# Constrained Bipartite Vertex Cover: The Easy Kernel is Essentially Tight*

**Bart M. P. Jansen**

**Eindhoven University of Technology, P. O. Box 513, Eindhoven, The Netherlands**
b.m.p.jansen@tue.nl

**Abstract**

The Constrained Bipartite Vertex Cover problem asks, for a bipartite graph $G$ with partite sets $A$ and $B$, and integers $k_A$ and $k_B$, whether there is a vertex cover for $G$ containing at most $k_A$ vertices from $A$ and $k_B$ vertices from $B$. The problem has an easy kernel with $2k_A \cdot k_B$ edges and $4k_A \cdot k_B$ vertices, based on the fact that every vertex in $A$ of degree more than $k_B$ has to be included in the solution, together with every vertex in $B$ of degree more than $k_A$. We show that the number of vertices and edges in this kernel are asymptotically essentially optimal in terms of the product $k_A \cdot k_B$. We prove that if there is a polynomial-time algorithm that reduces any instance $(G, A, B, k_A, k_B)$ of Constrained Bipartite Vertex Cover to an equivalent instance $(G', A', B', k'_A, k'_B)$ such that $k'_A \in (k_A)^{\mathcal{O}(1)}$, $k'_B \in (k_B)^{\mathcal{O}(1)}$, and $|V(G')| \in \mathcal{O}((k_A \cdot k_B)^{1-\varepsilon})$, for some $\varepsilon > 0$, then $\mathsf{NP} \subseteq \mathsf{coNP/poly}$ and the polynomial-time hierarchy collapses. Using a different construction, we prove that if there is a polynomial-time algorithm that reduces any $n$-vertex instance into an equivalent instance (of a possibly different problem) that can be encoded in $\mathcal{O}(n^{2-\varepsilon})$ bits, then $\mathsf{NP} \subseteq \mathsf{coNP/poly}$.

**1998 ACM Subject Classification** F.2.2 Nonnumerical Algorithms and Problems, G.2.2 Graph Theory

**Keywords and phrases** kernel lower bounds, constrained bipartite vertex cover

**Digital Object Identifier** 10.4230/LIPIcs.STACS.2016.45

## 1 Introduction

**Motivation.** Vertex Cover is a classic problem in combinatorial optimization. It has served as a testbed for a myriad of different techniques in the field of parameterized algorithmics. In this paper we study a variant of this problem on bipartite graphs:

Constrained Bipartite Vertex Cover
**Input:** A bipartite graph $G$ with partite sets $A$ and $B$, and integers $k_A$ and $k_B$.
**Question:** Is there a vertex cover $S$ for $G$ such that $|S \cap A| \leq k_A$ and $|S \cap B| \leq k_B$?

While Vertex Cover is in P for bipartite graphs, this constrained variant is NP-complete [17]. It is motivated by work in reconfigurable VLSI, since it can be used to model the Sparse Allocation Problem. We refer to the recent paper by Bai and Fernau [1] for a detailed overview of the history of the problem and its applications. This paper deals with the limits of efficient preprocessing procedures for Constrained Bipartite Vertex Cover.

Let us call a vertex cover $S$ of a bipartite graph $(k_A, k_B)$-*constrained* if it satisfies $|S \cap A| \leq k_A$ and $|S \cap B| \leq k_B$. Observe that if an instance contains a vertex $v \in A$ of degree more than $k_B$, then any $(k_A, k_B)$-constrained vertex cover includes $a$. If $a$ is not used, then all

* This work was supported by NWO Veni grant “Frontiers in Parameterized Preprocessing” and NWO Gravitation grant “Networks”.


33rd Symposium on Theoretical Aspects of Computer Science (STACS 2016).
Editors: Nicolas Ollinger and Heribert Vollmer; Article No. 45; pp. 45:1–45:13
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany
SYMPOSIUM ON THEORETICAL ASPECTS OF COMPUTER SCIENCE

neighbors of $a$ are in the cover. However, all such neighbors belong to $B$ and there are more than $k_B$ of them. This suggests a simple reduction rule for the problem: if there is a vertex $a \in A$ of degree more than $k_B$, then remove vertex $a$ and decrease $k_A$ by one. Symmetrically, if there is a vertex $b \in B$ of degree more than $k_A$, remove it and decrease $k_B$. If $S$ is a $(k_A, k_B)$-constrained vertex cover of an exhaustively reduced graph, the $k_A$ vertices in $A$ cover at most $\max_{a \in A} \deg(a) \leq k_B$ edges each, and the $k_B$ vertices in $B$ each cover at most $k_A$ edges each. To be able to cover all the edges, the number of edges must therefore be bounded by $2(k_A \cdot k_B)$ and the instance can be rejected if this is not the case. After removing isolated vertices (which do not affect the answer) from the graph, the number of vertices can be bounded by $4(k_A \cdot k_B)$, since each edge contributes at most two vertices.

This simple kernelization strategy for Constrained Bipartite Vertex Cover has been known for over thirty years [10]. It is notably less effective than the well-known kernelization schemes for the classic Vertex Cover problem, which can efficiently reduce any instance $(G, k)$ to an equivalent one with at most $2k$ vertices [5] (cf. [11, Section 4]). It is therefore natural to ask whether a similar size bound can be attained for Constrained Bipartite Vertex Cover. This was posed as an open problem by Marcin Pilipczuk [18].

**Our Results.** We show that, under the assumption that $\mathsf{NP} \not\subseteq \mathsf{coNP/poly}$, neither the number of vertices nor the number of edges in the simple kernel for Constrained Bipartite Vertex Cover can be significantly improved below $\Theta(k_A \cdot k_B)$. The simple preprocessing procedure outlined above is therefore close to optimal in terms of the product $k_A \cdot k_B$.

Concretely, our first result shows that if there is an $\varepsilon > 0$ and a polynomial-time algorithm that reduces any instance $(G, A, B, k_A, k_B)$ of Constrained Bipartite Vertex Cover to an equivalent instance $(G', A', B', k'_A, k'_B)$ such that $k'_A \in (k_A)^{\mathcal{O}(1)}$, $k'_B \in (k_B)^{\mathcal{O}(1)}$, and $|V(G')| \in \mathcal{O}((k_A \cdot k_B)^{1-\varepsilon})$, then $\mathsf{NP} \subseteq \mathsf{coNP/poly}$ and the polynomial-time hierarchy collapses to its third level [19]. This result is obtained using the complementary witness lemma of Dell and van Melkebeek [8]. The lemma shows that $\mathsf{NP} \subseteq \mathsf{coNP/poly}$ follows if the following type of polynomial-time compression algorithm exists for some $c \in \mathbb{N}$: The input is a sequence $x_1, \ldots, x_{n^c}$ of size-$n$ inputs to an $\mathsf{NP}$-hard problem. The output is a single instance $x^*$ of Constrained Bipartite Vertex Cover whose truth status is the logical or of the answers to the inputs, with $|x^*| \in \mathcal{O}(n^c \log n^c)$. We present a construction (Lemma 2) that allows us to obtain precisely such a compression algorithm from a kernelization procedure for Constrained Bipartite Vertex Cover that satisfies the constraints set out above. The key in this construction is to embed the $n^c$ inputs $x_i$ into a Constrained Bipartite Vertex Cover instance on a graph that is *lopsided*: one partite set has size $\mathcal{O}(n^2 c \log n)$, while the other set has size $\mathcal{O}(n^{c+1})$. The produced graph is fairly sparse as the number of edges is at most the number of vertices times the size of the smaller partite set, $\mathcal{O}(n^2 c \log n)$, whose dependence on the *number* $n^c$ of embedded instances is minimal. For sufficiently large $c$, the number of edges in the graph is therefore roughly equal to the number of vertices. If a hypothetical kernel can reduce the order of the composed instance of Constrained Bipartite Vertex Cover substantially, then the relation between the vertex and edge count yields the following: after an application of the simple kernelization, the number of edges reduces substantially as well (since the parameter values $k_A$ and $k_B$ do not increase too much). A kernel with $\mathcal{O}((k_A \cdot k_B)^{1-\varepsilon})$ vertices would therefore give a compression algorithm satisfying the requirements of the complementary witness lemma and imply $\mathsf{NP} \subseteq \mathsf{coNP/poly}$.

Our second result deals with sparsification, i.e., reducing the number of edges in the graph to make it less dense. We prove that unless $\mathsf{NP} \subseteq \mathsf{coNP/poly}$, there is no polynomial-time algorithm that reduces any $n$-vertex instance of Constrained Bipartite Vertex Cover to an equivalent instance, of a possibly different problem, that can be encoded in $\mathcal{O}(n^{2-\varepsilon})$ bits for $\varepsilon > 0$. For this result we can use the framework of cross-composition to avoid invoking the complementary witness lemma directly. The key is to find a construction that embeds a series of $t$ inputs of an NP-hard problem, each of size at most $n$, into an instance of Constrained Bipartite Vertex Cover where both partite sets have roughly $n \cdot \sqrt{t}$ vertices. In sharp contrast to Lemma 2, this second construction produces a very *dense* graph whose partite sets are balanced in size.

After discarding isolated vertices (which play no role in this problem), the number of vertices in an instance is at most twice the number of edges. The lower bound on the number of vertices in the kernel given by the first result therefore implies a similar lower bound on the number of edges. This lower bound holds against kernelizations that incur a bounded increase in the budget values $k_A$ and $k_B$. The sparsification lower bound yields a more general edge lower bound (Corollary 14): even the existence of a kernelization for Constrained Bipartite Vertex Cover with $\mathcal{O}((k_A \cdot k_B)^{1-\varepsilon})$ edges that increases the budget values $k_A$ and $k_B$ arbitrarily, implies $\mathsf{NP} \subseteq \mathsf{coNP/poly}$.

**Related Work.** There is another problem in the literature, called Constrained Minimum Bipartite Vertex Cover, which is similar to ours but behaves differently. Its inputs also consist of a graph $G$ with partite sets $A$ and $B$, along with integers $k_A$ and $k_B$. However, the question is now whether there is a $(k_A, k_B)$-constrained vertex cover that is also a *minimum* vertex cover in $G$, i.e., for which there is no (unconstrained) vertex cover of $G$ that is strictly smaller. This minimality requirement can make it possible to infer that a vertex must belong to any valid solution, while this conclusion is not valid if a vertex cover is allowed whose size is not globally minimum. Chen and Kanj [4, Section 2] showed that the Dulmage-Mendelsohn decomposition of bipartite graphs can be exploited to reduce an instance of Constrained Minimum Bipartite Vertex Cover to an equivalent instance with at most $2(k_A + k_B)$ vertices. Their result does not carry over to the more general problem considered here.

Let us return to Constrained Bipartite Vertex Cover. It has received considerable attention and was studied using a number of different algorithmic paradigms. Fernau and Niedermeier [12] first used the framework of parameterized complexity to attack the Constrained Bipartite Vertex Cover problem. They developed a moderately exponential FPT branching algorithm, aided by the simple problem kernel. Bai and Fernau [1] simplified their algorithm a decade later and report on experimental results.

There are a handful of results concerning tight lower bounds for kernelizations. There is work by Dell and van Melkebeek [8] on Vertex Cover, by Dell and Marx [7] and Hermelin and Wu [14] on packing problems, by Kratsch et al. [16] on Point-Line Cover, and by Jansen [15] on Treewidth.

**Organization.** Section 2 contains preliminaries on parameterized complexity. In Section 3 we develop the lower bound on the number of vertices in kernels for Constrained Bipartite Vertex Cover. Section 4 uses a different construction to give a lower bound on the number of edges. We conclude in Section 5.

## 2 Preliminaries

A parameterized problem $\mathcal{Q}$ is a subset of $\Sigma^* \times \mathbb{N}$, where $\Sigma$ is a finite alphabet. The second component of a tuple $(x, k) \in \Sigma^* \times \mathbb{N}$ is called the *parameter* [6, 9]. A parameterized problem $\mathcal{Q}$ is (strongly uniformly) *fixed-parameter tractable* if there is an algorithm that decides whether $(x,k) \in \mathcal{Q}$ that runs in time $f(k)|x|^{\mathcal{O}(1)}$ for some computable function $f$. The set $\{1,2,\ldots,n\}$ is abbreviated as $[n]$. All logarithms are base 2. For a set $S$ and integer $k$ we denote by $\binom{S}{k}$ the collection of all size-$k$ subsets of $S$.

We say that a set $S$ *avoids* a set $T$ if $S \cap T = \emptyset$. Two vertices $x, y$ in a graph are *false twins* if they are not adjacent to each other, but have exactly the same (open) neighborhood. Two disjoint vertex sets $A, B$ in a graph are *adjacent* if there is an edge of the form $\{a,b\}$ with $a \in A$ and $b \in B$. The sets are *fully adjacent* if all members of $A$ are adjacent to all members of $B$.

▶ **Definition 1** (Generalized kernelization). Let $\mathcal{Q}, \mathcal{Q}' \subseteq \Sigma^* \times \mathbb{N}$ be parameterized problems and let $h\colon \mathbb{N} \to \mathbb{N}$ be a computable function. A *generalized kernelization for $\mathcal{Q}$ into $\mathcal{Q}'$ of size $h(k)$* is an algorithm that, on input $(x,k) \in \Sigma^* \times \mathbb{N}$, takes time polynomial in $|x|+k$ and outputs an instance $(x',k')$ such that:

- $|x'|$ and $k'$ are bounded by $h(k)$.
- $(x',k') \in \mathcal{Q}'$ if and only if $(x,k) \in \mathcal{Q}$.

The algorithm is a *kernelization*, or in short a *kernel*, for $\mathcal{Q}$ if $\mathcal{Q}' = \mathcal{Q}$. It is a *polynomial (generalized) kernelization* if $h(k)$ is a polynomial.

The notion of generalized kernelization, a term first used by Bodlaender et al. [2], is closely related to the notion of *compression* (cf. [3]). In both cases an instance is reduced to a small but equivalent instance of a different problem. In generalized kernelization the output is a parameterized instance, whereas in compression the output is a classical instance.

## 3 Vertex Lower Bound

The goal of this section is to prove a lower bound on the number of vertices in kernels for Constrained Bipartite Vertex Cover. Lemma 2 is the key ingredient.

▶ **Lemma 2.** *Let $n \in \mathbb{N}$ be even and $t \in \mathbb{N}$ be a power of two. There is an algorithm that, given $t$ graphs $G_1,\ldots,G_t$ with exactly $n$ vertices each, outputs a bipartite graph $G'$ with partite sets $A'$ and $B'$, along with integers $k'_A$ and $k'_B$, such that:*

1. *There is an index $i \in [t]$ such that $G_i$ contains a clique of size $n/2$ if and only if $G'$ has a $(k'_A, k'_B)$-constrained vertex cover.*
2. *$k'_A \in \mathcal{O}(n^2 \log t)$ and $k'_B \in \mathcal{O}(n \cdot t)$.*

*The running time is polynomial in $t$ and $n$.*

**Proof.** We describe the construction carried out by the algorithm. It will be easy to see that it can be done in polynomial time. Let the input consist of graphs $G_1,\ldots,G_t$ where $t$ is a power of two, and let each graph have $n$ vertices. For each graph $G_i$, we identify its vertex set with the integers in the range $[n]$. Construct the bipartite graph $G'$, whose partite sets we will denote by $A'$ and $B'$, as follows.

1. Add a *canonical* set $B'_c$ consisting of $n$ vertices $b_1,\ldots,b_n$ to $B'$. Add a *canonical* set $A'_c$ consisting of $\binom{n}{2}$ vertices $a_{i,j}$ for $1 \le i < j \le n$ to $A'$. Add an edge between $b_\ell$ and $a_{i,j}$ if $i = \ell$ or $j = \ell$. This ensures that the graph $G'[A'_c \cup B'_c]$ is the vertex-edge incidence graph of an $n$-vertex clique, which contains the vertex-edge incidence graphs of all $n$-vertex graphs as induced subgraphs. This is why the vertex sets are called canonical.
2. For $i \in [t]$ add a vertex set $B'_i$ to $B'$ consisting of $n$ vertices. These vertices will be false twins in the graph, with identical open neighborhoods. The adjacency between $B'_i$ and the canonical set $A'_c$ encodes the structure of the input graph $G_i$. For each pair $1 \le i < j \le n$ such that $\{i,j\}$ is *not* an edge in $G_i$, make all vertices of $B'_i$ adjacent to $a_{i,j}$.

3. For $i \in [\log t]$, add two vertex sets $A'_{0,i}$ and $A'_{1,i}$ to $A'$, of $\binom{n}{2}$ vertices each. The vertices in $A'_{0,i}$ will be false twins, as will the vertices in $A'_{1,i}$. The adjacencies between the sets $A'_{0/1,j}$ and the sets $B'_i$ are based on the binary encoding of the number $i$. As $t$ is a power of two, the integers in the range $[t]$ can be uniquely identified by $(\log t)$-bit strings, treating the number $t$ as the all-zero string. For each $j \in [\log t]$, for each $i \in [t]$, do the following. If the $j$-th bit of the number $i$ is a zero, then make all vertices in $B'_i$ adjacent to all vertices in $A'_{0,j}$. If the bit is a one, make $B'_i$ fully adjacent to $A'_{1,j}$ instead.

To conclude the construction, set $k'_A := (\binom{n}{2} - \binom{n/2}{2}) + \binom{n}{2}\log t$ and $k'_B := \frac{n}{2} + (t-1) \cdot n$. It is easy to see that the construction can be carried out in time polynomial in $n$ and $t$, and that $k'_A$ and $k'_B$ satisfy the claimed bounds. It remains to prove the connection between cliques in the input graphs and constrained bipartite vertex covers of $G'$.

▶ **Claim 3.** *If there is an index $i^* \in [t]$ such that $G_{i^*}$ has a clique of size $n/2$, then $G'$ has a $(k'_A, k'_B)$-constrained vertex cover.*

**Proof.** Suppose $G_{i^*}$ contains a clique $D \subseteq [n]$ of size $n/2$. We construct a constrained vertex cover $S$ of $G'$ as follows.

- Add all vertices $b_j$ to $S$ for which $j \in D$. This contributes $n/2$ vertices to $S$.
- Add all vertices $a_{i,j}$ to $S$ for which $i \notin D$ or $j \notin D$, contributing $\binom{n}{2} - \binom{n/2}{2}$ vertices to $S$.
- For each $i \in [t] \setminus \{i^*\}$, add all vertices in $B'_i$ to $S$. This contributes $(t-1) \cdot n$ vertices to $S$.
- For each $j \in [\log t]$, if the $j$-th bit of $i^*$ is a zero then add all vertices of $A'_{0,j}$ to $S$. Otherwise add all vertices of $A'_{1,j}$ to $S$. This contributes $\binom{n}{2}\log t$ vertices to $S$.

It is easy to verify that $S$ contains $k'_A$ vertices from $A'$ and $k'_B$ vertices from $B'$. It remains to check that $S$ is a vertex cover of $G'$.

1. To see that all edges of $G'[A'_c \cup B'_c]$ are covered by $S$, consider an edge between $b_\ell$ and $a_{i,j}$, which exists only if $i = \ell$ or $j = \ell$. If $\ell \in D$ then $b_\ell \in S$ covers the edge. Otherwise, $a_{i,j}$ is contained in $S$ by the second step, and covers the edge.
2. To see that the edges between sets $B'_i$ and $A'_c$ are covered, observe that this trivially holds for all $i \neq i^*$ since $B'_i$ is contained entirely in $S$. For $i^*$ note that $B'_{i^*}$ is only adjacent to vertices $a_{i,j}$ with $1 \leq i < j \leq n$ if $\{i,j\}$ is not an edge of $G_{i^*}$. In this case, we know that $i$ and $j$ are not both contained in the clique $D$ in graph $G_{i^*}$, and therefore $a_{i,j}$ was added to $S$ to cover such edges during the construction of $S$ above.
3. To see that the edges between sets $B'_i$ and $A'_{0/1,j}$ are covered, observe that this trivially holds for all $i \neq i^*$ as $B'_i$ is contained in $S$. For $i^*$ note that the adjacency between $B'_{i^*}$ and $A'_{0/1}$'s follows the binary encoding of the number $i^*$. As we added the sets $A'_{0/1,j}$ to $S$ that match the bit values of $i^*$, all such edges are covered.

Since all edges of $G'$ are covered by $S$, this proves Claim 3. ◀

The next claim establishes several properties of constrained vertex covers in $G'$, leading to a proof that a $(k'_A, k'_B)$-constrained vertex cover implies the existence of a clique of size $n/2$ in one of the input graphs.

▶ **Claim 4.** *For any $(k'_A, k'_B)$-constrained vertex cover $S$ of $G'$, the following holds.*

1. *For every $j \in [\log t]$ the set $S$ contains all vertices of $A'_{0,j}$ or all vertices of $A'_{1,j}$.*
2. *There is an index $i^* \in [t]$ such that $S$ contains all vertices of $B'_i$ for all $i \in [t] \setminus \{i^*\}$.*
3. *There are at least $\binom{n/2}{2}$ distinct vertices $a_{i,j} \in A'_c$ for which $a_{i,j} \notin S$.*
4. *Let $D$ contain the integers $\ell \in [n]$ for which there is a vertex $a_{i,j} \notin S$ with $i = \ell$ or $j = \ell$. Then $|D| = n/2$.*
5. *For every $\{i,j\} \in \binom{D}{2}$ we have $a_{i,j} \notin S$.*
6. *The set $D$ forms a clique of size $n/2$ in the graph $G_{i^*}$, with $i^*$ as in (2).*

**Proof.** Let $S$ be a vertex cover of $G'$ with $|S \cap A'| \leq k'_A$ and $|S \cap B'| \leq k'_B$.

(1) Suppose there is a bit position $j^* \in [\log t]$ such that $S$ avoids both a vertex in $A'_{0,j^*}$ and in $A'_{1,j^*}$. Since every set $B'_i$ for $i \in [t]$ is fully adjacent to one of the sets $A'_{0,j^*}$ or $A'_{1,j^*}$ it follows that to cover such edges the set $S$ contains all vertices of $B'_i$ for all $i \in [t]$. Hence $|S \cap B'| \geq t \cdot n > k'_B$, which is a contradiction.

(2) Suppose there are two indices $i_1, i_2 \in [t]$ such that $S$ avoids both a vertex of $B'_{i_1}$ and of $B'_{i_2}$. Since the numbers $i_1$ and $i_2$ differ, there is an index $j^*$ where their binary representations differ. Since $B'_{i_1}$ is fully adjacent to one of the sets $(A'_{0,j^*}, A'_{1,j^*})$, and $B'_{i_2}$ is fully adjacent to the other set, the fact that $S$ avoids a vertex from both $B'_{i_1}$ and $B'_{i_2}$ implies that the vertex cover $S$ contains all vertices of both $A'_{0,j^*}$ and of $A'_{1,j^*}$, contributing $2\binom{n}{2}$ vertices to $S \cap A'$. By (1), we know that for all $j \in [\log t] \setminus \{j^*\}$ the set $S$ contains at least $\binom{n}{2}$ vertices from $A'_{0,j} \cup A'_{1,j}$. But then $S$ contains at least $2\binom{n}{2} + ((\log t) - 1)\binom{n}{2} > k'_A$ vertices from $A'$, a contradiction.

(3) Since $|S \cap A'| \leq k'_A = (\binom{n}{2} - \binom{n/2}{2}) + \binom{n}{2} \log t$ and (1) shows that $S$ fully contains at least one of the sets $A'_{0,j}$, $A'_{1,j}$ for every $j \in [\log t]$, it follows that $|S \cap (A' \setminus A'_c)| \geq \binom{n}{2} \log t$ and therefore that $S$ contains at most $\binom{n}{2} - \binom{n/2}{2}$ vertices from $A'_c$. As $|A'_c| = \binom{n}{2}$, set $S$ avoids at least $\binom{n/2}{2}$ vertices from $A'_c$.

(4) Every pair $\{i, j\} \in \binom{[n]}{2}$ corresponds to an edge in the complete $n$-vertex graph. For every vertex $a_{i,j} \notin S$, corresponding to an edge $\{i, j\}$, the vertex cover $S$ contains both vertices $b_i$ and $b_j$, since those vertices are adjacent to $a_{i,j}$. Any $\binom{n/2}{2}$ edges span at least $n/2$ endpoints, and there are at least $\binom{n/2}{2}$ pairs represented by members of $A'_c \setminus S$ by (3). It follows that the set $D$ defined in the claim statement has size at least $n/2$, and consequently that $S$ contains at least $n/2$ vertices from $B'_c$. Assume for a contradiction that $|D| > n/2$, implying that $S$ contains more than $n/2$ vertices from $B'_c$. Since $S$ also contains at least $(t-1) \cdot n$ vertices from $\bigcup_{i=1}^{t} B'_i$, by (2), it follows that $|S \cap B'| > k'_B$, a contradiction. Hence $|D| = n/2$.

(5) The definition of $D$ implies that for every $i \in [n] \setminus D$, we have $a_{i,j} \in S$ for all $i < j \leq n$. Put differently, for each $i \notin D$ we know that for each pair $\{i, j\}$ involving $i$ the corresponding vertex $a_{i,j}$ is contained in $S$. Since $|D| = n/2$, there are $\binom{n}{2} - \binom{n/2}{2}$ unordered pairs over $[n]$ involving a vertex not in $D$, and the corresponding $a_{i,j}$ vertices are in $S$ for all these pairs. Since at least $\binom{n/2}{2}$ vertices from $A'_c$ are not in $S$ by (3), it follows that for each pair $\{i, j\} \in \binom{D}{2}$ we must have $a_{i,j} \notin S$.

(6) Assume for a contradiction that $\{i, j\} \in \binom{D}{2}$ with $i < j$ is a pair that is not connected by an edge in $G_{i^*}$. By (5) we have $a_{i,j} \notin S$. Since $\{i, j\}$ is not an edge of $G_{i^*}$, the construction of $G'$ has made all vertices in $B'_{i^*}$ adjacent to vertex $a_{i,j}$. Since $a_{i,j}$ is not in $S$, all vertices of $B'_{i^*}$ must be. But by the choice of $i^*$, all vertices $B'_i$ for $i \in [t] \setminus \{i^*\}$ are also in $S$. Hence $|S \cap B'| \geq t \cdot n > k'_B$, a contradiction. It follows that every pair of vertices in $D$ is connected by an edge in $G_{i^*}$. Since $|D| = n/2$, graph $G_{i^*}$ contains a clique of size $n/2$. ◀

The two claims give the equivalence between the existence of an $n/2$-clique in an input graph and constrained bipartite vertex covers of $G'$. This completes the proof of Lemma 2. ◀

Using Lemma 2 we can prove a lower bound for the number of vertices in kernels for Constrained Bipartite Vertex Cover. We also need the following simplified version of the complementary witness lemma due to Dell and van Melkebeek [8, Lemma 4].

▶ **Lemma 5.** *Let $L, L' \subseteq \Sigma^*$ be two languages. If there is a constant $c$ and a polynomial-time algorithm that, given a list of $t := s^c$ strings $x_1, \ldots, x_t$, each of length at most $s$, outputs a string $x^*$ such that:*

- $x^* \in L' \Leftrightarrow \exists i \in [t] : x_i \in L$, *and*
- $|x^*| \in \mathcal{O}(t \log t)$,

*then* $L \in$ coNP/poly.

As our terminology differs from that of Dell and van Melkebeek, let us point out how the statement above follows from their complementary witness lemma. Dell and van Melkebeek formulate their lemma in terms of (possibly co-nondeterministic) oracle communication protocols. These are two-player protocols in which the player holding the inputs is restricted to polynomial-time computation and the other player is computationally unbounded but does not know the input. The goal is for the first player to correctly decide whether there is at least one input that belongs to $L$, using as little communication as possible to the second player. The connection comes from the fact that a polynomial-time algorithm as described in Lemma 5 easily gives such a protocol: the first player runs the algorithm on its inputs to obtain the instance $x^*$ that expresses the logical OR of his inputs, sends this small instance to the oracle, which sends back one bit that tells whether or not $x^*$ is contained in $L'$. Using Lemmata 2 and 5 we now prove the vertex lower bound for Constrained Bipartite Vertex Cover kernelization.

▶ **Theorem 6.** *Let $\varepsilon > 0$ be a real number. If there is a polynomial-time algorithm that reduces any instance $(G, A, B, k_A, k_B)$ of* Constrained Bipartite Vertex Cover *to an equivalent instance $(G', A', B', k'_A, k'_B)$ of the same problem, such that $k'_A \in (k_A)^{\mathcal{O}(1)}, k'_B \in (k_B)^{\mathcal{O}(1)}$, and $|V(G')| \in \mathcal{O}((k_A \cdot k_B)^{1-\varepsilon})$, then* NP $\subseteq$ coNP/poly.

**Proof.** Assume such a kernelization algorithm exists and call it $\mathcal{K}$. Let $\frac{n}{2}$-Clique be the problem of deciding whether a graph of even order has a clique containing exactly half of its vertices. An easy padding argument proves that $\frac{n}{2}$-Clique is NP-complete. We will show that, using $\mathcal{K}$ and the construction of Lemma 2, we can make an algorithm that compresses the logical OR of a series of $\frac{n}{2}$-Clique instances into an equivalent instance of Constrained Bipartite Vertex Cover whose size satisfies the requirements of Lemma 5. This will show that $\frac{n}{2}$-Clique is contained in coNP/poly and yield NP $\subseteq$ coNP/poly.

Let $L$ be the language over alphabet $\{0, 1\}$ such that a string $x$ is contained in $L$ if and only if it encodes the adjacency matrix of a graph with an even number $n$ of vertices that has a clique of size $n/2$. We construct an algorithm $\mathcal{R}$ that satisfies the conditions of Lemma 5 with this $L$, while $L'$ is the (classical) language encoding Constrained Bipartite Vertex Cover. The value of $c$ depends on $\epsilon$ and will be specified later. On input a list of strings $x_1, \ldots, x_t$, where each string has length at most $s$ and $t = s^c$, algorithm $\mathcal{R}$ proceeds as follows. It first checks which strings encode adjacency matrices of undirected graphs and throws away the others. If the resulting number of instances is not a power of two, it duplicates one instance until the nearest power of two is reached. After this step the input consists of $t' \leq 2t = 2s^c$ strings $x_1, \ldots, x_{t'}$ that encode graphs $G_1, \ldots, G_{t'}$ of at most $n := \lfloor \sqrt{s} \rfloor$ vertices each. As the next step, we pad the instances to ensure they all have the same size. For each input, while it has less than $n$ vertices, add both an isolated vertex and a universal vertex to the graph. This increases the maximum clique size by exactly one, and the graph size by two, so that the new graph has a clique of half its vertices if and only if the original graph has one. We obtain a series of $t'$ graphs $G_1, \ldots, G_{t'}$, each on exactly $n$ vertices, in which the goal is to detect a clique of size $n/2$.

We invoke the construction of Lemma 2 to the graphs $G_1, \ldots, G_{t'}$ and obtain an instance $(G, A, B, k_A, k_B)$ of Constrained Bipartite Vertex Cover of size polynomial in $t'$ and $n$, such that $k_A \in \mathcal{O}(n^2 \log t')$ and $k_B \in \mathcal{O}(n \cdot t')$, which is a yes-instance if and only if one of the inputs contains a clique of size $n/2$. Now we apply the hypothetical kernel $\mathcal{K}$ on the instance $(G, A, B, k_A, k_B)$ to obtain an equivalent instance $(G', A', B', k'_A, k'_B)$ of Constrained Bipartite Vertex Cover in which the two parameters have grown only polynomially, i.e., $k'_A \in \mathcal{O}((n^2 \log t')^{\alpha_1})$ and $k'_B \in \mathcal{O}((n \cdot t')^{\alpha_2})$, such that $|V(G')|$ is bounded by $\mathcal{O}((k_A \cdot k_B)^{1-\varepsilon})$ for some fixed $\varepsilon > 0$. The crucial following step is to apply the reduction rule from the simple kernel on this reduced instance: while there is a vertex in $B'$ of degree more than $k'_A$, we delete it from the graph and decrease the budget by one. Similarly, remove all isolated vertices. Let $(G^*, A^*, B^*, k^*_A, k^*_B)$ be the resulting exhaustively reduced instance. Since the number of vertices does not increase by this step, we know that $|V(G^*)| \in \mathcal{O}((k_A \cdot k_B)^{1-\varepsilon})$. More importantly, we also get a bound on the number of edges. The edges of a bipartite graph can be counted by summing the degrees of all vertices in one partite set. The reduction rule ensures that all vertices of $B^*$ have degree at most $k^*_A \leq k'_A$. We therefore find that the number of edges in $G^*$ is bounded by:

$$\begin{aligned}
\mathcal{O}(|V(G^*)| \cdot k'_A) &\in \mathcal{O}((k_A \cdot k_B)^{1-\varepsilon} \cdot (n^2 \log t')^{\alpha_1}) \\
&\in \mathcal{O}((n^2 \log t' \cdot n \cdot t')^{1-\varepsilon} \cdot (n^2 \log t')^{\alpha_1}) \\
&\in \mathcal{O}((s \log(2s^c) \cdot \sqrt{s} \cdot (2s^c))^{1-\varepsilon} \cdot (s \log(2s^c))^{\alpha_1}) \qquad t' \leq 2s^c, n \leq \sqrt{s}. \\
&\in \mathcal{O}((s^{1.5+c} \cdot c \cdot \log s)^{1-\varepsilon} \cdot (s^{\alpha_1} \cdot c^{\alpha_1} \cdot \log^{\alpha_1} s)) \\
&\in \mathcal{O}((s^{(1-\varepsilon)(1.5+c)+\alpha_1}) \cdot c^{1+\alpha_1} \cdot \log^{(1-\varepsilon)+\alpha_1} s).
\end{aligned}$$

The derivation shows that if we choose $c := \lceil (2.5 + \alpha_1)/\varepsilon \rceil$ (implying that $s^{(1-\varepsilon)(1.5+c)+\alpha_1} < s^{c-1}$), the number of edges in the final graph is $\mathcal{O}(s^{c-1} \cdot \log^{\mathcal{O}(1)} s)$. As $G^*$ has no isolated vertices, the same bound applies to the number of vertices. The instance of Constrained Bipartite Vertex Cover that results from the procedure can be encoded as a string $x^*$ using an adjacency list, which requires $\mathcal{O}(|E(G^*)| \cdot \log |V(G^*)|)$ bits. The string $x^*$ is given as the output. Tracing back the chain of equivalences, we know that $x^*$ is a yes-instance of Constrained Bipartite Vertex Cover if and only if there is string $x_i$ that is a yes-instance of $\frac{n}{2}$-Clique. It is easy to verify that the suggested algorithm $\mathcal{R}$ takes polynomial time. Using the bounds obtained above we find that $|x^*| \in \mathcal{O}(s^{c-1}(\log s)^{\mathcal{O}(1)})$, which is $\mathcal{O}(s^c)$. Hence our choice of $c$ makes algorithm $\mathcal{R}$ satisfy all requirements of Lemma 5, proving that $\frac{n}{2}$-Clique is in coNP/poly and therefore that NP $\subseteq$ coNP/poly. Theorem 6 follows. ◀

## 4 Sparsification Lower Bound

We establish a sparsification lower bound for the parameterization of Constrained Bipartite Vertex Cover by the total number of vertices in the graph. From this, a general lower bound on the number of edges (which also holds against kernels that increase the budget values arbitrarily) will follow as an easy corollary. We employ the cross-composition framework by Bodlaender et al. [3], which builds on earlier work by several authors [2, 8, 13]. The following two definitions form the core of the framework.

▶ **Definition 7** (Polynomial equivalence relation). An equivalence relation $\mathcal{R}$ on $\Sigma^*$ is called a *polynomial equivalence relation* if the following conditions hold:

1. There is an algorithm that, given two strings $x, y \in \Sigma^*$, decides whether $x$ and $y$ belong to the same equivalence class in time polynomial in $|x| + |y|$.
2. For any finite set $S \subseteq \Sigma^*$ the equivalence relation $\mathcal{R}$ partitions the elements of $S$ into a number of classes that is polynomially bounded in the size of the largest element of $S$.

▶ **Definition 8** (Cross-composition). Let $L \subseteq \Sigma^*$ be a language, let $\mathcal{R}$ be a polynomial equivalence relation on $\Sigma^*$, let $\mathcal{Q} \subseteq \Sigma^* \times \mathbb{N}$ be a parameterized problem, and let $f \colon \mathbb{N} \to \mathbb{N}$ be a function. An OR-*cross-composition of $L$ into $\mathcal{Q}$* (with respect to $\mathcal{R}$) *of cost $f(t)$* is an algorithm that, given $t$ instances $x_1, x_2, \ldots, x_t \in \Sigma^*$ of $L$ belonging to the same equivalence class of $\mathcal{R}$, takes time polynomial in $\sum_{i=1}^{t} |x_i|$ and outputs an instance $(y, k) \in \Sigma^* \times \mathbb{N}$ such that:

- The parameter $k$ is bounded by $\mathcal{O}(f(t) \cdot (\max_i |x_i|)^c)$, where $c$ is some constant independent of $t$.
- $(y, k) \in \mathcal{Q}$ if and only if there is an $i \in [t]$ such that $x_i \in L$.

The following theorem shows how these concepts give kernelization lower bounds.

▶ **Theorem 9** ([3, Theorem 6]). *Let $L \subseteq \Sigma^*$ be a language, let $\mathcal{Q} \subseteq \Sigma^* \times \mathbb{N}$ be a parameterized problem, and let $d, \varepsilon$ be positive reals. If $L$ is* NP*-hard under Karp reductions, has an* OR*-cross-composition into $\mathcal{Q}$ with cost $f(t) = t^{1/d+o(1)}$, where $t$ denotes the number of instances, and $\mathcal{Q}$ has a polynomial (generalized) kernelization with size bound $\mathcal{O}(k^{d-\varepsilon})$, then* NP $\subseteq$ coNP/poly.

We use a restricted version of Constrained Bipartite Vertex Cover as the source problem in a cross-composition:

Equally Constrained Bipartite Vertex Cover
**Input:** A bipartite graph $G$ with partite sets $A$ and $B$ such that $|A| = |B|$ is even.
**Question:** Is there a vertex cover $S$ for $G$ such that $|S \cap A| \leq |A|/2$ and $|S \cap B| \leq |B|/2$?

▶ **Proposition 10.** Equally Constrained Bipartite Vertex Cover *is* NP*-complete.*

Proposition 10 follows from a simple padding argument.

▶ **Theorem 11.** Constrained Bipartite Vertex Cover *parameterized by the number of vertices $n$ does not have a generalized kernel of bitsize $\mathcal{O}(n^{2-\varepsilon})$, for any $\varepsilon > 0$, unless* NP $\subseteq$ coNP/poly.

**Proof.** With the aim of giving a cross-composition, we start by defining a polynomial equivalence relation $\mathcal{R}$ on inputs of Equally Constrained Bipartite Vertex Cover. We define any two strings that do not encode valid instances of Equally Constrained Bipartite Vertex Cover to be equivalent. Two well-formed instances $(G_1, A_1, B_1)$ and $(G_2, A_2, B_2)$ are equivalent if and only if $|A_1| = |B_1| = |A_2| = |B_2|$. It is easy to verify that this is a polynomial equivalence relation. We proceed to give a cross-composition of cost $\mathcal{O}(\sqrt{t} \cdot \log t) \in \mathcal{O}(t^{\frac{1}{2}+o(1)})$ from Equally Constrained Bipartite Vertex Cover into Constrained Bipartite Vertex Cover parameterized by the number of vertices $n$. Strings that do not encode valid inputs can be recognized in polynomial time, and can be reduced to a trivial NO-instance. In the remainder, we focus on the case that the input encodes a list of instances $(G_1, A_1, B_1), \ldots, (G_t, A_t, B_t)$ such that all partite sets of all input graphs have the same number of $n$ vertices. If $t$ is not equal to $2^{2r}$ for some integer $r$, then we can repeatedly duplicate an instance until this holds. This only blows up the size of the input by a constant factor and does not change whether there is at least one YES-instance in the inputs. In the remainder we can assume that $t = 2^{2r}$ for some $r$, implying that both $\sqrt{t}$ and $\log \sqrt{t}$ are integers. We can therefore index the inputs as $(G_{i,j}, A_{i,j}, B_{i,j})$ for $i, j \in [\sqrt{t}]$. For ease of presentation, the vertices in each partite set are identified with the integers in the range $[n]$. We can assume $n \geq 3$, as the instances are trivially solvable otherwise. We construct an instance $(G', A', B', k'_A, k'_B)$ of Constrained Bipartite Vertex Cover that expresses the logical OR of the input instances, as follows.

1. For each $i \in [\sqrt{t}]$, add a vertex set $A'_i$ consisting of $n$ vertices $a_{i,1}, \ldots, a_{i,n}$ to $A'$.
2. For each $i \in [\sqrt{t}]$, add a vertex set $B'_i$ consisting of $n$ vertices $b_{i,1}, \ldots, b_{i,n}$ to $B'$.
3. The adjacency information of the input graphs is embedded into $G'$. For $i, j \in [\sqrt{t}]$, for $p, q \in [n]$, do the following. If $\{p, q\} \in E(G_{i,j})$, then make vertex $a_{i,p}$ adjacent to $b_{j,q}$. Afterward we have for every $i, j \in [\sqrt{t}]$ that the graph $G'[A'_i \cup B'_j]$ is isomorphic to $G_{i,j}$.
4. For each $i \in [\sqrt{t}]$, add a vertex set $C'_i$ consisting of $n/2 - 1$ *checking vertices* to the partite set $A'$. Make all vertices of $C'_i$ adjacent to all vertices of $B'_i$.
5. Define $w := \sqrt{t} \cdot n^2$. For each $j \in [\log \sqrt{t}]$, for each $x \in \{0, 1\}$, add a vertex set $B'_{x,j}$ to the partite set $B'$ consisting of $w$ vertices. For all $i \in [\sqrt{t}]$, make all vertices of $B'_{x,j}$ adjacent to all vertices of $A'_i$ if the $j$-th bit of the binary representation of number $i$ is an $x$.

This concludes the description of the graph $G'$. It is easy to verify that $n' := |V(G')| = 2n\sqrt{t} + (n/2 - 1)\sqrt{t} + 2w \log \sqrt{t} \in \mathcal{O}(n^2\sqrt{t} \log t)$. The construction can be performed in polynomial time. Define $k'_A := n\sqrt{t} - 1$ and $k'_B := n/2 + (\sqrt{t} - 1) \cdot n + w \cdot \log \sqrt{t}$. We will prove that $(G', A', B', k'_A, k'_B)$ is a YES-instance of CONSTRAINED BIPARTITE VERTEX COVER if and only if one of the inputs is a YES-instance.

▶ **Claim 12.** *If there are indices $i^*, j^* \in [\sqrt{t}]$ such that $G_{i^*,j^*}$ has an $(n/2, n/2)$-constrained vertex cover, then $G'$ has a $(k'_A, k'_B)$-constrained vertex cover.*

**Proof.** Suppose there are sets $A^* \subseteq A_{i^*,j^*}$ and $B^* \subseteq B_{i^*,j^*}$, each of size at most $n/2$, which together form a vertex cover of $G_{i^*,j^*}$. We build a $(k'_A, k'_B)$-constrained vertex cover $S$ for $G'$.

- Add all vertices $a_{i^*,\ell}$ to $S$ for which $\ell \in A^*$. This contributes $n/2$ vertices to $S \cap A'$.
- Add all vertices $b_{j^*,\ell}$ to $S$ for which $\ell \in B^*$. This contributes $n/2$ vertices to $S \cap B'$.
- Add all vertices of all sets $A'_i$ with $i \in [\sqrt{t}] \setminus \{i^*\}$ to $S$. This contributes $(\sqrt{t} - 1)n$ vertices to $S \cap A'$.
- Add all vertices of all sets $B'_j$ with $j \in [\sqrt{t}] \setminus \{j^*\}$ to $S$. This contributes $(\sqrt{t} - 1)n$ vertices to $S \cap B'$.
- Add all vertices $C'_{j^*}$ to $S$. This contributes $n/2 - 1$ vertices to $S \cap A'$.
- For each $j \in [\log \sqrt{t}]$, if the $j$-th bit of $i^*$ is a zero then add all vertices of $B'_{0,j}$ to $S$. Otherwise add all vertices of $B'_{1,j}$ to $S$. This contributes $w \log \sqrt{t}$ vertices to $S \cap B'$.

It is easy to verify that $S$ contains $k'_A$ vertices from $A'$ and $k'_B$ vertices from $B'$. It remains to check that $S$ is a vertex cover of $G'$. We discuss the edges of $G'$ in the order in which they were added to the graph by the construction.

- The edges of the induced subgraph $G'[A'_{i^*} \cup B'_{j^*}]$ are covered since $S$ includes the vertices corresponding to $A^*$ and $B^*$, which form a vertex cover for $G_{i^*,j^*}$. Recall that $G_{i^*,j^*}$ is isomorphic to $G'[A'_{i^*} \cup B'_{j^*}]$. Edges between $A'_i$ and $B'_j$ for $i \neq i^*$ or $j \neq j^*$ are covered because $S$ includes all vertices of $A'_i$ for $i \neq i^*$, and all vertices of $B'_j$ for $j \neq j^*$.
- The edges between the checking vertices $C'_{j^*}$ and $B'_{j^*}$ are covered because $S$ includes $C'_{j^*}$. The edges between $C'_j$ and $B'_j$ for $j \neq j^*$ are covered because $S$ contains $B'_j$.
- The edges between $A'_{i^*}$ and sets $B'_{x,j}$ for $x \in \{0, 1\}$, $j \in [\log \sqrt{t}]$ are covered because $S$ contains all sets $B'_{x,j}$ whose bit value matches that of the binary representation of $i^*$. For $i \neq i^*$, the edges between $A'_i$ and sets $B'_{x,j}$ are covered because $S$ contains $A'_i$.

As $S$ covers all edges of $G'$, the claim follows. ◀

▶ **Claim 13.** *For any inclusionwise-minimal $(k'_A, k'_B)$-constrained vertex cover $S$ of $G'$, the following holds.*

1. *For every $j \in [\log \sqrt{t}]$ the set $S$ contains all vertices of $B'_{0,j}$ or all vertices of $B'_{1,j}$.*
2. *For every $j \in [\log \sqrt{t}]$ we have $S \cap B'_{0,j} = \emptyset$ or $S \cap B'_{1,j} = \emptyset$.*
3. *There is an index $\ell \in [\sqrt{t}]$ such that $B'_\ell \setminus S \neq \emptyset$.*
4. *The set $S$ avoids at least $n/2$ vertices from the set $\bigcup_{i \in [\sqrt{t}]} A'_i$.*

5. *There is an index $i^* \in [\sqrt{t}]$ such that $S$ contains $A'_i$ for all $i \in [\sqrt{t}] \setminus \{i^*\}$.*
6. *There is an index $j^* \in [\sqrt{t}]$ such that $S$ contains $B'_j$ for all $j \in [\sqrt{t}] \setminus \{j^*\}$.*
7. *Let $i^*$ and $j^*$ be as defined in (5) and (6). Then the graph $G_{i^*,j^*}$ has a vertex cover containing at most $n/2$ vertices from each partite set.*

**Proof.** Let $S$ be an inclusionwise-minimal $(k'_A, k'_B)$-constrained vertex cover of $G'$.

(1) Assume for a contradiction that there is an index $j \in [\log \sqrt{t}]$ such that $S$ avoids both a vertex of $B'_{0,j}$ and of $B'_{1,j}$. To cover all the edges between $B'_{0,j}$ and sets $A'_i$ for $i \in [\sqrt{t}]$, the set $S$ must contain all sets $A'_i$ for which the $j$-th bit of $i$ is a zero. Similarly, $S$ contains all sets $A'_i$ for which the $j$-th bit is a one, to cover the edges between $B'_{0,j}$ and the $A'_i$'s. So $S \supseteq \bigcup_{i \in [\sqrt{t}]} A'_i$, showing that $|S \cap A'| \geq n \cdot \sqrt{t} > k'_A$, a contradiction.

(2) Assume for a contradiction that there is an index $j \in [\log \sqrt{t}]$ such that $S \cap B'_{0,j} \neq \emptyset$ and $S \cap B'_{1,j} \neq \emptyset$. Observe that all vertices in $B'_{x,j}$ are false twins for $x \in \{0,1\}$. Hence if one vertex $v$ of $B'_{x,j}$ is contained in the inclusionwise-minimal vertex cover $S$, then there is a neighbor $u$ of $v$ that is not in $S$, implying that all other vertices of $B'_{x,j}$ are also adjacent to $u$. Therefore all vertices of $B'_{0,j}$ and $B'_{1,j}$ are contained in $S$. Together with the fact that $S$ contains at least one of the sets $B'_{0,j'}, B'_{1,j'}$ fully for all $j' \in [\sqrt{t}] \setminus \{j\}$, by (1), we find that $|S \cap B'| \geq (\log \sqrt{t} - 1)w + 2w = w + w \cdot \log \sqrt{t} = n^2\sqrt{t} + w \cdot \log \sqrt{t} > k'_B$; a contradiction.

(3) Suppose that $B'_\ell \subseteq S$ for all $\ell \in [\sqrt{t}]$, contributing $n \cdot \sqrt{t}$ vertices to $S \cap B'$. By (1) we know that $S$ contains at least $w \cdot \log \sqrt{t}$ vertices from the sets $B'_{x,i}$. Hence $|S \cap B'| \geq n \cdot \sqrt{t} + w \cdot \log \sqrt{t} > k'_B$, a contradiction.

(4) Suppose that $|\bigcup_{i \in [\sqrt{t}]} A'_i \setminus S| < n/2$. Then $|\bigcup_{i \in [\sqrt{t}]} A'_i \cap S| > n \cdot \sqrt{t} - n/2$. By (3) we know that there is an index $i^* \in [\sqrt{t}]$ such that $S$ avoids at least one vertex from $B'_{i^*}$. Since all vertices of $C'_{i^*}$ are adjacent to all vertices of $B'_{i^*}$, this implies that $S$ contains all $n/2 - 1$ vertices of $C'_{i^*}$. Since $C'_{i^*}$ also belongs to the $A'$ partite set, this shows that $|S \cap A'| > (n\sqrt{t} - n/2) + (n/2 - 1) = n\sqrt{t} - 1 = k'_A$, a contradiction.

(5) Consider the number $i^*$ whose $j$-th bit is a one if $S$ avoids $B'_{0,j}$ and is a zero otherwise. By (2), in the second case $S$ avoids $B'_{1,j}$. For each $i \neq i^*$, the binary representation of $i$ differs from that of $i^*$ in at least one bit. Suppose that the $j$-th bit of $i$ is a zero, and the $j$-th bit of $i^*$ is a one. Then $S$ avoids $B'_{0,j}$ and $A'_i$ is adjacent to $B'_{0,j}$ by construction. Consequently, all vertices of $A'_i$ are contained in $S$. Similarly, if the $j$-th bit of $i$ is a one and the $j$-th bit of $i^*$ is a zero, then $S$ avoids $B'_{1,j}$ while $B'_{1,j}$ is adjacent to $A'_i$; hence $A'_i$ is fully contained in $S$. It follows that for every index $i \neq i^*$ the set $A'_i$ is contained in $S$.

(6) Suppose that there are two indices $j', j''$ such that $S$ avoids a vertex of both $B'_{j'}$ and $B'_{j''}$. Then $S$ contains all vertices of $C'_{j'}$ and $C'_{j''}$. By (5) we know that there is an index $i^* \in [\sqrt{t}]$ such that $S$ contains $\bigcup_{i \in [\sqrt{t}] \setminus \{i^*\}} A'_i$. Hence $|S \cap A'| \geq 2(n-1) + (\sqrt{t} - 1) \cdot n > k'_A$, a contradiction. The last step uses the fact that $n \geq 3$.

(7) Let $i^*$ and $j^*$ be as defined in (5) and (6), implying that $S$ contains all sets $A'_i$ for $i \in [\sqrt{t}] \setminus \{i^*\}$ and all sets $B'_j$ for $j \in [\sqrt{t}] \setminus \{j^*\}$. By (4) the set $S$ avoids at least $n/2$ vertices from $\bigcup_{i \in [\sqrt{t}]} A'_i$, which implies that $|S \cap A'_{i^*}| \leq n/2$, since $S$ fully contains the other sets $A'_i$. We proceed to show that $|S \cap B'_{j^*}| \leq n/2$. To see that, observe that $S$ fully contains $(\sqrt{t} - 1) \cdot n$ vertices from $\bigcup_{j \in [\sqrt{t}] \setminus \{j^*\}} B'_j$. In addition, $S$ contains at least $w \cdot \log \sqrt{t}$ vertices from $\bigcup_{j \in [\log \sqrt{t}], x \in \{0,1\}} B'_{x,j}$, by (1). Since the total size of $S \cap B'$ is at most $k'_B = n/2 + (\sqrt{t} - 1) \cdot n + w \cdot \log \sqrt{t}$ we find that $|S \cap B'_{j^*}| \leq n/2$. Now define $S^* := S \cap (A'_{i^*} \cup B'_{j^*})$. It follows that $S^*$ is a vertex cover of the graph $G'[A'_{i^*} \cup B'_{j^*}]$ containing at most $n/2$ vertices from each partite set. Since $G'[A'_{i^*} \cup B'_{j^*}]$ is isomorphic to the input graph $G_{i^*,j^*}$ by construction of $G'$, the claim follows. ◀

The last item of Claim 13 shows that if there is a $(k'_A, k'_B)$-constrained vertex cover of $G'$, then one of the input instances has answer YES: if a vertex cover with such size constraints exists, then there is also an inclusionwise-minimal vertex cover satisfying the same size bounds, causing one of the input graphs to have a vertex cover containing at most $n/2$ vertices from each partite set. By the definition of EQUALLY CONSTRAINED BIPARTITE VERTEX COVER, this certifies the YES-answer for that input. Together with the first claim, we have therefore established that the composed instance acts as the logical OR of the inputs. The construction thus satisfies all requirements of a cross-composition of EQUALLY CONSTRAINED BIPARTITE VERTEX COVER into CONSTRAINED BIPARTITE VERTEX COVER parameterized by the number of vertices. The parameter of the composed instance is $n' \in \mathcal{O}(n^2\sqrt{t}\log t) \in \mathcal{O}(n^{O(1)} \cdot t^{\frac{1}{2}+o(1)})$, showing that the cross-composition has cost $f(t) \in \mathcal{O}(t^{\frac{1}{2}+o(1)})$. As the starting problem is NP-complete (Proposition 10), Theorem 11 now follows from Theorem 9. ◀

▶ **Corollary 14.** *Let $\varepsilon > 0$ be a real number. If there is a polynomial-time algorithm that reduces any instance $(G, A, B, k_A, k_B)$ of* CONSTRAINED BIPARTITE VERTEX COVER *to an equivalent instance of the same problem with $\mathcal{O}((k_A \cdot k_B)^{1-\varepsilon})$ edges, then* NP $\subseteq$ coNP/poly*.*

**Proof.** Suppose such a kernelization algorithm exists and call it $\mathcal{K}$. Using $\mathcal{K}$ we create a generalized kernelization $\mathcal{A}$ of subquadratic size for CONSTRAINED BIPARTITE VERTEX COVER parameterized by the number of vertices $n$. Presented with an instance $(G, A, B, k_A, k_B)$, the algorithm does the following. Let $n$ be the number of vertices in $G$. If $k_A \geq n$ or $k_B \geq n$, then the answer is trivially YES as we may take all of $A$ or all of $B$ to form the desired constrained vertex cover. We can therefore output a constant-size YES-instance as the output of the compression. In the remaining cases we know $k_A, k_B < n$. The algorithm then invokes $\mathcal{K}$ on the input, obtaining an equivalent instance $(G', A', B', k'_A, k'_B)$ where $|E(G')| \in \mathcal{O}((k_A \cdot k_B)^{1-\varepsilon}) \in \mathcal{O}((n \cdot n)^{1-\varepsilon}) \in \mathcal{O}(n^{2-2\varepsilon})$. After removing isolated vertices from the graph, which do not affect the answer, the number of vertices in $G'$ is at most twice the number of edges, which is $\mathcal{O}(n^{2-2\varepsilon})$. This instance is encoded using an adjacency list representation. In general, an adjacency list encoding of a graph uses $\mathcal{O}(|V| + |E| \log |V|)$ bits. In this case, we find that $G'$ can be encoded in $\mathcal{O}(n^{2-2\varepsilon}\log(n^{2-2\varepsilon})) \in \mathcal{O}(n^{2-\varepsilon})$ bits. After encoding the values of $k'_A$ and $k'_B$ in binary, which does not exceed this space bound, the algorithm outputs the resulting instance. Since it is equivalent to the input instance, this is a generalized kernel for CONSTRAINED BIPARTITE VERTEX COVER. As the size is $\mathcal{O}(n^{2-\varepsilon})$ for some positive $\varepsilon$, by Theorem 11 we now obtain NP $\subseteq$ coNP/poly. ◀

## 5 Conclusion

In this paper we presented two kernelization lower bounds for CONSTRAINED BIPARTITE VERTEX COVER. We proved that it is unlikely that the $\Theta(k_A \cdot k_B)$ bound on the number of vertices and edges from the easy kernel can be significantly improved. The easy kernel is therefore close to optimal when bounding the kernel size in terms of the product $k_A \cdot k_B$. Our results do not rule out the existence of kernels for CONSTRAINED BIPARTITE VERTEX COVER with $\mathcal{O}(k_A + k_B)$ vertices, which we leave as an open problem.

**Acknowledgments.** The author is grateful to Marcin and Michał Pilipczuk for bringing the problem to his attention.

## References


1 Guoqiang Bai and Henning Fernau. Constraint bipartite vertex cover: simpler exact algorithms and implementations. *J. Comb. Optim.*, 23(3):331–355, 2012. `doi:10.1007/s10878-010-9289-7`.

2 Hans L. Bodlaender, Rodney G. Downey, Michael R. Fellows, and Danny Hermelin. On problems without polynomial kernels. *J. Comput. Syst. Sci.*, 75(8):423–434, 2009. `doi:10.1016/j.jcss.2009.04.001`.

3 Hans L. Bodlaender, Bart M. P. Jansen, and Stefan Kratsch. Kernelization lower bounds by cross-composition. *SIAM J. Discrete Math.*, 28(1):277–305, 2014. `doi:10.1137/120880240`.

4 Jianer Chen and Iyad A. Kanj. Constrained minimum vertex cover in bipartite graphs: complexity and parameterized algorithms. *J. Comput. Syst. Sci.*, 67(4):833–847, 2003. `doi:10.1016/j.jcss.2003.09.003`.

5 Jianer Chen, Iyad A. Kanj, and Weijia Jia. Vertex cover: Further observations and further improvements. *J. Algorithms*, 41(2):280–301, 2001. `doi:10.1006/jagm.2001.1186`.

6 Marek Cygan, Fedor V. Fomin, Lukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michal Pilipczuk, and Saket Saurabh. *Parameterized Algorithms*. Springer, 2015. `doi:10.1007/978-3-319-21275-3`.

7 Holger Dell and Dániel Marx. Kernelization of packing problems. In *Proc. 23rd SODA*, pages 68–81, 2012. `doi:10.1137/1.9781611973099.6`.

8 Holger Dell and Dieter van Melkebeek. Satisfiability allows no nontrivial sparsification unless the polynomial-time hierarchy collapses. *J. ACM*, 61(4):23:1–23:27, 2014. `doi:10.1145/2629620`.

9 Rodney G. Downey and Michael R. Fellows. *Fundamentals of Parameterized Complexity*. Texts in Computer Science. Springer, 2013.

10 R.C. Evans. Testing repairable RAMs and mostly good memories. In *Proc. IEEE International Test Conference*, pages 49–55, 1981.

11 Michael R. Fellows, Bart M. P. Jansen, and Frances Rosamond. Towards fully multivariate algorithmics: Parameter ecology and the deconstruction of computational complexity. *European J. Combin.*, 34(3):541–566, 2013. `doi:10.1016/j.ejc.2012.04.008`.

12 Henning Fernau and Rolf Niedermeier. An efficient exact algorithm for constraint bipartite vertex cover. *J. Algorithms*, 38(2):374–410, 2001. `doi:10.1006/jagm.2000.1141`.

13 Lance Fortnow and Rahul Santhanam. Infeasibility of instance compression and succinct PCPs for NP. *J. Comput. Syst. Sci.*, 77(1):91–106, 2011. `doi:10.1016/j.jcss.2010.06.007`.

14 Danny Hermelin and Xi Wu. Weak compositions and their applications to polynomial lower bounds for kernelization. In *Proc. 23rd SODA*, pages 104–113, 2012. `doi:10.1137/1.9781611973099.9`.

15 Bart M. P. Jansen. On sparsification for computing treewidth. *Algorithmica*, 71(3):605–635, 2015. `doi:10.1007/s00453-014-9924-2`.

16 Stefan Kratsch, Geevarghese Philip, and Saurabh Ray. Point line cover: The easy kernel is essentially tight. In *Proc. 25th SODA*, pages 1596–1606. SIAM, 2014. `doi:10.1137/1.9781611973402.116`.

17 S.-Y. Kuo and W.K. Fuchs. Efficient spare allocation for reconfigurable arrays. *IEEE Design Test of Computers*, 4(1):24–31, 1987. `doi:10.1109/MDT.1987.295111`.

18 Marcin Pilipczuk. Banach's algorithmic corner. http://corner.mimuw.edu.pl/?p=539, October 2013.

19 Chee-Keng Yap. Some consequences of non-uniform conditions on uniform classes. *Theor. Comput. Sci.*, 26:287–300, 1983. `doi:10.1016/0304-3975(83)90020-8`.