formally.

Assume that $\pi : V(G) \rightarrow [n]$ is an optimal ordering minimizing the number of edges of maximum stretch and assume that π is compatible with σ and \mathcal{B} . Furthermore, let $v \in V(C)$ be a vertex which is contained in an edge of maximum stretch with regard to π and the cluster C containing v is distributed over more than one bucket. In this case, v must be either the leftmost or the rightmost vertex of C . Assuming $v \in B_i$ is the leftmost vertex of C (the other case is analogous), we can observe that every vertex $v' \in B_i$ appearing further to the right than v must have a neighbor contained in a bucket to the right of B_i and no neighbor to the left of v . Otherwise, we can reduce the stretch of the edge containing v without increasing the stretch of any edge incident to v' (and hence reducing the number of edges of maximum stretch without increasing the maximum stretch) by exchanging v and v' . Therefore, we can assume that each bucket is partitioned into a left, a middle, and a right part and every vertex having neighbors only to the buckets to the left (respectively, right) of B_i appears in

the left (respectively, right) part. Additionally, this allows us to assume that within each bucket the vertices of one cluster appear consecutively (Property (Π_1)).

Now assume that $\{v, w\}$ is an edge of maximum stretch as before (v appears left of w in π) and $\{v', w'\}$ is another edge such that v' appears in the same bucket as v and w' in the same bucket as w . If v' appears before v then w' has to appear before w as $\{v, w\}$ is of maximum stretch. On the other hand, if v' appears after v then w' must appear after w as otherwise exchanging w and w' either reduces the number of edges of maximum stretch or reduces the maximum stretch itself. Hence, we can assume that the relative order of the leftmost vertices of a set of clusters is the same as the relative order of the rightmost vertices of the same clusters (Property (Π_2)).

Lastly, assume that C and C' are clusters of type $\tau \in \mathcal{T}$ which are not contained in just one bucket and appear next to each other (in their leftmost bucket). Assume B_ℓ is the bucket containing the leftmost vertex of C and C' and B_r the bucket containing the rightmost vertex of C and C' . We can essentially exchange $V(C) \cap B_\ell$ with $V(C') \cap B_\ell$ and at the same time $V(C) \cap B_r$ with $V(C') \cap B_r$ if certain properties about the size of these sets hold. This allows us to order the buckets in such a way, that clusters whose intersection with the leftmost (rightmost, respectively) bucket they intersect is of the same size, appear consecutively (Property (Π_3)).

To state Properties (Π_1) , (Π_2) , and (Π_3) formally we use the following notation. For a distribution-type $\tau \in \mathcal{T}$ and $i \in [0, k']$, we write τ_i to denote the column of τ which is indexed by i . We define $\text{LB}(\tau)$ to be the largest index $i \in [0, k']$ such that $\|\tau_j\|_1 = 0$ for any $j \in [0, i - 1]$, i.e., B_i is the leftmost bucket containing vertices from clusters of type τ . We define $\text{RB}(\tau)$ analogously to be the minimum index $i \in [0, k']$ such that $\|\tau_j\|_1 = 0$ for any $j \in [i + 1, k']$. Additionally, we let $\#\text{L}(\tau)$ be $\|\tau_{\text{LB}(\tau)}\|_1$ and $\#\text{R}(\tau)$ be $\|\tau_{\text{RB}(\tau)}\|_1$. For every $\ell \leq r \in [0, k']$ and every $n_L, n_R \in [0, \omega(G)]$, we define

$$\mathcal{T}^{(\ell, r, n_L, n_R)} = \{\tau \in \mathcal{T} : \text{LB}(\tau) = \ell, \text{RB}(\tau) = r, \#\text{L}(\tau) = n_L, \#\text{R}(\tau) = n_R\}.$$

Definition 6 (Property (Π_1)) We say that an S' -extremal ordering $\pi : V(G) \rightarrow [n]$ which is compatible with σ and \mathcal{B} has *Property (Π_1)* if for every $i \in [0, k']$

1. the vertices of $V(C) \cap B_i$ appear consecutively in π for every cluster $C \notin \mathcal{C}$,
2. we can partition the interval $\pi(B_i)$ into three (possibly empty) intervals $I_R^i < I_M^i < I_L^i$ such that for every $\tau \in \mathcal{T}$ and every cluster C of distribution-type τ
 - $\pi(V(C) \cap B_i) \subseteq I_R^i$ if $\text{LB}(\tau) \neq i$ and $\text{RB}(\tau) = i$,
 - $\pi(V(C) \cap B_i) \subseteq I_L^i$ if $\text{LB}(\tau) = i$ and $\text{RB}(\tau) \neq i$,
 - $\pi(V(C) \cap B_i) \subseteq I_M^i$ if either $\text{LB}(\tau) \neq i$ and $\text{RB}(\tau) \neq i$, or $\text{LB}(\tau) = \text{RB}(\tau) = i$.

Notice that while I_R^i contains the leftmost ordered vertices of B_i , we use the index R since those vertices are the rightmost vertices of their corresponding cliques. Analogously, we use I_L^i for the rightmost ordered vertices of B_i .

Lemma 7 Given an S' -extremal ordering $\pi : V(G) \rightarrow [n]$ which is compatible with σ and \mathcal{B} , there exists an S' -extremal ordering $\pi' : V(G) \rightarrow [n]$ with $\text{stretch}(\pi') \leq \text{stretch}(\pi)$ which is compatible with σ and \mathcal{B} and has *Property (Π_1)* .

Proof Let $\pi : V(G) \rightarrow [n]$ be an S' -extremal ordering of $V(G)$ which is compatible with σ and \mathcal{B} . For every $i \in [0, k']$, let $R_i \subseteq B_i$ contain the vertices that belong to the clusters of distribution-type τ , for which it holds that $\text{LB}(\tau) \neq i$ and $\text{RB}(\tau) = i$. Additionally, let $L_i \subseteq B_i$ contain the vertices that belong to the clusters of distribution-type τ , for which it holds that $\text{LB}(\tau) = i$ and $\text{RB}(\tau) \neq i$. Lastly, let $M_i = B_i \setminus (L_i \cup R_i)$, i.e., M_i contains the vertices of B_i that belong to the clusters of distribution-type τ , for which it holds that either $\text{LB}(\tau) \neq i$ and $\text{RB}(\tau) \neq i$, or $\text{LB}(\tau) = \text{RB}(\tau) = i$.

Fix an $i \in [0, k']$, and for every cluster C with $V(C) \cap B_i \neq \emptyset$, let $v_{\min}^{C,i}$ and $v_{\max}^{C,i}$ denote its leftmost and rightmost vertex in B_i on ordering π , i.e., $\pi(v_{\min}^{C,i}) = \min(\pi(V(C) \cap B_i))$ and $\pi(v_{\max}^{C,i}) = \max(\pi(V(C) \cap B_i))$.

We first define an ordering of the vertices of R_i . Let C_1, \dots, C_ℓ denote the cliques of $G - S'$ that have vertices in R_i , where $\max(\pi(V(C_j) \cap R_i)) < \max(\pi(V(C_{j+1}) \cap R_i))$, for all $j \in [\ell - 1]$. Then, define $\pi_{R_i} : R_i \rightarrow [|R_i|]$ as follows:

- $\pi_{R_i}(V(C_1) \cap R_i) = [|C_1|]$,
- for all $j \in [2, \ell]$, let $\pi_{R_i}(V(C_j) \cap R_i) = [\sum_{k=1}^{j-1} |V(C_k) \cap R_i| + 1, \sum_{k=1}^j |V(C_k) \cap R_i|]$,

where the ordering of the vertices of clique C_j in π_{R_i} is the same as in π , thus $v_{\max}^{C_j,i}$ denotes the rightmost vertex of C_j in π_{R_i} .

Next, we define an ordering of the vertices of L_i . Let C_1, \dots, C_r denote the cliques of $G - S'$ that have vertices in L_i , where $\min(\pi(V(C_j) \cap L_i)) < \min(\pi(V(C_{j+1}) \cap L_i))$, for all $j \in [\ell - 1]$. Then, define $\pi_{L_i} : L_i \rightarrow [|L_i|]$ as follows:

- $\pi_{L_i}(V(C_1) \cap L_i) = [|C_1|]$,
- for all $j \in [2, r]$, let $\pi_{L_i}(V(C_j) \cap L_i) = [\sum_{k=1}^{j-1} |V(C_k) \cap L_i| + 1, \sum_{k=1}^j |V(C_k) \cap L_i|]$,

where the ordering of the vertices of clique C_j in π_{L_i} is the same as in π , thus $v_{\min}^{C_j,i}$ denotes the leftmost vertex of C_j in π_{L_i} .

Intuitively, ordering π_{R_i} (respectively, π_{L_i}) is obtained by “sliding” all the non-rightmost (respectively, non-leftmost) vertices of cliques next to the rightmost (respectively, leftmost) vertex of the corresponding clique.

Lastly, define ordering $\pi_{M_i} : M_i \rightarrow [|M_i|]$ such that for every clique having vertices in M_i , its vertices appear consecutively in π_{M_i} , for some arbitrary ordering of the cliques.

Consider the ordering $\pi' : V(G) \rightarrow [|V(G)|]$, such that $\pi'(s) = \pi(s)$, for all $s \in S'$, while for every $i \in [0, k']$ and every $v \in B_i$, it holds that

- $\pi'(v) = i + \sum_{j=0}^{i-1} |B_j| + \pi_{R_i}(v)$, if $v \in R_i$,
- $\pi'(v) = i + \sum_{j=0}^{i-1} |B_j| + |R_i| + \pi_{M_i}(v)$, if $v \in M_i$,
- $\pi'(v) = i + \sum_{j=0}^{i-1} |B_j| + |R_i| + |M_i| + \pi_{L_i}(v)$, if $v \in L_i$.

Let C_1 and C_2 be cliques such that $V(C_1) \cap R_i \neq \emptyset$ while $V(C_2) \cap L_i \neq \emptyset$. Notice that $\max(\pi'(V(C_1) \cap B_i)) = \pi'(v_{\max}^{C_1,i})$, while $\pi'(v_{\max}^{C_1,i}) \leq \pi(v_{\max}^{C_1,i})$, since for any vertex $v \in B_i$ such that $\pi'(v) < \pi'(v_{\max}^{C_1,i})$, it holds that $\pi(v) < \pi(v_{\max}^{C_1,i})$. In an analogous way, it follows that $\min(\pi'(V(C_2) \cap B_i)) = \pi'(v_{\min}^{C_2,i})$, while $\pi(v_{\min}^{C_2,i}) \leq \pi'(v_{\min}^{C_2,i})$, since for any vertex $v \in B_i$ such that $\pi'(v_{\min}^{C_2,i}) < \pi'(v)$, it holds that $\pi(v_{\min}^{C_2,i}) < \pi(v)$.

It holds that π' has Property (Π_1) , it is compatible with σ and \mathcal{B} , and since π is S' -extremal, that is also the case for π' . It remains to argue that $\text{stretch}(\pi') \leq \text{stretch}(\pi)$. First, notice that since π' is S' -extremal, it suffices to argue about the edges belonging to cliques of $G - S'$. Let C be a cluster of distribution-type τ , and let $e = \{v_{\min}^{C,i_1}, v_{\max}^{C,i_2}\} \in E(C)$ denote the edge of $E(C)$ of maximum stretch on ordering π , for some $i_1, i_2 \in [0, k']$. If all the vertices of C are contained in a single bucket, then it holds that $i_1 = i_2 = i$, i.e., $V(C) \subseteq B_i$ for some $i \in [0, k']$. Since the vertices of C appear consecutively in π' , it follows that $\text{stretch}_{\pi'}(e) = |C| - 1$, which is minimum. Alternatively, notice that e is the edge of $E(C)$ of maximum stretch also in π' , while $v_{\min}^{C,i_1} \in L_{i_1}$ and $v_{\max}^{C,i_2} \in R_{i_2}$. Then, $\text{stretch}_{\pi'}(e) = \pi'(v_{\max}^{C,i_2}) - \pi'(v_{\min}^{C,i_1}) \leq \pi(v_{\max}^{C,i_2}) - \pi(v_{\min}^{C,i_1}) = \text{stretch}_{\pi}(e)$ follows. \square

Definition 8 (*Property (Π_2)*) We say that an S' -extremal ordering $\pi : V(G) \rightarrow [n]$ which is compatible with σ and \mathcal{B} has *Property (Π_2)* if for any two distribution-types $\tau, \tau' \in \mathcal{T}$ and any two clusters C and C' of distribution-type τ and τ' respectively, the following holds.

- If either $\text{LB}(\tau) = \text{LB}(\tau')$ or $\text{RB}(\tau) = \text{RB}(\tau')$, then for any $v \in V(C) \cap B_{\text{LB}(\tau)}$, $v' \in V(C') \cap B_{\text{LB}(\tau')}$, $w \in V(C) \cap B_{\text{RB}(\tau)}$, $w' \in V(C') \cap B_{\text{RB}(\tau')}$ we have that $\pi(v) < \pi(v')$ if and only if $\pi(w) < \pi(w')$.

Lemma 9 *Given an S' -extremal ordering $\pi : V(G) \rightarrow [n]$ which is compatible with σ and \mathcal{B} , there exists an S' -extremal ordering $\pi' : V(G) \rightarrow [n]$ with $\text{stretch}(\pi') \leq \text{stretch}(\pi)$ which is compatible with σ and \mathcal{B} and has Properties (Π_1) and (Π_2) .*

Proof Let $\pi : V(G) \rightarrow [n]$ be an S' -extremal ordering of $V(G)$ which is compatible with σ and \mathcal{B} . By Lemma 7 we can assume that π has Property (Π_1) . For every clique C , let v_{\min}^C and v_{\max}^C denote its leftmost and rightmost vertex in ordering π , i.e., $\pi(v_{\min}^C) = \min(\pi(V(C)))$ and $\pi(v_{\max}^C) = \max(\pi(V(C)))$.

Fix an $\ell \in [0, k']$ and let C and C' be two clusters of distribution-type τ and τ' respectively, such that

- $\text{LB}(\tau) = \text{LB}(\tau') = \ell$,
- $\pi(V(C') \cap B_\ell) = [\max(\pi(V(C) \cap B_\ell)) + 1, \max(\pi(V(C) \cap B_\ell)) + |V(C') \cap B_\ell|]$, i.e., the vertices of $V(C') \cap B_\ell$ appear right after the vertices of $V(C) \cap B_\ell$ in the ordering π ,
- $\pi(V(C') \cap B_{\text{RB}(\tau')}) < \pi(V(C) \cap B_{\text{RB}(\tau)})$.

Notice that it holds that $\text{RB}(\tau) \neq \ell$ and $\text{RB}(\tau') \neq \ell$, since otherwise π does not have Property (Π_1) . Let $\pi_{C \leftrightarrow C'} : V(G) \rightarrow [|V(G)|]$ be the ordering obtained by “sliding” the vertices of $V(C') \cap B_\ell$ to the left of the vertices of $V(C) \cap B_\ell$, i.e.,

- $\pi_{C \leftrightarrow C'}(v) = \pi(v)$, for all $v \notin (V(C) \cup V(C')) \cap B_\ell$,
- $\pi_{C \leftrightarrow C'}(v) = \pi(v) - \#L(\tau)$, for all $v \in V(C') \cap B_\ell$,
- $\pi_{C \leftrightarrow C'}(v) = \pi(v) + \#L(\tau')$, for all $v \in V(C) \cap B_\ell$,

where the ordering of the vertices in every clique C remains as in π , thus for v_{\min}^C and v_{\max}^C it holds that $\pi_{C \leftrightarrow C'}(v_{\min}^C) = \min(\pi_{C \leftrightarrow C'}(V(C)))$ and $\pi_{C \leftrightarrow C'}(v_{\max}^C) = \max(\pi_{C \leftrightarrow C'}(V(C)))$. Notice that due to Property (Π_1) , for all vertices v for which

$\pi_{C \leftrightarrow C'}(v) \neq \pi(v)$, it holds that $\pi(v) \in I_L^\ell$. It holds that $\pi_{C \leftrightarrow C'}$ has Property (Π_1) , it is compatible with σ and \mathcal{B} , and since π is S' -extremal, that is also the case for $\pi_{C \leftrightarrow C'}$. We argue that $\text{stretch}(\pi_{C \leftrightarrow C'}) \leq \text{stretch}(\pi)$. Notice that it suffices to argue about the stretches of edges in $E(C) \cup E(C')$. Let $e = \{v_{\min}^C, v_{\max}^C\}$ and $e' = \{v_{\min}^{C'}, v_{\max}^{C'}\}$ denote the edge of maximum stretch (in both orderings π and $\pi_{C \leftrightarrow C'}$) of $E(C)$ and $E(C')$ respectively. Since $\pi(v_{\min}^C) < \pi_{C \leftrightarrow C'}(v_{\min}^C)$, while $\pi(v_{\max}^C) = \pi_{C \leftrightarrow C'}(v_{\max}^C)$, it follows that $\text{stretch}_{\pi_{C \leftrightarrow C'}}(e) < \text{stretch}_\pi(e)$. As for e' , it holds that $\pi_{C \leftrightarrow C'}(v_{\min}^{C'}) = \pi(v_{\min}^{C'})$, while $\pi_{C \leftrightarrow C'}(v_{\max}^{C'}) < \pi(v_{\max}^{C'})$, therefore $\text{stretch}_{\pi_{C \leftrightarrow C'}}(e') < \text{stretch}_\pi(e')$ follows.

In an analogous way, one can prove that for any pair of clusters C and C' of distribution-type τ and τ' respectively, where

- $\text{RB}(\tau) = \text{RB}(\tau') = r$ and
- the vertices of $V(C') \cap B_r$ appear right after the vertices of $V(C) \cap B_r$,

there exists an S' -extremal ordering that has Property (Π_1) , it is compatible with σ and \mathcal{B} , is of no bigger stretch, reorders only vertices v for which $\pi(v) \in I_R^r$, and the vertices of $V(C) \cap B_{\text{LB}(\tau)}$ appear before those of $V(C') \cap B_{\text{LB}(\tau')}$.

By exhaustively applying both arguments, and due to the transitivity of $<$, the statement follows. □

Lastly, we want the buckets to be ordered by distribution-types which will enable us to express the stretch within clusters by linear constraints. To achieve this, we define two orderings of distribution-types, dictating in which order (in a nice, optimal vertex ordering) cliques of a certain type will appear within a bucket. First, let $\mathcal{T}_R^i = \bigcup_{\substack{\ell \in [0, i-1], \\ n_L, n_R \in [\omega(G)]}} \mathcal{T}^{(\ell, i, n_L, n_R)}$ and define the ordering $\rho_i : \mathcal{T}_R^i \rightarrow [|\mathcal{T}_R^i|]$ in the following way. For any $\tau \in \mathcal{T}^{(\ell, i, n_L, n_R)}$, $\tau' \in \mathcal{T}^{(\ell', i, n'_L, n'_R)}$, we have that $\rho_i(\tau) < \rho_i(\tau')$ if either

- $\ell < \ell'$ or
- $\ell = \ell'$, $n_L \geq n_R$ and $n'_L < n'_R$ or
- $\ell = \ell'$, $n_L \geq n_R$, $n'_L \geq n'_R$ and $n_R < n'_R$ or
- $\ell = \ell'$, $n_L < n_R$, $n'_L < n'_R$ and $n_L > n'_L$ or
- $\ell = \ell'$, $n_L \geq n_R$, $n'_L \geq n'_R$, $n_R = n'_R$ and $\tau \leq_{\text{lex}} \tau'$ or
- $\ell = \ell'$, $n_L < n_R$, $n'_L < n'_R$, $n_L = n'_L$ and $\tau \leq_{\text{lex}} \tau'$.

Here \leq_{lex} refers to the lexicographic order on matrices in \mathcal{T} where we read the entries by lines top to bottom. However, we can replace this by any total ordering (\leq_{lex} is an arbitrary choice).

Moreover, let $\mathcal{T}_L^i = \bigcup_{\substack{r \in [i+1, k'], \\ n_L, n_R \in [\omega(G)]}} \mathcal{T}^{(i, r, n_L, n_R)}$ and define the ordering $\lambda_i : \mathcal{T}_L^i \rightarrow [|\mathcal{T}_L^i|]$ by letting $\lambda_i(\tau) < \lambda_i(\tau')$ for any $\tau \in \mathcal{T}^{(i, r, n_L, n_R)}$, $\tau' \in \mathcal{T}^{(i, r', n'_L, n'_R)}$ if either

- $r < r'$ or
- $r = r'$ and $\rho_i(\tau) < \rho_i(\tau')$.

Remark 10 Note that we can compute all ρ_i and λ_i in time quadratic in the size of \mathcal{T} .

Definition 11 (*Property (Π_3)*) We say that an S' -extremal ordering $\pi : V(G) \rightarrow [n]$ which is compatible with σ and \mathcal{B} has *Property (Π_3)* if for every $i \in [0, k']$ we can

partition the interval $\pi(B_i)$ into (possibly empty) intervals

$$J_R^{(i,1)} < \dots < J_R^{(i,|\mathcal{T}_R^i|)} < J_M^i < J_L^{(i,1)} < \dots < J_L^{(i,|\mathcal{T}_L^i|)}$$

such that for every distribution-type $\tau \in \mathcal{T}$ and every cluster C of type τ and every $j \in [|\mathcal{T}_R^i|]$, $j' \in [|\mathcal{T}_L^i|]$,

- $\pi(V(C) \cap B_i) \subseteq J_R^{(i,j)}$ if $\rho_i(\tau) = j$ and
- $\pi(V(C) \cap B_i) \subseteq J_L^{(i,j')}$ if $\lambda_i(\tau) = j'$.

Lemma 12 *Given an S' -extremal ordering $\pi : V(G) \rightarrow [n]$ which is compatible with σ and \mathcal{B} , there exists an S' -extremal ordering $\pi' : V(G) \rightarrow [n]$ with $\text{stretch}(\pi') \leq \text{stretch}(\pi)$ which is compatible with σ and \mathcal{B} and has Properties (Π_1) , (Π_2) , and (Π_3) .*

Proof Let $\pi : V(G) \rightarrow [n]$ be an S' -extremal ordering which is compatible with σ and \mathcal{B} . By Lemma 9 we can assume that π has Properties (Π_1) and (Π_2) . For every $i \in [0, k']$ let $I_R^i < I_M^i < I_L^i$ be the partition of $\pi(B_i)$ as in Property (Π_1) . Since π has Property (Π_2) it holds that for any fixed $i \in [0, k']$ we can partition the interval I_R^i into intervals $I_R^{(i,0)} < \dots < I_R^{(i,i-1)}$ such that $\pi(V(C) \cap B_i) \subseteq I_R^{(i,j)}$ if C is of distribution-type τ , $\text{RB}(\tau) = i$, and $\text{LB}(\tau) = j$. Equivalently, we can partition I_L^i into intervals $I_L^{(i,i+1)} < \dots < I_L^{(i,k')}$ such that $\pi(V(C) \cap B_i) \subseteq I_L^{(i,j)}$ if C is of distribution-type τ , $\text{LB}(\tau) = i$, and $\text{RB}(\tau) = j$.

Fix any $\ell < r \in [0, k']$ and let τ, τ' be two distribution-types with $\text{LB}(\tau) = \text{LB}(\tau') = \ell$ and $\text{RB}(\tau) = \text{RB}(\tau') = r$. We further let C be a cluster of distribution-type τ and C' a cluster of distribution-type τ' . We say that C *directly precedes* C' if $\pi(V(C) \cap B_\ell) < \pi(V(C') \cap B_\ell)$ and $\pi(V(C) \cap B_\ell) \cup \pi(V(C') \cap B_\ell)$ is an interval. Note that if $\pi(V(C) \cap B_\ell) \cup \pi(V(C') \cap B_\ell)$ is an interval then so is $\pi(V(C) \cap B_r) \cup \pi(V(C') \cap B_r)$, due to Property (Π_2) . Moreover, note that $\pi(V(C) \cap B_i)$ is an interval for every clique C and $i \in [0, k']$ as π has Property (Π_1) . If C directly precedes C' we define a new ordering $\pi_{C \leftrightarrow C'} : V(G) \rightarrow [n]$ which corresponds to exchanging C and C' in B_ℓ and B_r as follows.

- $\pi_{C \leftrightarrow C'}(v) = \pi(v) - \#L(\tau)$ for $v \in V(C') \cap B_\ell$,
- $\pi_{C \leftrightarrow C'}(v) = \pi(v) + \#L(\tau')$ for $v \in V(C) \cap B_\ell$,
- $\pi_{C \leftrightarrow C'}(v) = \pi(v) - \#R(\tau)$ for $v \in V(C') \cap B_r$,
- $\pi_{C \leftrightarrow C'}(v) = \pi(v) + \#R(\tau')$ for $v \in V(C) \cap B_r$, and
- $\pi_{C \leftrightarrow C'}(v) = \pi(v)$ for any vertex $v \notin (V(C) \cup V(C')) \cap (B_\ell \cup B_r)$.

Claim 13 identifies when exchanging C and C' cannot increase the stretch.

Claim 13 *If one of the following holds, then $\text{stretch}(\pi_{C \leftrightarrow C'}) \leq \text{stretch}(\pi)$.*

1. $\#L(\tau) \leq \#R(\tau)$ and $\#L(\tau') \geq \#R(\tau')$.
2. $\#L(\tau) \geq \#R(\tau)$, $\#L(\tau') \geq \#R(\tau')$, and $\#R(\tau) \geq \#R(\tau')$.
3. $\#L(\tau) \leq \#R(\tau)$, $\#L(\tau') \leq \#R(\tau')$, and $\#L(\tau) \leq \#L(\tau')$.

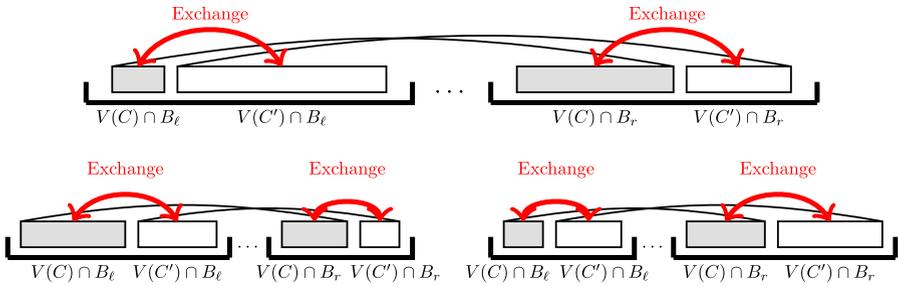


Fig. 2 Exchange arguments used in Claim 13

Proof First note that $\text{stretch}_\pi(e) = \text{stretch}_{\pi_{C \leftrightarrow C'}}(e)$ for any edge $e \notin E(C) \cup E(C')$. Let v_{\min}^C (respectively, v_{\max}^C) denote the leftmost (respectively, rightmost) vertex of $V(C)$ with regard to $\pi_{C \leftrightarrow C'}$, that is, $\pi(v_{\min}^C) = \min(\pi_{C \leftrightarrow C'}(V(C)))$ and $\pi(v_{\max}^C) = \max(\pi_{C \leftrightarrow C'}(V(C)))$. Define $v_{\min}^{C'}$ and $v_{\max}^{C'}$ analogously. Note that $\pi(v_{\min}^C) = \min(\pi(V(C)))$, $\pi(v_{\max}^C) = \max(\pi(V(C)))$, $\pi(v_{\min}^{C'}) = \min(\pi(V(C')))$, and $\pi(v_{\max}^{C'}) = \max(\pi(V(C')))$ as the order of vertices within C and C' is the same in π and in $\pi_{C \leftrightarrow C'}$. Note that $\text{stretch}_{\pi_{C \leftrightarrow C'}}(e) \leq \text{stretch}_{\pi_{C \leftrightarrow C'}}(\{v_{\min}^C, v_{\max}^C\})$ for any edge $e \in E(C)$ and $\text{stretch}_{\pi_{C \leftrightarrow C'}}(e') \leq \text{stretch}_{\pi_{C \leftrightarrow C'}}(\{v_{\min}^{C'}, v_{\max}^{C'}\})$ for any edge $e' \in E(C')$. Hence, we only consider the stretch of the two edges $\{v_{\min}^C, v_{\max}^C\}$ and $\{v_{\min}^{C'}, v_{\max}^{C'}\}$. Consider how the stretch of the edge $\{v_{\min}^C, v_{\max}^C\}$ is affected by the exchange of C and C' . Prior to the exchange, all $\#L(\tau')$ vertices of $V(C') \cap B_\ell$ contributed to the stretch of $\{v_{\min}^C, v_{\max}^C\}$ while after the exchange they do not. On the other hand, after the exchange of C and C' all $\#R(\tau')$ vertices in $V(C') \cap B_r$ contribute to the stretch of $\{v_{\min}^C, v_{\max}^C\}$ but prior to the exchange they did not. Other than that, the stretch of $\{v_{\min}^C, v_{\max}^C\}$ is not affected by the exchange (see Fig. 2 for an illustration). Hence, the difference between the stretch of $\{v_{\min}^C, v_{\max}^C\}$ before and after the exchange is $\#L(\tau') - \#R(\tau')$. In a very similar way, we can determine the difference between the stretch of edges before and after the exchange of C and C' to obtain the following:

- (i) $\text{stretch}_\pi(\{v_{\min}^C, v_{\max}^C\}) - \text{stretch}_{\pi_{C \leftrightarrow C'}}(\{v_{\min}^C, v_{\max}^C\}) = \#L(\tau') - \#R(\tau')$,
- (ii) $\text{stretch}_\pi(\{v_{\min}^{C'}, v_{\max}^{C'}\}) - \text{stretch}_{\pi_{C \leftrightarrow C'}}(\{v_{\min}^{C'}, v_{\max}^{C'}\}) = \#R(\tau) - \#L(\tau)$,
- (iii) $\text{stretch}_\pi(\{v_{\min}^C, v_{\max}^C\}) - \text{stretch}_{\pi_{C \leftrightarrow C'}}(\{v_{\min}^{C'}, v_{\max}^{C'}\}) = \#R(\tau) - \#R(\tau')$, and
- (iv) $\text{stretch}_\pi(\{v_{\min}^{C'}, v_{\max}^{C'}\}) - \text{stretch}_{\pi_{C \leftrightarrow C'}}(\{v_{\min}^C, v_{\max}^C\}) = \#L(\tau') - \#L(\tau)$.

Assuming $\#L(\tau) \leq \#R(\tau)$ and $\#L(\tau') \geq \#R(\tau')$, Equalities (i) and (ii) imply that $\text{stretch}(\pi_{C \leftrightarrow C'}) \leq \text{stretch}(\pi)$. Assuming $\#L(\tau) \geq \#R(\tau)$, $\#L(\tau') \geq \#R(\tau')$, and $\#R(\tau) \geq \#R(\tau')$ we obtain the statement using Equalities (i) and (iii). Lastly, assuming $\#L(\tau) \leq \#R(\tau)$, $\#L(\tau') \leq \#R(\tau')$, and $\#L(\tau) \leq \#L(\tau')$ we get the desired bound on the stretch using Equalities (ii) and (iv). For an illustration of the different cases see Fig. 2. \square

We now successively exchange clusters C of distribution-type τ with clusters C' of distribution-type τ' for any $\tau, \tau' \in \mathcal{T}$ with $\text{LB}(\tau) = \text{LB}(\tau') = \ell$ and $\text{RB}(\tau) = \text{RB}(\tau') = r$ whenever C is directly preceding C' and $\rho_\ell(\tau') < \rho_\ell(\tau)$. Call the resulting

ordering π' and note that in each step, the stretch does not increase by Claim 13 and hence $\text{stretch}(\pi') \leq \text{stretch}(\pi)$. Furthermore, since any exchange maintains \mathcal{B} , the ordering π' is compatible with σ and \mathcal{B} . Even more so, the exchanges maintain the partition of intervals $I_R^{(i,0)} < \dots < I_R^{(i,i-1)} < I_M^i < I_L^{(i,i+1)} < \dots < I_L^{(i,k')}$. By our exchange strategy we obtain a partition of $I_R^{(i,\ell)}$ for every $i \in [0, k']$, $\ell \in [0, i - 1]$ into intervals $J_R^{(i,\rho_i(\tau))}$ for every $\tau \in \bigcup_{n_L, n_R \in [\omega(G)]} \mathcal{T}^{(\ell, i, n_L, n_R)}$ such that $\pi(V(C) \cap B_i) \subseteq J_R^{(i,\rho_i(\tau))}$ if C is of type τ . Similarly, for every $i \in [0, k']$, $r \in [i + 1, k']$ we obtain a partition of $I_L^{(i,r)}$ into intervals $J_L^{(i,\lambda_i(\tau))}$ for every $\tau \in \bigcup_{n_L, n_R \in [\omega(G)]} \mathcal{T}^{(i, r, n_L, n_R)}$ such that $\pi(V(C) \cap B_i) \subseteq J_L^{(i,\lambda_i(\tau))}$ if C is of type τ . Furthermore, the exchange strategy guaranties that $J_R^{(i,\ell)} < J_R^{(i,\ell')}$ and $J_L^{(i,r)} < J_L^{(i,r')}$ for $\ell \leq \ell'$, $r \leq r'$. \square

3.3 ILP Formulation

Throughout this subsection we let G be a graph and S a cluster deletion set of G with $k = |S|$. Furthermore, for the remainder of this subsection we fix \mathcal{C} to be a representative set of clusters, $S' = S \cup \bigcup_{C \in \mathcal{C}} V(C)$ the extended deletion set containing vertices from \mathcal{C} and S , and $k' = |S'|$.

For every ordering $\sigma : S' \rightarrow [k']$, we will use an ILP to determine whether there is an S' -extremal ordering $\pi : V(G) \rightarrow [n]$ of stretch at most b which is compatible with σ . The ILP has two variables x_τ, y_τ for every distribution-type $\tau \in \mathcal{T}$. The variable x_τ expresses how many clusters of $G - S'$ have distribution-type τ in an optimal S' -extremal ordering compatible with σ . The variable y_τ is an indicator variable which is 1 if and only if $x_\tau > 0$ and 0 otherwise. We further use z_i for $i \in [0, k']$ in our ILP formulation as a placeholder for the expression $\sum_{\tau \in \mathcal{T}} (x_\tau \cdot \|\tau_i\|_1)$ which expresses the number of vertices in bucket i . For an assignment $\alpha : \{x_\tau, y_\tau : \tau \in \mathcal{T}\} \rightarrow \mathbb{N}$ of the variables of our ILP, we write $\alpha(z_i)$ to stand for the expression $\sum_{\tau \in \mathcal{T}} (\alpha(x_\tau) \cdot \|\tau_i\|_1)$.

We further need the leftmost and rightmost neighbor of any vertex of S in S' , thus define $v_{\min, \sigma}^s, v_{\max, \sigma}^s \in S'$ such that $\sigma(v_{\min, \sigma}^s) = \min(\sigma(N(s)))$ and $\sigma(v_{\max, \sigma}^s) = \max(\sigma(N(s)))$, for every ordering $\sigma : S' \rightarrow [k']$ and $s \in S$. Note that by choosing S to be minimum, we can assume that S contains no vertex with no neighbors in $G - S$ and hence $v_{\min, \sigma}^s$ and $v_{\max, \sigma}^s$ are well-defined.

For a fixed ordering $\sigma : S' \rightarrow [k']$, we can now formulate our set of linear constraints. The first three constraints ensure that we choose the number of clusters that have a certain distribution-type in a feasible way. That is, (T1) ensures that the quantities of distribution-types corresponding to an assignment of the variables x_τ correspond to a valid choice of allocating each available cluster in the input graph G a distribution-type. As for (T2), it ensures that $v_{\min, \sigma}^s$ is indeed the leftmost neighbor of s while $v_{\max, \sigma}^s$ is the rightmost neighbor of s for every $s \in S$ by ensuring that any distribution-type placing a neighbor of s in a bucket left of $v_{\min, \sigma}^s$ or right of $v_{\max, \sigma}^s$ does not occur. Finally, (T3) guarantees that y_τ indeed indicates whether or not distribution-type τ is used in the solution.

(T1) For every $\kappa \in \mathcal{K}$,

$$\#\kappa = \min\{\#\kappa, 2k\} + \sum_{\tau \in \mathcal{T}_\kappa} x_\tau.$$

(T2) For every $s \in S$ and every $\tau \in \mathcal{T}$ for which $\tau_{N,i} > 0$ for some $N \ni s$ and $i \in [0, \sigma(v_{\min,\sigma}^s) - 1] \cup [\sigma(v_{\max,\sigma}^s), k']$,

$$x_\tau = 0.$$

(T3) $x_\tau \cdot (1 - y_\tau) = 0$ and $(1 - x_\tau) \cdot y_\tau \leq 0$ for every $\tau \in \mathcal{T}$.

The purpose of all remaining constraints is to ensure that for the assignment of variables, which essentially corresponds to choosing a bucket distribution \mathcal{B} , there is an S' -extremal ordering $\pi : V(G) \rightarrow [n]$ which is compatible with σ and \mathcal{B} for which $\text{stretch}(\pi) \leq b$. (DS) expresses that the stretch of edges in $G[S']$ is bounded by b .

(DS) For every $s, s' \in S'$ with $\{s, s'\} \in E(G)$, $\sigma(s) < \sigma(s')$,

$$b \geq \sigma(s') - \sigma(s) + \sum_{i \in [\sigma(s), \sigma(s') - 1]} z_i.$$

The last three constraints deal with bounding the stretch of edges within clusters. For this we assume that the S' -extremal ordering which is consistent with σ and \mathcal{B} is nice, i.e., has Properties (Π_1) , (Π_2) , and (Π_3) . The first constraint (C1) is necessary to bound the stretch of clusters that are fully contained in one bucket. To bound the stretch of clusters contained in multiple buckets, we have one constraint for every distribution-type $\tau \in \mathcal{T}^{(\ell, r, n_L, n_R)}$ for any $\ell < r \in [0, k']$, $n_L, n_R \in [\omega(G)]$. By Property (Π_3) we know that there are intervals $J_L^{(\ell, \lambda_\ell(\tau))}$ containing all vertices from $B_\ell \cap V(C)$ and $J_R^{(r, \rho_r(\tau))}$ containing all vertices $B_r \cap V(C)$ for every cluster C of distribution-type τ . The trick now is to observe that if $n_L \geq n_R$ then the first cluster appearing in $J_L^{(\ell, \lambda_\ell(\tau))}$ observes the maximum stretch while if $n_L < n_R$ it is the last clique. Using this we can express with constraints (C2) and (C3) that the stretch of every cluster of distribution-type τ is bounded by b .

(C1) $b \geq \omega(G) - 1$.

(C2) For every $\ell < r \in [0, k']$, $n_L \geq n_R \in [\omega(G)]$, and $\tau \in \mathcal{T}^{(\ell, r, n_L, n_R)}$,

$$b \geq y_\tau \cdot \left(\sum_{\tau' \in \lambda_\ell^{-1}([\lambda_\ell(\tau), |\mathcal{T}_L^\ell|])} \#\mathbf{L}(\tau') \cdot x_{\tau'} + \sum_{\ell < i < r} z_i + (r - \ell) \right. \\ \left. + \sum_{\tau' \in \rho_r^{-1}([1, \rho_r(\tau) - 1])} \#\mathbf{R}(\tau') \cdot x_{\tau'} + n_R - 1 \right).$$

(C3) For every $\ell < r \in [0, k']$, $n_L < n_R \in [\omega(G)]$, and $\tau \in \mathcal{T}^{(\ell, r, n_L, n_R)}$,

$$\begin{aligned}
 b \geq & y_\tau \cdot \left(n_L + \sum_{\tau' \in \lambda_\ell^{-1}([\lambda_\ell(\tau)+1, |\mathcal{T}_L^\ell|])} \#L(\tau') \cdot x_{\tau'} + \sum_{\ell < i < r} z_i + (r - \ell) \right. \\
 & \left. + \sum_{\tau' \in \rho_r^{-1}([1, \rho_r(\tau)])} \#R(\tau') \cdot x_{\tau'} - 1 \right).
 \end{aligned}$$

Lemma 14 *For any ordering $\sigma : S' \rightarrow [k']$, there is an S' -extremal ordering $\pi : V(G) \rightarrow [n]$ of stretch at most b which is compatible with σ if and only if the system of linear equation (T1, T2, T3, DS, C1, C2, C3) for σ admits a solution.*

Proof Towards showing the forward direction, assume that there is an S' -extremal ordering $\pi : V(G) \rightarrow [n]$ which is compatible with σ and $\text{stretch}(\pi) \leq b$. Let $\mathcal{B} = (B_0, \dots, B_{k'})$ be the bucket distribution associated to π , i.e., $B_0 = \{v \in V(G) : \pi(v) < \pi(\sigma^{-1}(1))\}$, $B_{k'} = \{v \in V(G) : \pi(v) > \pi(\sigma^{-1}(k'))\}$, and $B_i = \{v \in V(G) : \pi(\sigma^{-1}(i)) < \pi(v) < \pi(\sigma^{-1}(i+1))\}$ for $i \in [k' - 1]$. By Lemma 12 there is an S' -extremal ordering $\pi' : V(G) \rightarrow [n]$ with $\text{stretch}(\pi') \leq \text{stretch}(\pi) \leq b$ which is compatible with σ and \mathcal{B} and has Properties (Π_1) , (Π_2) , and (Π_3) .

We now describe how to obtain an assignment $\alpha : \{x_\tau, y_\tau : \tau \in \mathcal{T}\} \rightarrow \mathbb{N}$ which yields a solution to the system of linear equations (T1, T2, T3, DS, C1, C2, C3). For every $\tau \in \mathcal{T}$ we set $\alpha(x_\tau)$ to be the number of clusters in $G - S'$ that have distribution-type τ in \mathcal{B} while we set $\alpha(y_\tau) = 0$ if $\alpha(x_\tau) = 0$ and $\alpha(y_\tau) = 1$ otherwise. We will argue in the following that this yields the desired solution.

Firstly, (T1) is satisfied since $\min\{\#\kappa, 2k\}$ is counting the clusters of cluster-type κ that are in \mathcal{C} while by choice of $\alpha(x_\tau)$ the sum $\sum_{\tau \in \mathcal{T}_\kappa} \alpha(x_\tau)$ is counting all clusters of cluster-type κ that are not contained in \mathcal{C} .

If (T2) was not satisfied then there is $s \in S$, $N \ni s$, $i \in [0, \sigma(v_{\min, \sigma}^s) - 1] \cup [\sigma(v_{\max, \sigma}^s), k']$, and $\tau \in \mathcal{T}$ such that $\tau_{N,i} > 0$ and $\alpha(x_\tau) > 0$. Hence there is a cluster C of distribution-type τ . Because $\tau_{N,i} > 0$ there is a vertex $c \in V(C)$ which is adjacent to s such that $c \in B_i$. But since $i \in [0, \sigma(v_{\min, \sigma}^s) - 1] \cup [\sigma(v_{\max, \sigma}^s), k']$ we get that either $\pi(c) < \min(\pi(N(s) \cap S'))$ or $\pi(c) > \max(\pi(N(s) \cap S'))$, which is a contradiction since π is S' -extremal. Hence (T2) is satisfied.

Furthermore, the constraint (T3) is clearly satisfied.

To show that (DS) is satisfied first observe that $\pi'(s) = \sigma(s) + \sum_{i \in [0, \sigma(s)-1]} z_i$ for every $s \in S'$ where $z_i = |B_i|$, that is the position of s in σ plus the size of all the buckets in between elements of S' that appear before s in σ . Hence for any $s, s' \in S$ with $\{s, s'\} \in E(G)$ and $\sigma(s) < \sigma(s')$ it holds that $\text{stretch}_{\pi'}(\{s, s'\}) = |\sigma(s) - \sigma(s')| + \sum_{i \in [\sigma(s), \sigma(s')-1]} z_i$. Since $\text{stretch}(\pi') \leq b$ this immediately implies (DS) is satisfied.

Next observe that (C1) is trivially satisfied as for any graph the bandwidth must be at least its clique number.

To prove (C2) is satisfied we use the nice ordering properties of π' . Let $J_R^{(i,1)} < \dots < J_R^{(i,|\mathcal{T}_R^i|)} < J_M^i < J_L^{(i,1)} < \dots < J_L^{(i,|\mathcal{T}_L^i|)}$ be the partition of $\pi'(B_i)$ for

which we have Property (Π_3) . Fix some $\ell < r \in [0, k']$, $n_L \leq n_R \in [\omega(G)]$, and $\tau \in \mathcal{T}^{(\ell, r, n_L, n_R)}$. If there is no cluster of distribution-type τ then $\alpha(x_\tau) = \alpha(y_\tau) = 0$ and (C2) is satisfied. Hence assume that there is at least one cluster of distribution-type τ . For every cluster C of distribution-type τ we let $v_{\min}^C, v_{\max}^C \in V(C)$ be the vertices such that $\pi'(v_{\min}^C) \leq \pi'(v) \leq \pi'(v_{\max}^C)$ for every $v \in V(C)$. Note that $v_{\min}^C \in B_\ell$ and $v_{\max}^C \in B_r$ as C has distribution-type τ and $\text{LB}(\tau) = \ell, \text{RB}(\tau) = r$. We let C be the cluster of distribution-type τ for which $\pi'(v_{\min}^C) < \pi'(v_{\min}^{C'})$ for any other cluster C' of distribution-type τ . We now let X be the set of all vertices $v \in V(G)$ such that $\pi'(v_{\min}^C) \leq \pi'(v) \leq \pi'(v_{\max}^C)$. By choice of C we know that $J_L^{(\ell, \lambda_\ell(\tau))} \subseteq X$ as this includes all vertices $V(C') \cap B_\ell$ for every cluster C' of distribution-type τ . Additionally, $J_L^{(\ell, j)} \subseteq X$ for all $j \in [\lambda_\ell(\tau) + 1, |\mathcal{T}_L^\ell|]$ according to the ordering of intervals $J_L^{(\ell, j)}$. Since $|J_L^{(\ell, \lambda_\ell(\tau))}| = \#\mathbf{L}(\tau') \cdot x_{\tau'}$ we conclude that $|X \cap B_\ell| \geq \sum_{\tau' \in \lambda_\ell^{-1}([\lambda_\ell(\tau), |\mathcal{T}_L^\ell|])} \#\mathbf{L}(\tau') \cdot x_{\tau'}$. On the other hand, $X \cap B_r$ contains every interval $J_R^{(r, i)}$ for which $i \in [1, \rho_r(\tau) - 1]$ since these are all intervals of bucket r preceding $J_R^{(r, \rho_r(\tau))}$. Additionally, $X \cap B_r$ contains the n_R vertices belonging to $V(C) \cap B_r$. Hence, $|X \cap B_r| \geq \sum_{\tau' \in \rho_r^{-1}([1, \rho_r(\tau) - 1])} \#\mathbf{R}(\tau') \cdot x_{\tau'} + n_R$. Finally, $X \setminus (B_\ell \cup B_r)$ consists of precisely the buckets and vertices from S' residing in between B_ℓ and B_r . This further gives us $|X \setminus (B_\ell \cup B_r)| = \sum_{\ell < i < r} z_i + (r - \ell)$. Since $\{v_{\min}^C, v_{\max}^C\} \in E(G)$ and $\text{stretch}_{\pi'}(\{v_{\min}^C, v_{\max}^C\}) = |X| - 1 \leq b$ the constraint (C2) is satisfied.

Validity of constraint (C3) can be shown with a symmetric argument in which we choose C to be the cluster with $\pi'(v_{\min}^C) > \pi'(v_{\min}^{C'})$ for any other cluster C' of distribution-type τ and then mirror the roles of the left and right bucket. This concludes the proof that we indeed find a solution of the system of linear equations (T1, T2, T3, DS, C1, C2, C3).

Towards showing the backwards direction, assume that the ILP admits a solution and let $\alpha : \{x_\tau, y_\tau : \tau \in \mathcal{T}\} \rightarrow \mathbb{N}$ be an assignment to the variables satisfying all constraints. We define an ordering $\pi : V(G) \rightarrow [n]$ in the following way. For every $\kappa \in \mathcal{K}$ we pick arbitrarily a partition $(P_\tau)_{\tau \in \mathcal{T}_\kappa}$ of the set of clusters of type κ excluding clusters in \mathcal{C} such that $|P_\tau| = \alpha(x_\tau)$. This is possible because (T1) is satisfied. We now pick a bucket distribution $\mathcal{B} = (B_0, \dots, B_{k'})$ of S' as follows. For every $\tau \in \mathcal{T}$ and every $C \in P_\tau$ we pick arbitrarily $\tau_{N, i}$ vertices v of C with $N_S(v) = N$ and add them to bucket i for every $i \in [0, k']$. This is possible since any clique allocated to P_τ for some $\tau \in \mathcal{T}_\kappa$ has cluster-type κ and therefore contains precisely $\kappa_N = \sum_{i \in [0, k']} \tau_{N, i}$ vertices v with $N_S(v) = N$ by definition of \mathcal{T}_κ . Clearly, every $C \in P_\tau$ has distribution-type τ in the bucket-distribution \mathcal{B} . We now obtain π by choosing an arbitrary ordering which is compatible with σ and \mathcal{B} , i.e., we order the vertices of S' as prescribed by σ and place vertices in B_i in between vertex $\sigma^{-1}(i)$ and $\sigma^{-1}(i + 1)$ while ordering the vertices in every B_i arbitrarily.

We first argue that the ordering π is S' -extremal. Recall that, for $s \in S$, $v_{\min, \sigma}^s$ and $v_{\max, \sigma}^s$ denote the leftmost and rightmost neighbor of $s \in S$ in S' , i.e., $\sigma(v_{\min, \sigma}^s) = \min(\sigma(N(s) \cap S'))$ and $\sigma(v_{\max, \sigma}^s) = \max(\sigma(N(s) \cap S'))$. Assume that π is not S' -extremal and let $v \in V(G) \setminus S'$ be a vertex such that for some $s \in S$ either $\pi(v) = \min(\pi(N(s)))$ or $\pi(v) = \max(\pi(N(s)))$, which implies that either $\pi(v) < \pi(v_{\min, \sigma}^s)$

or $\pi(v) > \pi(v_{\max,\sigma}^s)$ respectively. Let C be the cluster containing v , τ the distribution-type of C in \mathcal{B} , and $i \in [0, k']$ for which $v \in B_i$. By definition the existence of v and C of distribution-type τ implies that $\tau_{N_S(v),i} > 0$ and $\alpha(x_\tau) > 0$. Since (T2) is satisfied it follows that $i \in [\sigma(v_{\min,\sigma}^s), \sigma(v_{\max,\sigma}^s) - 1]$. Consequently, $\pi(v_{\min,\sigma}^s) < \pi(v) < \pi(v_{\max,\sigma}^s)$ since π is compatible with σ and \mathcal{B} which yields a contradiction.

Note that the ordering π does not necessarily have $\text{stretch}(\pi) \leq b$. But by Lemma 12 we can obtain an S' -extremal ordering $\pi': V(G) \rightarrow [n]$ which is compatible with σ and \mathcal{B} and has Properties (Π_1) , (Π_2) , and (Π_3) . We claim that π' satisfies that $\text{stretch}(\pi') \leq b$ and hence is the desired ordering.

To argue that $\text{stretch}(\pi') \leq b$ we first consider any edge $e = \{v, s\}$ for which $v \in V(G)$ and $s \in S$. Consider the case that $\pi'(v) < \pi'(s)$ (the other case can be argued in a symmetric way). Since π' is S' -extremal and v is adjacent to s we know that $\pi'(v_{\min,\sigma}^s) \leq \pi'(v)$. Hence

$$\text{stretch}_{\pi'}(e) \leq \text{stretch}_{\pi'}(\{v_{\min,\sigma}^s, s\}) = |\sigma(s) - \sigma(v_{\min,\sigma}^s)| + \sum_{i \in [\sigma(v_{\min,\sigma}^s), \sigma(s) - 1]} z_i \leq b,$$

where the last inequality is due to (DS).

Now consider any edge $e = \{u, w\} \in E(G)$ such that $u, w \notin S'$. Since $S' \supseteq S$, where S is a cluster deletion set, there is a cluster C in $G - S'$ such that $u, w \in V(C)$. Assume $\tau \in \mathcal{T}$ is the distribution-type of C and additionally let $\ell = \text{LB}(\tau)$, $r = \text{RB}(\tau)$, $n_L = \#L(\tau)$, and $n_R = \#R(\tau)$. We let $v_{\min}^C, v_{\max}^C \in V(C)$ be the two vertices with $\pi'(v_{\min}^C) \leq \pi'(v) \leq \pi'(v_{\max}^C)$ for every $v \in V(C)$. Since $\text{stretch}_{\pi'}(e) \leq \text{stretch}_{\pi'}(\{v_{\min}^C, v_{\max}^C\})$ it is sufficient to show that $\text{stretch}_{\pi'}(\{v_{\min}^C, v_{\max}^C\}) \leq b$. First assume that $\ell = r$, which implies that every vertex of C is contained in the same bucket. Due to Property (Π_1) it follows that all vertices of C appear consecutively in π' , therefore $\text{stretch}_{\pi'}(\{v_{\min}^C, v_{\max}^C\}) \leq |C| - 1 \leq \omega(G) - 1 \leq b$, where the last inequality follows from (C1).

Now consider the case where $\ell < r$. We use Property (Π_3) to obtain a bound on the stretch. Therefore, for every $i \in [0, k']$ let $J_R^{(i,1)} < \dots < J_R^{(i,|T_R^i|)} < J_M^i < J_L^{(i,1)} < \dots < J_L^{(i,|T_L^i|)}$ be the partition of $\pi'(B_i)$ as in Property (Π_3) . We let X be the set of all vertices $v \in V(G)$ such that $\pi'(v_{\min}^C) \leq \pi'(v) \leq \pi'(v_{\max}^C)$.

We first assume that $n_L \geq n_R$. Let Y be the set of vertices v such that $v \in V(C')$ for some cluster C' of type τ and $\pi'(v) < \pi'(v_{\min}^C)$. Since π' has Property (Π_1) we know that there is a set of clusters \mathcal{C}_Y such that $Y = \bigcup_{C' \in \mathcal{C}_Y} V(C') \cap B_\ell$. Since π' has Property (Π_2) we observe that $X \cap J_L^{(\ell, \lambda_\ell(\tau))} = J_L^{(\ell, \lambda_\ell(\tau))} \setminus (\bigcup_{C' \in \mathcal{C}_Y} (V(C') \cap B_\ell))$ while $X \cap J_R^{(r, \rho_r(\tau))} = \bigcup_{C' \in \mathcal{C}_Y} (V(C') \cap B_r) \cup (V(C) \cap B_r)$. We further know that $X \setminus (J_L^{(\ell, \lambda_\ell(\tau))} \cup J_R^{(r, \rho_r(\tau))})$ consists of precisely the vertices in $J_L^{(\ell, i)}$ for every $i > \lambda_\ell(\tau)$, the vertices in B_i for $\ell < i < r$, the vertices $\{s \in S' : \ell < \sigma(s) < r\}$, as well as the vertices in $J_R^{(r, i)}$ for every $i < \rho_r(\tau)$. We conclude that

$$X = \left(J_L^{(\ell, \lambda_\ell(\tau))} \setminus \bigcup_{C' \in \mathcal{C}_Y} (V(C') \cap B_\ell) \right) \cup \bigcup_{i \in [\lambda_\ell(\tau) + 1, |T_L^i|]} J_L^{(\ell, i)}$$

$$\cup \bigcup_{\ell < i < r} B_i \cup \{s \in S' : \ell < \sigma(s) < r\}$$

$$\cup \bigcup_{i \in [1, \rho_r(\tau) - 1]} J_R^{(r, i)} \cup \bigcup_{C' \in \mathcal{C}_Y} (V(C') \cap B_r) \cup (V(C) \cap B_r).$$

To determine the size of X note that $J_L^{(\ell, \lambda_\ell(\tau'))} = \#L(\tau') \cdot \alpha(x_{\tau'})$ and $J_R^{(r, \rho_r(\tau'))} = \#R(\tau') \cdot \alpha(x_{\tau'})$. Hence we obtain

$$\begin{aligned} \text{stretch}_{\pi'}(\{v_{\min}^C, v_{\max}^C\}) &= |X| - 1 \\ &= \sum_{\tau' \in \lambda_\ell^{-1}([\lambda_\ell(\tau), |\mathcal{T}_L^i|])} \#L(\tau') \cdot \alpha(x_{\tau'}) - \sum_{C' \in \mathcal{C}_Y} n_L \\ &\quad + \sum_{\ell < i < r} \alpha(z_i) + (r - \ell) \\ &\quad + \sum_{\tau' \in \rho_r^{-1}([1, \rho_r(\tau) - 1])} \#R(\tau') \cdot \alpha(x_{\tau'}) + \sum_{C' \in \mathcal{C}_Y} n_R + n_R - 1 \\ &\leq \sum_{\tau' \in \lambda_\ell^{-1}([\lambda_\ell(\tau), |\mathcal{T}_L^i|])} \#L(\tau') \cdot \alpha(x_{\tau'}) + \sum_{\ell < i < r} \alpha(z_i) + (r - \ell) \\ &\quad + \sum_{\tau' \in \rho_r^{-1}([1, \rho_r(\tau) - 1])} \#R(\tau') \cdot \alpha(x_{\tau'}) + n_R - 1 \\ &\leq b, \end{aligned}$$

where the first inequality follows due to the assumption that $n_L \geq n_R$ (thus $\sum_{C' \in \mathcal{C}_Y} n_R - \sum_{C' \in \mathcal{C}_Y} n_L \leq 0$) and the last inequality follows due to (C2) as $\alpha(y_\tau) = 1$ (as there exists some cluster of distribution-type τ).

In the case that $n_L < n_R$ we can conclude that $\text{stretch}_{\pi'}(\{v_{\min}^C, v_{\max}^C\}) \leq b$ by a symmetric argument. Hence $\text{stretch}(\pi') \leq b$ concluding the proof of the statement. \square

Using Lemma 14 we obtain an FPT-algorithm which computes $S, \#\kappa$ for every cluster-type κ , and arbitrary picks an extended deletion set S' . Then, for every ordering $\sigma : S' \rightarrow [k']$, the algorithm verifies whether the ILP admits a solution in which case the input is a YES-instance of BANDWIDTH.

Proof of Theorem 2 Let (G, b) be an instance of BANDWIDTH, and let $S \subseteq V(G)$ be a cluster deletion set of G of size $|S| = \text{cvd}(G) = k$ (such a set can be computed in time $1.811^k n^{O(1)}$ using the algorithm of [51]). Notice that one can compute $\#\kappa$ for every cluster-type $\kappa \in \mathcal{K}$ in polynomial time, by checking the S -neighborhood of every vertex in every clique of $G - S$. Moreover, the size of every clique of $G - S$ is at most $\omega(G)$. Let $S' = S \cup \bigcup_{C \in \mathcal{C}} C$ be an extended deletion set of G , for some representative clique set \mathcal{C} . Due to Observation 5 and Remark 10, one can compute λ_i and ρ_i for $i \in [0, |S'|]$ in time FPT in $\text{cvd}(G) + \omega(G)$. Due to Lemma 3, there exists an S' -extremal ordering π of $V(G)$ such that $\text{stretch}(\pi) = \text{bw}(G)$. Notice that π is S' -extremal and compatible with σ , for some ordering σ of S' .

Fix an ordering $\sigma : S' \rightarrow [|S'|]$ of S' . Due to Lemma 14, one can verify whether there exists an S' -extremal ordering of $V(G)$ compatible with σ of stretch at most b by solving the ILP. Notice that there are exactly $|S'|!$ different orderings σ , thus one can decide whether $\text{bw}(G) \leq b$ by solving an ILP for each such ordering.

Due to Observation 5 as well as the formulation of the ILP, it follows that both the number of ILPs solved as well as the number of variables of the ILPs are bounded by a function of $\text{cvd}(G) + \omega(G)$. Since ILP is FPT parameterized by the number of variables, it follows that BANDWIDTH is FPT parameterized by $\text{cvd}(G) + \omega(G)$. \square

Remark 15 Using a minimization ILP, we can in fact *construct* an ordering of *minimum* stretch (and not just argue about the existence of an ordering of stretch at most b), since all the exchange arguments of Sect. 3.2 are constructive.

4 W[1]-Hardness Parameterized by Cluster Vertex Deletion Number

In this section we prove that BANDWIDTH is W[1]-hard when parameterized by the cluster vertex deletion number of the input graph. In order to do so, we present a parameterized reduction from UNARY BIN PACKING parameterized by the number of bins, which is well-known to be W[1]-hard [37].

UNARY BIN PACKING

Instance: A multiset $A = \{a_1, \dots, a_n\}$ of integers in unary, as well as $k \in \mathbb{N}$.

Goal: Determine whether there is a partition of $[n]$ into k subsets $\mathcal{I}_1, \dots, \mathcal{I}_k$, such that for all $i \in [k]$, $\sum_{j \in \mathcal{I}_i} a_j = \sum_{j \in [n]} a_j / k$.

Before we present the details of the construction, we first give some high-level intuition. For an instance (A, k) of UNARY BIN PACKING we want to construct an equivalent instance (G, b) of BANDWIDTH, such that $\text{cvd}(G) = f(k)$ for some function f . Roughly, the graph G consists of cliques representing the items of the UNARY BIN PACKING instance and cliques that act as delimiters separating the items contained in some bucket from the items contained in the next bucket. However, in order to guarantee that the entirety of every item clique is placed in between two consecutive delimiter cliques and that the values of the items in between two delimiter cliques add up to B (the capacity of the bins in the UNARY BIN PACKING instance (A, k)), some extra structure is needed. First we introduce two cliques of size $b + 1$ that will be used as boundaries. By making each item clique and each delimiter clique of the graph adjacent to some vertex in both of the boundary cliques, it follows that in any ordering of stretch at most b , all item cliques and all delimiter cliques of the graph will be positioned in between the two boundary cliques.

As the size of the deletion set cannot depend on the number or values of the items, item cliques cannot be incident to individual deletion set vertices. This makes it tricky to enforce that every vertex of an item clique is contained in between the same two delimiter cliques as a majority of the item cliques would not be incident to any edge of maximum stretch and therefore allow them a lot of freedom of movement. In order to

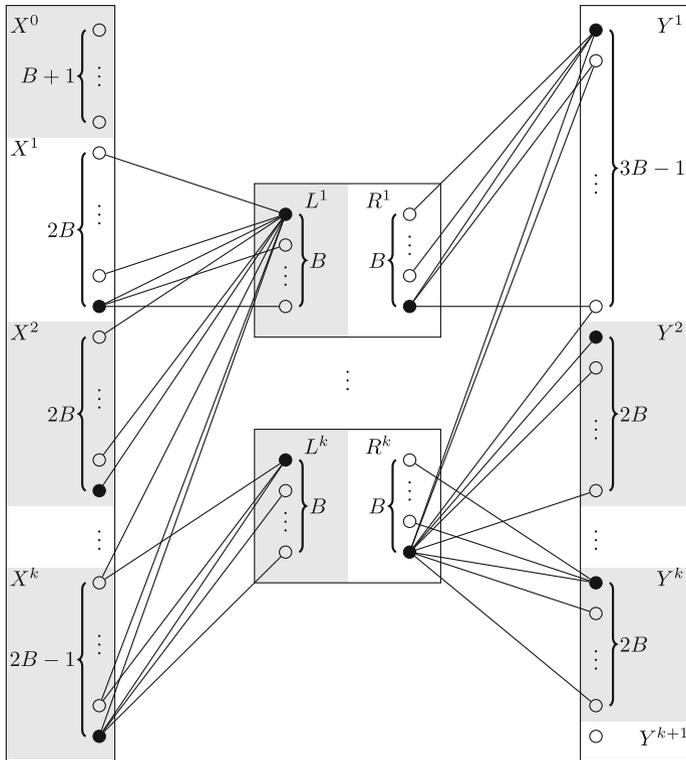


Fig. 3 Part of G , showing only the boundary and the delimiter cliques. Rectangles denote cliques, brackets denote number of vertices, and black vertices compose a cluster deletion set

cope with this issue, we introduce a perfect copy of the delimiter and item cliques, as well as edges between the original cliques and their copies resulting in them becoming twice as big consisting of a left part, the original vertices, and a right part, the copy vertices. The left part of all cliques will be connected to the left boundary clique and will therefore appear to the left of the right parts. The right part will be connected to the right hand boundary cliques. The item cliques will now be kept in place by having maximum stretch between the vertices of the left part and the vertices of the right part.

Theorem 16 BANDWIDTH is $W[1]$ -hard when parameterized by the cluster vertex deletion number of the input graph.

Construction. Let (A, k) be an instance of UNARY BIN PACKING, where $A = \{a_1, \dots, a_n\}$. Moreover, let $B = \sum_{j \in [n]} a_j / k$ be the capacity of every bin, where $B \in \mathbb{N}$, since otherwise this would have been a trivial instance. Set $b = 2kB + B - 1$. We will construct an equivalent instance (G, b) of BANDWIDTH as follows.

Boundary cliques. First, we create two cliques X and Y , referred to as *boundary cliques*, where $V(X) = \{x_1, \dots, x_{2kB+B}\}$ and $V(Y) = \{y_1, \dots, y_{2kB+B}\}$. We consider the following partition of the vertices of X : let $X^0 = \{x_1, \dots, x_{B+1}\}$ and for every $i \in [k - 1]$ we denote the set $\{x_{2iB-B+2}, \dots, x_{2iB+B+1}\}$ by X^i , while

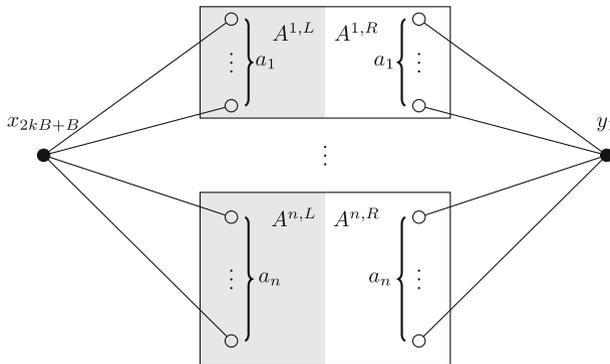


Fig. 4 Rectangles denote cliques. Black vertices compose a cluster deletion set

$X^k = \{x_{2kB-B+2}, \dots, x_{2kB+B}\}$. Note that $|X^0| = B + 1$, $|X^k| = 2B - 1$, and $|X^i| = 2B$, for all $i \in [k - 1]$. Moreover, we partition the vertices of Y in a similar but slightly asymmetric way: let $Y^1 = \{y_1, \dots, y_{3B-1}\}$ and for every $i \in [2, k]$ we denote the set $\{y_{2iB-B}, \dots, y_{2iB+B-1}\}$ by Y^i , while $Y^{k+1} = \{y_{2kB+B}\}$. Note that $|Y^1| = 3B - 1$, $|Y^{k+1}| = 1$, and $|Y^i| = 2B$, for all $i \in [2, k]$.

Delimiter cliques. For every $i \in [k]$ we create a clique on vertex set $\{\ell_1^i, \dots, \ell_B^i, r_1^i, \dots, r_B^i\}$ of size $2B$. We denote the set $\{\ell_1^i, \dots, \ell_B^i\}$ by L^i and the set $\{r_1^i, \dots, r_B^i\}$ by R^i . Moreover, let $L = \bigcup_{i=1}^k L^i$ and $R = \bigcup_{i=1}^k R^i$. We add the following edges:

- For every $i \in [k]$, $x \in \bigcup_{j=i}^k X^j$, we add the edge $\{\ell_1^i, x\}$.
- For every $i \in [k - 1]$, $\ell \in L^i$, we add the edge $\{x_{2iB+B+1}, \ell\}$. Moreover, we add an edge between x_{2kB+B} and every vertex of L^k .
- For every $i \in [k]$, $y \in \bigcup_{j=1}^i Y^j$, we add the edge $\{r_B^i, y\}$.
- For every $i \in [2, k]$, $r \in R^i$, we add the edge $\{y_{2iB-B}, r\}$. Moreover, we add an edge between y_1 and every vertex of R^1 .

For an illustration of the boundary and delimiter cliques, see Fig. 3.

Item cliques. For element $a_i \in A$, we construct a clique A^i on vertex set $\{a_j^{i,L}, a_j^{i,R} : j \in [a_i]\}$ of size $2a_i$. We denote the set of vertices $\{a_j^{i,L} : j \in [a_i]\}$ by $A^{i,L}$ and the set of vertices $\{a_j^{i,R} : j \in [a_i]\}$ by $A^{i,R}$. We add edges $\{x_{2kB+B}, a\}$ for every $a \in \bigcup_{i \in [k]} A^{i,L}$ and edges $\{y_1, a\}$ for every $a \in \bigcup_{i \in [k]} A^{i,R}$. For an illustration, see Fig. 4.

This concludes the construction of G . Figure 5 illustrates an example of an ordering of stretch b obtained by a YES-instance of UNARY BIN PACKING. In the following, we prove the equivalence of (G, b) to the initial instance of UNARY BIN PACKING.

Lemma 17 *If (A, k) is a YES-instance of UNARY BIN PACKING, then (G, b) is a YES-instance of BANDWIDTH.*

Proof Let $(\mathcal{I}_1, \dots, \mathcal{I}_k)$ be a partition of $[n]$, where $\sum_{j \in \mathcal{I}_i} a_j = B$, for all $i \in [k]$. For all $i \in [k]$, set $\mathcal{S}_i = \bigcup_{j \in \mathcal{I}_i} A^{j,L}$ and $\mathcal{S}'_i = \bigcup_{j \in \mathcal{I}_i} A^{j,R}$. Notice that $|\mathcal{S}_i| = |\mathcal{S}'_i| = B$, since $|A^{j,L}| = |A^{j,R}| = a_j$.

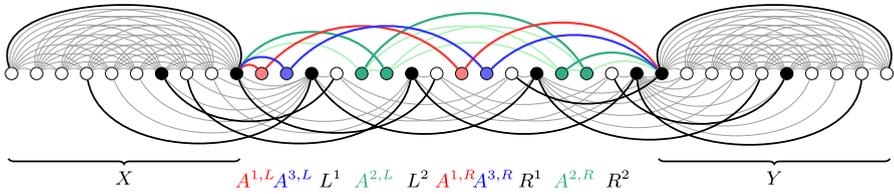


Fig. 5 For the instance $(\{a_1, a_2, a_3\}, 2)$ of UNARY BIN PACKING with $a_1 = 1, a_2 = 2,$ and $a_3 = 1$ the figure shows the graph G from the corresponding instance $(G, 9)$ of BANDWIDTH. Here the ordering of the vertices of G with stretch 9 corresponds to the solution of $(\{a_1, a_2, a_3\}, 2)$ in which a_1, a_3 are placed in the first bin and a_2 in the second

For $i \in [k]$, we define $\pi_{S_i}: S_i \rightarrow [|S_i|]$ to be an arbitrary ordering of S_i and $\pi_{S'_i}: S'_i \rightarrow [|S'_i|]$ to be an arbitrary ordering of S'_i . In order to define the ordering of the vertices of the graph $\pi: V(G) \rightarrow [|V(G)|]$, we will make use of these orderings. In particular, define π as follows:

- First, let $\pi(x_i) = i$ for all $x_i \in V(X)$.
- For every $i \in [k]$, let
 - $\pi(v) = 2kB + B + (i - 1)2B + \pi_{S_i}(v)$ for every $v \in S_i$,
 - $\pi(\ell_j^i) = 2kB + 2B + (i - 1)2B + j = 2kB + 2iB + j$, for every $\ell_j^i \in L^i$,
 - $\pi(v) = 4kB + B + (i - 1)2B + \pi_{S'_i}(v)$ for every $v \in S'_i$,
 - $\pi(r_j^i) = 4kB + 2B + (i - 1)2B + j = 4kB + 2iB + j$, for every $r_j^i \in R^i$.
- Lastly, let $\pi(y_i) = 6kB + B + i$ for all $y_i \in V(Y)$.

On a high level, we place first the vertices of X , then the vertices of S_1 , followed by the vertices of L^1 , then of S_2 , and so on. After the vertices of L^k , we proceed in an analogous way, by placing the vertices of S'_1 , followed by the vertices of R^1 , then of S'_2 , and so on. After the vertices of R^k , we place the vertices of Y . The ordering of the vertices within $X, Y, L^i,$ and R^i is dictated by their index, whereas the ordering of S_i and S'_i is dictated by their corresponding ordering π_{S_i} and $\pi_{S'_i}$. See Fig. 5 for an illustration of the ordering by means of a small example instance of UNARY BIN PACKING.

It remains to show that $\text{stretch}_\pi(e) \leq b = 2kB + B - 1$ for all $e \in E(G)$. In order to do so, we will consider all the possible cases regarding the endpoints of e .

We first consider the edges $e \in E(X) \cup E(Y)$. Notice that the vertices of each of the cliques X and Y have been placed consecutively in π , thus any edge $e \in E(X) \cup E(Y)$ has stretch at most equal to the size of these cliques minus one, therefore $\text{stretch}_\pi(e) \leq 2kB + B - 1 = b$.

Now we consider the edges $e = \{x, \ell\}$, where $x \in V(X)$ and $\ell \in L$. Notice that $\pi(x) < \pi(\ell)$. Suppose that $x \in X^i$ for $i \in [k]$, and set x_{\min}^i and x_{\max}^i such that $\pi(x_{\min}^i) = \min(\pi(X^i))$ and $\pi(x_{\max}^i) = \max(\pi(X^i))$. Notice that it holds $x_{\min}^i = x_{2iB-B+2}$. Moreover $x_{\max}^i = x_{2iB+B+1}$ if $i \neq k$, otherwise $x_{\max}^k = x_{2kB+B}$, and $\pi(x_{\max}^i) \geq 2iB+B$ for all $i \in [k]$ follows. Notice that if $x \neq x_{\max}^i$, then $\max(\pi(N(x) \cap L)) = \pi(\ell_1^i)$, since then $N(x) \cap L = \{\ell_1^j : j \in [i]\}$. In that case, it holds that $\pi(\ell) - \pi(x) \leq \pi(\ell_1^i) - \pi(x_{\min}^i) = 2kB + 2iB + 1 - (2iB - B + 2) = 2kB + B - 1 = b$.

Alternatively, if $x = x_{\max}^i$, then notice that $\max(\pi(N(x) \cap L)) = \pi(\ell_B^i)$, since then $N(x) \cap L = \bigcup_{j=1}^i L^j$. In that case, it holds that $\pi(\ell) - \pi(x) \leq \pi(\ell_B^i) - \pi(x) \leq 2kB + 2iB + B - (2iB + B) = 2kB < b$.

Next we consider the edges $e = \{y, r\}$, where $y \in V(Y)$ and $r \in R$. Notice that $\pi(r) < \pi(y)$. Suppose that $y \in Y^i$ for $i \in [k]$, and set y_{\max}^i and y_{\min}^i such that $\pi(y_{\max}^i) = \max(\pi(Y^i))$ and $\pi(y_{\min}^i) = \min(\pi(Y^i))$. Notice that it holds that $y_{\max}^i = y_{2iB+B-1}$. Moreover, $y_{\min}^i = y_{2iB-B}$ if $i \neq 1$, otherwise $y_{\min}^1 = y_1$, and $\pi(y_{\min}^i) \leq 6kB + B + 2iB - B$ for all $i \in [k]$ follows. Notice that if $y \neq y_{\min}^i$, then $\min(\pi(N(y) \cap R)) = \pi(r_B^i)$, since $N(y) \cap R = \{r_B^j : j \in [i, k]\}$. In that case, it holds that $\pi(y) - \pi(r) \leq \pi(y_{\max}^i) - \pi(r_B^i) = 6kB + B + 2iB + B - 1 - (4kB + 2iB + B) = 2kB + B - 1 = b$. Alternatively, if $y = y_{\min}^i$, then notice that $\min(\pi(N(y) \cap R)) = \pi(r_1^i)$, since then $N(y) \cap R = \bigcup_{j=i}^k R^j$. In that case, it holds that $\pi(y) - \pi(r) \leq \pi(y) - \pi(r_1^i) \leq 6kB + B + 2iB - B - (4kB + 2iB + 1) = 2kB - 1 < b$.

Now we consider the edges $e = \{u, v\}$ where $u, v \in L^i \cup R^i$ for some $i \in [k]$. Notice that the single edge of maximum stretch out of those edges is $\{\ell_1^i, r_B^i\}$, therefore $\text{stretch}_\pi(e) \leq \pi(r_B^i) - \pi(\ell_1^i) = 4kB + 2iB + B - (2kB + 2iB + 1) = 2kB + B - 1 = b$.

Lastly, consider the edges that are incident to the vertices of A^j for some $j \in [n]$. We consider three cases.

First, assume that both $u, v \in V(A^j)$. Then, notice that there exists $i \in [k]$ such that $A^{j,L} \subseteq S_i$ and $A^{j,R} \subseteq S'_j$. In that case, $\text{stretch}_\pi(e) \leq \max(\pi(S'_j)) - \min(\pi(S_i))$. It holds that $\max(\pi(S'_j)) = \pi(r_1^i) - 1 = 4kB + 2iB$, while $\min(\pi(S_i)) = \pi(\ell_B^{i-1}) + 1 = 2kB + 2iB - B + 1$ if $i \neq 1$ and $\min(\pi(S_1)) = 2kB + B + 1$ otherwise, i.e., $\min(\pi(S_i)) = 2kB + 2iB - B + 1$ for all $i \in [k]$. Consequently, it holds that $\text{stretch}_\pi(e) \leq 4kB + 2iB - (2kB + 2iB - B + 1) = 2kB + B - 1 = b$.

For the second case, assume that $e = \{u, x_{2kB+B}\}$, where $u \in A^{j,L}$ for some $j \in [n]$. Notice that then $\pi(x_{2kB+B}) < \pi(u)$, while $\pi(u) \leq \pi(\ell_B^k) - 1 = 4kB$, therefore $\text{stretch}_\pi(e) = \pi(u) - \pi(x_{2kB+B}) \leq 4kB - (2kB + B) = 2kB - B < b$.

Finally, assume that $e = \{u, y_1\}$, where $u \in A^{j,R}$ for some $j \in [n]$. Notice that then $\pi(u) < \pi(y_1)$, while $\pi(u) \geq \pi(\ell_B^k) + 1 = 4kB + B + 1$, therefore $\text{stretch}_\pi(e) = \pi(y_1) - \pi(u) \leq 6kB + B + 1 - (4kB + B + 1) = 2kB < b$.

Since it holds that $\text{stretch}_\pi(e) \leq b$ for all $e \in E(G)$, it follows that (G, b) is a YES-instance of BANDWIDTH. \square

Lemma 18 *If (G, b) is a YES-instance of BANDWIDTH, then (A, k) is a YES-instance of UNARY BIN PACKING.*

Proof Let $\pi : V(G) \rightarrow [|V(G)|]$ be an ordering of the vertices of G such that $\text{stretch}_\pi(e) \leq b$, for all $e \in E(G)$. We first prove that the vertices of X and Y appear at the start and the end of the ordering respectively, or vice versa.

Claim 19 One of the following holds:

- either $\pi(V(X)) = [b + 1]$ and $\pi(V(Y)) = [|V(G)| - b, |V(G)|]$,
- or $\pi(V(Y)) = [b + 1]$ and $\pi(V(X)) = [|V(G)| - b, |V(G)|]$.

Proof First, observe that since X (respectively, Y) is a clique of size $b + 1$, if its vertices were not occupying $b + 1$ consecutive positions in π , there would exist an

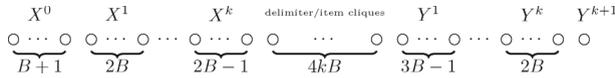


Fig. 6 Layout of $X^0, \dots, X^k, Y^1, \dots, Y^{k+1}$ in π , as derived in Claim 20

edge of stretch more than b , which is a contradiction. Consequently, the vertices of X (respectively, Y) occupy $b + 1$ consecutive positions in π .

Now, assume that the vertices of $Z \in \{X, Y\}$ appear neither at the start nor at the end of the ordering π . Then, there exist vertices placed in the ordering π both before and after the positions occupied by the vertices of Z . Notice that $G' = G - V(Z)$ is connected, and let $v_1, v_2 \in V(G')$ such that $\{v_1, v_2\} \in E(G')$ and $\pi(v_1) < \pi(z) < \pi(v_2)$, for every $z \in V(Z)$. Since $|V(Z)| = b + 1$, it follows that $\pi(v_2) - \pi(v_1) > b$, which is a contradiction. \square

As a consequence of Claim 19, it follows that the remaining $|V(G)| - 2(b + 1) = 2kB + 2 \sum_{i=1}^n a_i = 4kB$ vertices of the delimiter and item cliques are positioned between the vertices of the boundary cliques in π . Without loss of generality, assume that the first case of Claim 19 holds, thus the vertices of X appear at the start and the vertices of Y at the end of ordering π (if that is not the case, all arguments made in the following for π hold for the reverse ordering $\pi'(x) = |V(G)| + 1 - \pi(x)$ and hence the statement follows using a symmetric argument). Next, we will prove that the vertices of $X^0, X^1, X^2, \dots, X^k$ and of Y^1, \dots, Y^k, Y^{k+1} appear sequentially in π , as depicted in Fig. 6. Moreover, we will specify the positions of vertices ℓ_1^i and r_B^i , for $i \in [k]$.

Claim 20 Regarding the vertices of X and Y , the following hold:

- $\pi(X^0) = [B + 1]$,
- for every $i \in [k]$, $\pi(X^i) = [w_i^X, w_i^X + |X^i| - 1]$, where $w_i^X = B + 1 + (i - 1)2B + 1 = 2iB - B + 2$,
- $\pi(Y^1) = [6kB + B + 1, 6kB + 4B - 1]$,
- for every $i \in [2, k + 1]$, $\pi(Y^i) = [w_i^Y, w_i^Y + |Y^i| - 1]$, where $w_i^Y = 6kB + 4B - 1 + (i - 2)2B + 1 = 6kB + 2iB$.

Additionally, for all $i \in [k]$, it holds that

- $\pi(\ell_1^i) = \min(\pi(L^i)) = 2kB + B + B + 2B \cdot (i - 1) + 1 = 2kB + 2iB + 1$ and
- $\pi(r_B^i) = \max(\pi(R^i)) = 2kB + B + 2kB + i \cdot 2B = 4kB + 2iB + B$.

Proof Let for $v_1, v_2 \in V(G)$, $g_\pi(v_1, v_2) = |\pi(v_1) - \pi(v_2)| + 1$ denote the number of vertices positioned between $\pi(v_1)$ and $\pi(v_2)$ including v_1 and v_2 . Notice that if $e = \{v_1, v_2\} \in E(G)$, then $g_\pi(v_1, v_2) \leq b + 1$, since $\text{stretch}_\pi(e) \leq b$.

Fix some $i \in [k]$. Moreover, let x_{\min}^i be the vertex such that $\pi(x_{\min}^i) = \min(\pi(N(\ell_1^i) \cap V(X)))$, i.e., x_{\min}^i denotes the leftmost positioned neighbor of ℓ_1^i . Analogously, let y_{\max}^i be the vertex such that $\pi(y_{\max}^i) = \max(\pi(N(r_B^i) \cap V(Y)))$, i.e., y_{\max}^i denotes the rightmost positioned neighbor of r_B^i .

Notice that for ℓ_1^i it holds that $N(\ell_1^i) \cap V(X) = \bigcup_{j=i}^k X^j$, therefore ℓ_1^i has exactly $(k - i + 1)2B - 1$ neighbors belonging to X . On the other hand, for r_B^i it holds

that $N(r_B^i) \cap V(Y) = \bigcup_{j=1}^i Y^j$, therefore r_B^i has exactly $3B - 1 + (i - 1)2B = 2iB + B - 1$ neighbors belonging to Y . Consequently, it holds that $g_\pi(x_{\min}^i, y_{\max}^i) \geq (k - i + 1)2B - 1 + 2iB + B - 1 + 4kB = 6kB + 3B - 2 = 3(b + 1) - 2$. Furthermore, $g_\pi(x_{\min}^i, y_{\max}^i) \leq g_\pi(x_{\min}^i, \ell_1^i) + g_\pi(\ell_1^i, r_B^i) + g_\pi(r_B^i, y_{\max}^i) - 2$, since at least two vertices, namely ℓ_1^i and r_B^i , are counted more than once. Since the edges $e_1^i = \{x_{\min}^i, \ell_1^i\}$, $e_2^i = \{\ell_1^i, r_B^i\}$ and $e_3^i = \{r_B^i, y_{\max}^i\}$ are present in G , it follows that $g_\pi(x_{\min}^i, \ell_1^i) + g_\pi(\ell_1^i, r_B^i) + g_\pi(r_B^i, y_{\max}^i) - 2 \leq 3(b + 1) - 2$. Hence, it follows that $g_\pi(x_{\min}^i, \ell_1^i) = g_\pi(\ell_1^i, r_B^i) = g_\pi(r_B^i, y_{\max}^i) = b + 1$, while $g_\pi(x_{\min}^i, y_{\max}^i) = 3(b + 1) - 2$, thus only vertices ℓ_1^i and r_B^i are counted twice in the previous sum.

Consequently, it holds that $\pi(\bigcup_{j=i}^k X^j) = [2kB + B - |\bigcup_{j=i}^k X^j| + 1, 2kB + B]$, while $\pi(\bigcup_{j=1}^i Y^j) = [6kB + B + 1, 6kB + B + 1 + |\bigcup_{j=1}^i Y^j| - 1]$, i.e., the vertices of $N(\ell_1^i) \cap V(X)$ appear in the rightmost positions occupied by the vertices of X , while the vertices of $N(r_B^i) \cap V(Y)$ appear in the leftmost positions occupied by the vertices of Y . Having determined $\pi(x_{\min}^i)$ and $\pi(y_{\max}^i)$, as well as that $\pi(\ell_1^i) - \pi(x_{\min}^i) = \pi(y_{\max}^i) - \pi(r_B^i) = b$, the values of $\pi(\ell_1^i)$ and $\pi(r_B^i)$ follow. Since $\pi(r_B^i) - \pi(\ell_1^i) = b$ and the vertices of $L^i \cup R^i$ form a clique, it follows that $\pi(\ell_1^i) = \min(\pi(L^i))$ and $\pi(r_B^i) = \max(\pi(R^i))$; if that were not the case, then there would exist an edge of stretch larger than b .

Lastly, since $N(\ell_1^i) \cap V(X) = (N(\ell_1^{i+1}) \cap V(X)) \cup X^i$, and $N(r_B^{i+1}) \cap V(Y) = (N(r_B^i) \cap V(Y)) \cup Y^{i+1}$, the vertices of X^i and Y^i are positioned as in Fig. 6. \square

Next, we prove that all vertices of L^i (respectively, R^i) appear before the vertices of L^{i+1} (respectively, R^{i+1}).

Claim 21 For $i \in [k]$ and $j \in [B]$, it holds that

- $\pi(\ell_1^i) \leq \pi(\ell_j^i) < \pi(\ell_1^i) + |X^i|$ and
- $\pi(r_B^i) - |Y^i| < \pi(r_j^i) \leq \pi(r_B^i)$.

Proof Fix some $i \in [k]$. Set $x = x_{2iB+B+1}$ if $i \neq k$ and $x = x_{2kB+B}$ otherwise. Additionally, set $y = y_{2iB-B}$ if $i \neq 1$ and $y = y_1$ otherwise. Notice that $x \in X^i$ while $y \in Y^i$. Let x_{\min}^i and y_{\max}^i be vertices such that $\pi(x_{\min}^i) = \min(\pi(N(\ell_1^i)))$ and $\pi(y_{\max}^i) = \max(\pi(N(r_B^i)))$. Due to Claim 20, it holds that $\pi(x_{\min}^i) \leq \pi(x) \leq \pi(x_{\min}^i) + |X^i| - 1$, while $\pi(y_{\max}^i) - |Y^i| + 1 \leq \pi(y) \leq \pi(y_{\max}^i)$.

Let $j \in [B]$. We will prove that $\pi(\ell_j^i) < \pi(\ell_1^i) + |X^i|$ as well as $\pi(r_B^i) - |Y^i| < \pi(r_j^i)$, since the other inequalities follow from Claim 20.

For the first inequality, notice that $\{x, \ell_j^i\} \in E(G)$, therefore it holds that $\pi(\ell_j^i) - \pi(x) \leq b$. Moreover, due to Claim 20 it holds that $\pi(\ell_1^i) - \pi(x_{\min}^i) = b$. In that case, it follows that

$$\begin{aligned} \pi(\ell_j^i) &\leq b + \pi(x) \\ &= \pi(\ell_1^i) - \pi(x_{\min}^i) + \pi(x) \\ &\leq \pi(\ell_1^i) - \pi(x_{\min}^i) + \pi(x_{\min}^i) + |X^i| - 1 \end{aligned}$$

$$= \pi(\ell_1^i) + |X^i| - 1.$$

As for the second inequality, notice that $\{y, r_j^i\} \in E(G)$, therefore it holds that $\pi(y) - \pi(r_j^i) \leq b$. Moreover, due to Claim 20 it holds that $\pi(y_{\max}^i) - \pi(r_B^i) = b$. In that case,

$$\begin{aligned} \pi(r_j^i) &\geq \pi(y) - b \\ &= \pi(y) - \pi(y_{\max}^i) + \pi(r_B^i) \\ &\geq \pi(y_{\max}^i) - |Y^i| + 1 - \pi(y_{\max}^i) + \pi(r_B^i) \\ &= \pi(r_B^i) - |Y^i| + 1, \end{aligned}$$

and the statement follows. □

In the following, we will refer to the vertex placed at position $2kB + B$ as ℓ_1^0 , while we will additionally refer to ℓ_1^k as r_B^0 . Notice that, due to Claims 20 and 21, it follows that the vertices of the item cliques are positioned in the following way:

- B vertices are placed between positions $\pi(\ell_1^0)$ and $\pi(\ell_1^1)$,
- B vertices are placed between positions $\pi(\ell_1^i)$ and $\pi(\ell_1^{i+1})$, for every $i \in [k - 1]$, since $B - 1$ out of the $2B - 1$ free positions are occupied by vertices of L^i ,
- B vertices are placed between positions $\pi(\ell_1^k)$ and $\pi(r_B^1)$, since $2(B - 1)$ out of the $3B - 2$ free positions are occupied by vertices of L^k and R^1 ,
- B vertices are placed between positions $\pi(r_B^i)$ and $\pi(r_B^{i+1})$, for every $i \in [k - 1]$, since $B - 1$ out of the $2B - 1$ free positions are occupied by vertices of R^i .

Set $\mathcal{S}_i = \{a \in \bigcup_{j \in [n]} A^{j,L} : \pi(\ell_1^{i-1}) < \pi(a) < \pi(\ell_1^i)\}$, while $\mathcal{S}'_i = \{a \in \bigcup_{j \in [n]} A^{j,R} : \pi(r_B^{i-1}) < \pi(a) < \pi(r_B^i)\}$, for every $i \in [k]$. Notice that for every $a \in \bigcup_{j \in [n]} A^{j,R}$, $\pi(\ell_1^k) < \pi(a)$; if that were not the case, then $\min(\pi(Y^1)) - \pi(a) > 6kB + B + 1 - (4kB + 1) = 2kB + B > b$, while $\{a, y_1\} \in E(G)$ and $y_1 \in Y^1$, which is a contradiction. Consequently, sets \mathcal{S}_i partition $\bigcup_{j=1}^n A^{j,L}$, while sets \mathcal{S}'_i partition $\bigcup_{j=1}^n A^{j,R}$. Next, define $\mathcal{I}_i = \{j \in [n] : \mathcal{S}_i \cap A^{j,L} \neq \emptyset\}$ to be the set of indices of the items of the UNARY BIN PACKING instance for which vertices of the corresponding item clique are present in \mathcal{S}_i .

Claim 22 For any $i \in [k]$, it holds that $\mathcal{S}_i = \bigcup_{j \in \mathcal{I}_i} A^{j,L}$ as well as $\mathcal{S}'_i = \bigcup_{j \in \mathcal{I}_i} A^{j,R}$.

Proof We will prove the statement by induction on $i \in [k]$. Let $i \in [k]$ and suppose that the statement holds for all $z \in [1, i - 1]$ (notice that this is trivially true when $i = 1$). We will show that the statement also holds for i .

First, it holds that $\mathcal{I}_i \cap \mathcal{I}_z = \emptyset$ for all $z \in [1, i - 1]$, as for any such z , if $j \in \mathcal{I}_z$ then $A^{j,L} \subseteq \mathcal{S}_z$ due to the induction hypothesis, therefore $A^{j,L} \cap \mathcal{S}_i = \emptyset$, which implies that $j \notin \mathcal{I}_i$.

We now argue that $\bigcup_{j \in \mathcal{I}_i} A^{j,R} \subseteq \mathcal{S}'_i$. Fix an arbitrary $j \in \mathcal{I}_i$. In that case, there exists $a \in A^{j,L} \cap \mathcal{S}_i$ with $\pi(a) < \pi(\ell_1^i)$. Additionally, for all $z \in [1, i - 1]$ it holds that $j \notin \mathcal{I}_z$, since $\mathcal{I}_i \cap \mathcal{I}_z = \emptyset$, thus $A^{j,R} \cap \mathcal{S}'_z = \emptyset$ due to the induction

hypothesis. Moreover, recall that for every $a' \in A^{j,R}$ it holds that $\pi(a') > \pi(\ell_1^k)$, thus $\pi(a') > \pi(r_B^{i-1})$ follows. Assume there exists $a' \in A^{j,R}$ such that $\pi(a') > \pi(r_B^i)$. Then, $\pi(a') - \pi(a) > \pi(r_B^i) - \pi(\ell_1^i) = 2kB + B - 1 = b$, which is a contradiction since $e = \{a, a'\} \in E(G)$, thus $\text{stretch}_\pi(e) \leq b$. Consequently, it follows that $A^{j,R} \subseteq \mathcal{S}'_i$ for any $j \in \mathcal{I}_i$, thus $\bigcup_{j \in \mathcal{I}_i} A^{j,R} \subseteq \mathcal{S}'_i$, which implies that $|\bigcup_{j \in \mathcal{I}_i} A^{j,L}| = |\bigcup_{j \in \mathcal{I}_i} A^{j,R}| \leq |\mathcal{S}'_i| = B$.

Notice that, by the definition of \mathcal{I}_i , it holds that $\bigcup_{j \in \mathcal{I}_i} A^{j,L} \supseteq \mathcal{S}_i$, which implies that $|\bigcup_{j \in \mathcal{I}_i} A^{j,R}| = |\bigcup_{j \in \mathcal{I}_i} A^{j,L}| \geq |\mathcal{S}_i| = B$.

Since we have shown both that $\mathcal{S}_i \subseteq \bigcup_{j \in \mathcal{I}_i} A^{j,L}$ while $|\mathcal{S}_i| = |\bigcup_{j \in \mathcal{I}_i} A^{j,L}|$, and that $\mathcal{S}'_i \supseteq \bigcup_{j \in \mathcal{I}_i} A^{j,R}$ while $|\mathcal{S}'_i| = |\bigcup_{j \in \mathcal{I}_i} A^{j,R}|$, the claim follows. \square

We argue that $(\mathcal{I}_1, \dots, \mathcal{I}_k)$ certifies that (A, k) is a YES-instance of UNARY BIN PACKING. Claim 22 implies that $\mathcal{I}_i \cap \mathcal{I}_j = \emptyset$ for all $1 \leq i < j \leq k$. Additionally, for every $j \in [n]$, there exists $i \in [k]$ such that $j \in \mathcal{I}_i$, since, as already argued, sets $\mathcal{S}_1, \dots, \mathcal{S}_k$ partition set $\bigcup_{j=1}^n A^{j,L}$. Consequently, $(\mathcal{I}_1, \dots, \mathcal{I}_k)$ partitions $[n]$. It remains to show that $\sum_{j \in \mathcal{I}_i} a_j = B$ for all $i \in [k]$. In order to do so, notice that $a_j = |A^{j,L}|$, for all $j \in [n]$. Moreover, $|\mathcal{S}_i| = B$, while due to Claim 22 it holds that $\mathcal{S}_i = \bigcup_{j \in \mathcal{I}_i} A^{j,L}$ for all $i \in [k]$. Consequently,

$$\sum_{j \in \mathcal{I}_i} a_j = \sum_{j \in \mathcal{I}_i} |A^{j,L}| = \left| \bigcup_{j \in \mathcal{I}_i} A^{j,L} \right| = |\mathcal{S}_i| = B,$$

and the statement follows. \square

Lemma 23 *It holds that $\text{cvd}(G) = \mathcal{O}(k)$.*

Proof Let $S = \{x_{2kB+B}\} \cup \{x_{2iB+B+1} : i \in [k-1]\} \cup \{y_1\} \cup \{y_{2iB-B} : i \in [2, k]\} \cup \{\ell_1^i, r_B^i : i \in [k]\}$. It holds that $|S| = 4k = \mathcal{O}(k)$, while $G - S$ is a collection of cliques, therefore $\text{cvd}(G) = \mathcal{O}(k)$ follows. \square

5 Conclusion

In the current work we extend our understanding of BANDWIDTH in the setting of parameterized complexity. In particular, we have shown that the problem is FPT when parameterized by the cluster vertex deletion number cvd plus the clique number ω of the input graph, becomes W[1]-hard when parameterized solely by cvd .

The most natural research direction would be to explore the tractability of the problem when parameterized by twin cover, modular-width, or vertex integrity, given the lack of any relevant FPT/XP algorithms or hardness results. As a matter of fact, it is not even known whether the problem is in XP when parameterized by cvd or tree-depth.

Finally, most tractability results for the various structural parameters rely on some ILP formulation. This raises the question of whether any other kind of approach is applicable, as is the case for CUTWIDTH[11].

Acknowledgements We would like to thank Virginia Ardévol Martínez and Yota Otachi for interesting discussions at the preliminary stages of this work.

Funding Our research visit to Nagoya University, Japan was funded by the PRC CNRS JSPS2019- 2020 program, project PARAGA (Parameterized Approximation Graph Algorithms). *Tatsuya Gima*: Partially supported by JSPS KAKENHI Grant Number JP23KJ1066. Eun Jung Kim: Supported by ANR project ANR-18-CE40-0025-01 (ASSK). Noleen Köhler: Supported by ANR project ANR-18-CE40-0025-01 (ASSK). *Nikolaos Melissinos*: Supported by the CTU Global postdoc fellowship program. Manolis Vasilakis: Partially supported by ANR project ANR-21-CE48-0022 (S-EX-AP-PE-AL).

Data Availability No datasets were generated or analysed during the current study.

Declarations

Conflict of interest The authors declare no competing interests.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

1. Bakken, O.R.: Arrangement Problems Parameterized by Neighbourhood Diversity. Master's thesis, University of Bergen (2018)
2. Banik, A., Kasthurirangan, P.N., Raman, V.: Dominator coloring and CD coloring in almost cluster graphs. In: Algorithms and Data Structures: 18th International Symposium, WADS 2023, volume 14079 of Lecture Notes in Computer Science, pp. 106–119. Springer (2023). https://doi.org/10.1007/978-3-031-38906-1_8
3. Bodlaender, H.L.: Parameterized complexity of bandwidth of caterpillars and weighted path emulation. In: Graph-Theoretic Concepts in Computer Science: 47th International Workshop, WG 2021, volume 12911 of Lecture Notes in Computer Science, pp. 15–27. Springer (2021). https://doi.org/10.1007/978-3-030-86838-3_2
4. Bodlaender, H.L., Groenland, C., Nederlof, J., Swennenhuis, C.: Parameterized problems complete for nondeterministic FPT time and logarithmic space. *Inf. Comput.* **300**, 105195 (2024). <https://doi.org/10.1016/J.IC.2024.105195>
5. Boral, A., Cygan, M., Kociumaka, T., Pilipczuk, M.: A fast branching algorithm for cluster vertex deletion. *Theory Comput. Syst.* **58**(2), 357–376 (2016). <https://doi.org/10.1007/s00224-015-9631-7>
6. Bruhn, H., Chopin, M., Joos, F., Schaudt, O.: Structural parameterizations for boxicity. *Algorithmica* **74**(4), 1453–1472 (2016). <https://doi.org/10.1007/s00453-015-0011-0>
7. Chinn, P.Z., Chvatalova, J., Dewdney, A.K., Gibbs, N.E.: The bandwidth problem for graphs and matrices: a survey. *J. Graph Theory* **6**(3), 223–254 (1982). <https://doi.org/10.1002/jgt.3190060302>
8. Chlebíková, J., Chopin, M.: The firefighter problem: further steps in understanding its complexity. *Theor. Comput. Sci.* **676**, 42–51 (2017). <https://doi.org/10.1016/J.TCS.2017.03.004>
9. Chopin, M., Nichterlein, A., Niedermeier, R., Weller, M.: Constant thresholds can make target set selection tractable. *Theory Comput. Syst.* **55**(1), 61–83 (2014). <https://doi.org/10.1007/s00224-013-9499-3>
10. Cygan, M., Fomin, F.V., Kowalik, L., Lokshtanov, D., Marx, D., Pilipczuk, M., Pilipczuk, M., Saurabh, S.: Parameterized Algorithms. Springer (2015). <https://doi.org/10.1007/978-3-319-21275-3>

11. Cygan, M., Lokshtanov, D., Pilipczuk, M., Pilipczuk, M., Saurabh, S.: On cutwidth parameterized by vertex cover. *Algorithmica* **68**(4), 940–953 (2014). <https://doi.org/10.1007/s00453-012-9707-6>
12. Cygan, M., Pilipczuk, M.: Exact and approximate bandwidth. *Theor. Comput. Sci.* **411**(40–42), 3701–3713 (2010). <https://doi.org/10.1016/j.tcs.2010.06.018>
13. Cygan, M., Pilipczuk, M.: Bandwidth and distortion revisited. *Discret. Appl. Math.* **160**(4–5), 494–504 (2012). <https://doi.org/10.1016/j.dam.2011.10.032>
14. Cygan, M., Pilipczuk, M.: Even faster exact bandwidth. *ACM Trans. Algorithms* **8**(1), 8:1–8:14 (2012). <https://doi.org/10.1145/2071379.2071387>
15. Díaz, J., Petit, J., Serna, M.J.: A survey of graph layout problems. *ACM Comput. Surv.* **34**(3), 313–356 (2002). <https://doi.org/10.1145/568522.568523>
16. Diestel, R.: *Graph Theory, Volume 173 of Graduate Texts in Mathematics*. Springer (2017). <https://doi.org/10.1007/978-3-662-53622-3>
17. Doucha, M., Kratochvíl, J.: Cluster vertex deletion: a parameterization between vertex cover and clique-width. In: *Mathematical Foundations of Computer Science 2012: 37th International Symposium, MFCS 2012, volume 7464 of Lecture Notes in Computer Science*, pp. 348–359. Springer (2012). https://doi.org/10.1007/978-3-642-32589-2_32
18. Dregi, M.S., Lokshtanov, D.: Parameterized complexity of bandwidth on trees. In: *Automata, Languages, and Programming: 41st International Colloquium, ICALP 2014, volume 8572 of Lecture Notes in Computer Science*, pp. 405–416. Springer (2014). https://doi.org/10.1007/978-3-662-43948-7_34
19. Dubey, C.K., Feige, U., Unger, W.: Hardness results for approximating the bandwidth. *J. Comput. Syst. Sci.* **77**(1), 62–90 (2011). <https://doi.org/10.1016/J.JCSS.2010.06.006>
20. Feige, U., Talwar, K.: Approximating the bandwidth of caterpillars. *Algorithmica* **55**(1), 190–204 (2009). <https://doi.org/10.1007/S00453-007-9002-0>
21. Fellows, M.R., Hermelin, D., Rosamond, F.A., Shachnai, H.: Tractable parameterizations for the minimum linear arrangement problem. *ACM Trans. Comput. Theory* **8**(2), 6:1–6:12 (2016). <https://doi.org/10.1145/2898352>
22. Fellows, M.R., Lokshtanov, D., Misra, N., Mnich, M., Rosamond, F.A., Saurabh, S.: The complexity ecology of parameters: an illustration using bounded max leaf number. *Theory Comput. Syst.* **45**(4), 822–848 (2009). <https://doi.org/10.1007/s00224-009-9167-9>
23. Fellows, M.R., Lokshtanov, D., Misra, N., Rosamond, F.A., Saurabh, S.: Graph layout problems parameterized by vertex cover. In: *Algorithms and Computation, 19th International Symposium, ISAAC 2008, volume 5369 of Lecture Notes in Computer Science*, pp. 294–305. Springer (2008). https://doi.org/10.1007/978-3-540-92182-0_28
24. Frank, A., Tardos, É.: An application of simultaneous diophantine approximation in combinatorial optimization. *Combinatorica* **7**, 49–65 (1987). <https://doi.org/10.1007/BF02579200>
25. Fürer, M., Gaspers, S., Kasiviswanathan, S.P.: An exponential time 2-approximation algorithm for bandwidth. *Theor. Comput. Sci.* **511**, 23–31 (2013). <https://doi.org/10.1016/j.tcs.2013.03.024>
26. Gajarský, J., Lampis, M., Ordyniak, S.: Parameterized algorithms for modular-width. In: *Parameterized and Exact Computation: 8th International Symposium, IPEC 2013, volume 8246 of Lecture Notes in Computer Science*, pp. 163–176. Springer (2013). https://doi.org/10.1007/978-3-319-03898-8_15
27. Ganian, R.: Improving vertex cover as a graph parameter. *Discret. Math. Theor. Comput. Sci.* **17**(2), 77–100 (2015). <https://doi.org/10.46298/DMTCS.2136> <https://doi.org/10.46298/DMTCS.2136>
28. Ganian, R., Hliněný, P., Nešetřil, J., Obdržálek, J., de Mendez, P.O.: Shrub-depth: capturing height of dense graphs. *Log. Methods Comput. Sci.* **1**, 10 (2019). [https://doi.org/10.23638/LMCS-15\(1:7\)2019](https://doi.org/10.23638/LMCS-15(1:7)2019) [https://doi.org/10.23638/LMCS-15\(1:7\)2019](https://doi.org/10.23638/LMCS-15(1:7)2019)
29. Ganian, R., Kim, E.J., Szeider, S.: Algorithmic applications of tree-cut width. *SIAM J. Discret. Math.* **36**(4), 2635–2666 (2022). <https://doi.org/10.1137/20m137478x>
30. Garey, M.R., Graham, R.L., Johnson, D.S., Knuth, D.E.: Complexity results for bandwidth minimization. *SIAM J. Appl. Math.* **34**(3), 477–495 (1978). <https://doi.org/10.1137/0134037>
31. Gima, T., Hanaka, T., Kiyomi, M., Kobayashi, Y., Otachi, Y.: Exploring the gap between treedepth and vertex cover through vertex integrity. *Theor. Comput. Sci.* **918**, 60–76 (2022). <https://doi.org/10.1016/j.tcs.2022.03.021>
32. Gima, T., Kim, E.J., Köhler, N., Melissinos, N., Vasilakis, M.: Bandwidth parameterized by cluster vertex deletion number. In: *18th International Symposium on Parameterized and Exact Computation, IPEC: volume 285 of LIPIcs*, pp. 21:1–21:15. Schloss Dagstuhl - Leibniz-Zentrum für Informatik 2023, (2023). <https://doi.org/10.4230/LIPICS.IPEC.2023.21>

33. Gupta, A.: Improved bandwidth approximation for trees and chordal graphs. *J. Algorithms* **40**(1), 24–36 (2001). <https://doi.org/10.1006/JAGM.2000.1118>
34. Gurari, E.M., Sudborough, I.H.: Improved dynamic programming algorithms for bandwidth minimization and the mincut linear arrangement problem. *J. Algorithms* **5**(4), 531–546 (1984). [https://doi.org/10.1016/0196-6774\(84\)90006-3](https://doi.org/10.1016/0196-6774(84)90006-3)
35. Harper, L.H.: Optimal assignments of numbers to vertices. *J. Soc. Ind. Appl. Math.* **12**(1), 131–135 (1964). <https://doi.org/10.1137/0112012>
36. Hüffner, F., Komusiewicz, C., Moser, H., Niedermeier, R.: Fixed-parameter algorithms for cluster vertex deletion. *Theory Comput. Syst.* **47**(1), 196–217 (2010). <https://doi.org/10.1007/s00224-008-9150-x>
37. Jansen, K., Kratsch, S., Marx, D., Schlotter, I.: Bin packing with fixed number of bins revisited. *J. Comput. Syst. Sci.* **79**(1), 39–49 (2013). <https://doi.org/10.1016/j.jcss.2012.04.004>
38. Kannan, R.: Minkowski’s convex body theorem and integer programming. *Math. Oper. Res.* **12**, 415–440 (1987). <https://doi.org/10.1287/moor.12.3.415>
39. Kare, A.S., Reddy, I.V.: Parameterized algorithms for graph burning problem. In: *Combinatorial Algorithms: 30th International Workshop, IWOCA 2019*, volume 11638 of *Lecture Notes in Computer Science*, pp. 304–314. Springer (2019). https://doi.org/10.1007/978-3-030-25005-8_25
40. Kucera, M., Suchý, O.: Minimum eccentricity shortest path problem with respect to structural parameters. *Algorithmica* **85**(3), 762–782 (2023). <https://doi.org/10.1007/s00453-022-01006-x>
41. Lampis, M.: Algorithmic meta-theorems for restrictions of treewidth. *Algorithmica* **64**(1), 19–37 (2012). <https://doi.org/10.1007/S00453-011-9554-X>
42. Lenstra, H.W., Jr.: Integer programming with a fixed number of variables. *Math. Oper. Res.* **8**(4), 538–548 (1983). <https://doi.org/10.1287/moor.8.4.538>
43. Lokshtanov, D.: *Parameterized Integer Quadratic Programming: Variables and Coefficients* (2017). [arXiv:1511.00310](https://arxiv.org/abs/1511.00310)
44. Majumdar, D., Raman, V.: Structural parameterizations of undirected feedback vertex set: FPT algorithms and kernelization. *Algorithmica* **80**(9), 2683–2724 (2018). <https://doi.org/10.1007/S00453-018-0419-4>
45. Misra, N., Mittal, H.: Imbalance parameterized by twin cover revisited. *Theor. Comput. Sci.* **895**, 1–15 (2021). <https://doi.org/10.1016/j.tcs.2021.09.017>
46. Monien, B.: The bandwidth minimization problem for caterpillars with hair length 3 is np-complete. *SIAM J. Algebraic Discrete Methods* **7**(4), 505–512 (1986). <https://doi.org/10.1137/0607057>
47. Muradian, D.: The bandwidth minimization problem for cyclic caterpillars with hair length 1 is np-complete. *Theor. Comput. Sci.* **307**(3), 567–572 (2003). [https://doi.org/10.1016/S0304-3975\(03\)00238-X](https://doi.org/10.1016/S0304-3975(03)00238-X)
48. Jaroslav Nesetril and Patrice Ossona de Mendez: Tree-depth, subgraph coloring and homomorphism bounds. *Eur. J. Comb.* **27**(6), 1022–1041 (2006). <https://doi.org/10.1016/j.ejc.2005.01.010>
49. Papadimitriou, C.H.: The np-completeness of the bandwidth minimization problem. *Computing* **16**(3), 263–270 (1976). <https://doi.org/10.1007/BF02280884>
50. Saxe, J.B.: Dynamic-programming algorithms for recognizing small-bandwidth graphs in polynomial time. *SIAM J. Algebraic Discret. Methods* **1**(4), 363–369 (1980). <https://doi.org/10.1137/0601042>
51. Tsur, D.: Faster parameterized algorithm for cluster vertex deletion. *Theory Comput. Syst.* **65**(2), 323–343 (2021). <https://doi.org/10.1007/s00224-020-10005-w>

Publisher’s Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Authors and Affiliations

Tatsuya Gima¹  · Eun Jung Kim^{2,3,4}  · Noleen Köhler⁵  · Nikolaos Melissinos⁶  · Manolis Vasilakis⁷ 

✉ Noleen Köhler
n.koehler@leeds.ac.uk

Tatsuya Gima
gima@ist.hokudai.ac.jp

Eun Jung Kim
eunjung.kim@kaist.ac.kr

Nikolaos Melissinos
nikolaos.melissinos@fit.cvut.cz

Manolis Vasilakis
emmanouil.vasilakis@dauphine.eu

- 1 Hokkaido University, Sapporo, Japan
- 2 KAIST, Daejeon, South Korea
- 3 CNRS, Paris, France
- 4 Discrete Mathematics Group, Institute for Basic Science (IBS), Daejeon, South Korea
- 5 University of Leeds, Leeds, UK
- 6 Department of Theoretical Computer Science, Faculty of Information Technology, Czech Technical University, Prague, Czech Republic
- 7 CNRS UMR7243, LAMSADE, Université Paris-Dauphine, PSL University, Paris, France