# On the Parameterized Complexity of Biclique Cover and Partition*

**Sunil Chandran[1], Davis Issac[2], and Andreas Karrenbauer[3]**

1 **Indian Institute of Science, Bangalore, India**
2 **Max Planck Institute for Informatics, Saarland Informatics Campus, Saarbrücken, Germany**
3 **Max Planck Institute for Informatics, Saarland Informatics Campus, Saarbrücken, Germany**

## Abstract

Given a bipartite graph $G$, we consider the decision problem called BicliqueCover for a fixed positive integer parameter $k$ where we are asked whether the edges of $G$ can be covered with at most $k$ complete bipartite subgraphs (a.k.a. bicliques). In the BicliquePartition problem, we have the additional constraint that each edge should appear in exactly one of the $k$ bicliques. These problems are both known to be NP-complete but fixed parameter tractable. However, the known FPT algorithms have a running time that is doubly exponential in $k$, and the best known kernel for both problems is exponential in $k$. We build on this kernel and improve the running time for BicliquePartition to $\mathcal{O}^*(2^{2k^2+k\log k+k})$ by exploiting a linear algebraic view on this problem. On the other hand, we show that no such improvement is possible for BicliqueCover unless the Exponential Time Hypothesis (ETH) is false by proving a doubly exponential lower bound on the running time. We achieve this by giving a reduction from 3SAT on $n$ variables to an instance of BicliqueCover with $k = \mathcal{O}(\log n)$. As a further consequence of this reduction, we show that there is no subexponential kernel for BicliqueCover unless $P = NP$. Finally, we point out the significance of the exponential kernel mentioned above for the design of polynomial-time approximation algorithms for the optimization versions of both problems. That is, we show that it is possible to obtain approximation factors of $\frac{n}{\log n}$ for both problems, whereas the previous best approximation factor was $\frac{n}{\sqrt{\log n}}$.

**1998 ACM Subject Classification** F.2.2 Nonnumerical Algorithms and Problems

**Keywords and phrases** Biclique Cover/Partition, Linear algebra in finite fields, Lower bound

**Digital Object Identifier** 10.4230/LIPIcs.IPEC.2016.11

## 1 Introduction

The problems of covering or partitioning the edge set of bipartite graphs have a long history. It was shown by Orlin in 1977 that the covering problem, i.e., to decide for a given bipartite graph $G$ and given integer $k$, whether the edges of $G$ can be covered by at most $k$ complete bipartite subgraphs (also known as bicliques), is NP-complete [13]. He conjectured the partitioning problem, i.e., where each edge of $G$ must appear in exactly one of the $k$ bicliques, to also be NP-complete, which has since been answered in the affirmative in [10].

The minimum number of bicliques to cover the edges of a graph is also called the *bipartite dimension* [5], and Orlin called the minimum $k$ that admits a partition of the edge set into $k$

* This work is supported in part by DFG-grant KA 3042/3-1 and the Max Planck Center for Visual Computing and Communication (`http://www.mpc-vcc.org`).

bicliques the *bicontent* [13]. We prefer the terms *biclique cover number* and *biclique partition number*, respectively, to avoid any confusion.

There are numerous applications for biclique covering and partitioning. Related work can be found in the areas of bioinformatics [11, 12], computer security [3], database tiling [7], finite automata [9], and graph drawing [4].

Besides these applications, computing a biclique cover of a graph is equivalent to other important notions in mathematics: Given an $m$-by-$n$ matrix $A$, the *Boolean rank* of $A$ is the minimum $k$ for which there exist two 0-1 matrices $B$ and $C$ of dimensions $m \times k$ and $k \times n$, respectively, such that $A = B \odot C$, where $\odot$ denotes the matrix product over boolean arithmetic. It has been shown that computing the Boolean rank of a matrix is equivalent to computing the biclique cover number of a bipartite graph (see [8]). Similarly, the *binary rank* of a matrix $A \in \{0,1\}^{m \times n}$ is defined as the minimum $k$ for which there are $B \in \{0,1\}^{m \times k}$ and $C \in \{0,1\}^{k \times n}$ such that $A = B \cdot C$ using the standard arithmetic over the reals. It can be shown that the binary rank of the adjacency matrix of a bipartite graph is equal to its biclique partition number.

Low-rank decompositions are of particular interest in Data Analytics. However, it is even NP-hard to distinguish bipartite graphs or binary matrices that allow $k \in O(n^{\varepsilon})$ from ones that require $k \in \Omega(n^{1-\varepsilon})$ for all $\varepsilon > 0$ for both BicliqueCover as well as BicliquePartition [1]. In the same paper, only approximation factors of $\frac{n}{\sqrt{\log n}}$ were obtained with polynomial time algorithms.

From the parameterized complexity point of view, the picture is much brighter: BicliqueCover and BicliquePartition are in FPT when parameterized with $k$ [6], which even holds when the input graph is not bipartite. The authors obtain this result by providing rules for obtaining kernels with at most $3^k$ vertices for general graphs and with at most $2^{k+1}$ vertices for bipartite graphs. We will use this kernel to obtain some of our results.

## 1.1 Our contribution

We present an algorithm that decides whether a given bipartite graph has a BicliquePartition of size at most $k$ in time $\mathcal{O}^*(2^{2k^2+k\log k+k})$. This drastically improves the previous best bound [6], which is $\mathcal{O}^*(2^{2^{2k}\log k+3k})$ [12].[1] In contrast to this result, we prove that BicliqueCover seems to be much harder, i.e., there is no algorithm running in $\mathcal{O}^*(2^{2^{o(k)}})$ unless the Exponential Time Hypothesis (ETH) is false. This almost closes the gap between lower and best known upper bound, which is $\mathcal{O}^*(2^{2^k\log k+2k+\log k}/k!)$ for the latter [12]. Moreover, we prove an exponential lower bound for kernels for BicliqueCover that holds unless $P = NP$. We conclude by showing how the kernel for BicliqueCover and BicliquePartition improves the best known approximation factors for these two problems to $\mathcal{O}(n/\log n)$.

## 1.2 Preliminaries

For a bipartite graph $G$, we denote the two vertex bipartitions by $U(G)$ and $V(G)$. The edges are denoted by $E(G)$. For a subgraph $H$ of $G$, $U(H)$ denotes the set of vertices of $H$ that are in $U(G)$ and $V(H)$ denotes the set of vertices of $H$ that are in $V(G)$. All edges in this paper are undirected edges, and we may use $uv$ or $vu$ to denote an edge between vertices $u$ and $v$. For a graph $G$ and vertex $v$, we use $N_G(v)$ to denote the set of all the vertices that are adjacent to $v$ in $G$. We may choose to omit the subscript $G$ when the graph is clear from

[1] Note that the bound reported in [6] is inaccurate as also observed in [12].

the context. A **domino graph** is the graph $G$ with $U(G) = \{u_1, u_2, u_3\}$, $V(G) = \{v_1, v_2, v_3\}$ and $E(G) = \{u_1v_1, u_1v_2, u_2v_1, u_2v_2, u_2v_3, u_3v_2, u_3v_3\}$ (See $S_i$ in Figure 1). Two vertices $w_1$ and $w_2$ in graph $G$ are said to be **twins** of each other iff $N(w_1) = N(w_2)$. For a matrix $A$, we say that the $i^{\text{th}}$ row is $a_i^T$, the $j^{\text{th}}$ column is $A_j$, and the entry corresponding to the $i^{\text{th}}$ row and $j^{\text{th}}$ column is $a_{ij}$. For a prime number $p$, we use $GF(p)$ to denote the Galois Field over $\{0, 1, \ldots, p-1\}$ with modulo $p$ multiplication and addition.

## 2 FPT algorithm for BicliquePartition

This section is dedicated to the proof of the following theorem.

▶ **Theorem 1.** BICLIQUEPARTITION *has an algorithm which runs in* $\mathcal{O}^*(2^{2k^2+k\log k+k})$*-time.*

As mentioned above, the biclique partition number of a bipartite graph is equal to the binary rank of its adjacency matrix. Moreover, the binary decomposition $B \cdot C$ of a binary matrix $A$ with $A \in \{0,1\}^{m\times n}$, $B \in \{0,1\}^{m\times k}$, and $C \in \{0,1\}^{k\times n}$ also gives the set of bicliques in the corresponding biclique partition of the graph represented by $A$. Therefore, we consider the following problem in the remainder of this section: Given a binary matrix $A$, does $A$ have binary rank at most $k$? We develop an $\mathcal{O}^*(2^{2k^2+k\log k+k})$-time algorithm for this problem. Moreover, our algorithm also returns the binary decomposition $BC$ of $A$ if $A$ is a YES instance.

Let A be the given $m \times n$ binary matrix. If A has binary rank $k$, then there exist an $m \times k$ binary matrix $B$ and a $k \times n$ binary matrix $C$ such that $BC = A$. Let $p$ be the smallest prime that is greater than $k$. It is a well-known fact that $p$ is at most $2k$ and can be found in time polynomial in $k$. We will work with arithmetic over the prime field $GF(p)$. We know that $BC = A$ holds even over $GF(p)$. We can assume that $m, n \le 2^k$ because of the kernel given in [6]. Let $s$ be the rank of $B$ over $GF(p)$. Clearly, $s$ is at most $k$. Thus, we can guess the value of $s$. Observe that permuting rows do not change the binary rank of $A$. Moreover, the binary decomposition $BC$ still holds, provided we permute the rows of $B$ with the same permutation used for rows of $A$. We guess a permutation of the rows of $B$ such that the first $s$ rows of $B$ are linearly independent over $GF(p)$. Then, we apply the same permutation to the rows of $A$. Note that it suffices to try $\binom{m}{s}$ possibilities to find one such permutation. Since $m \le 2^k$ due to the kernel in [6] and $s \le k$, we get that $\binom{m}{s} \le 2^{k^2}$, and hence, this step is fine with respect to the running time that we want to achieve. Now, we guess the entries of the first $s$ rows of $B$. Since each row only has $2^k$ possibilities, there are only at most $(2^k)^s \le 2^{k^2}$ possibilities for this guess.

▶ **Lemma 2.** *For each* $i \in [m]$*, there exists a vector* $\lambda(i) \in \{0, 1, \ldots, p-1\}^s$ *such that* $\sum_{t=1}^{s} \lambda(i)_t a_t \equiv a_i \pmod p$ *and* $\sum_{t=1}^{s} \lambda(i)_t b_t \pmod p$ *is a* 0-1 *vector.*

**Proof.** For an arbitrary $i$, we exhibit a $\lambda(i)$, which satisfies the conditions stated in the statement of the lemma. Since the first $s$ rows of $B$ span all its rows over $GF(p)$, there exists $\lambda(i) \in \{0, 1, \ldots, p-1\}^s$ such that $\sum_{t=1}^{s} \lambda(i)_t b_t \equiv b_i \pmod p$, which is a 0-1 vector. Furthermore, $\sum_{t=1}^{s} \lambda(i)_t a_t^T \equiv \sum_{t=1}^{s} \lambda(i)_t b_t^T C \equiv b_i^T C \equiv a_i^T \pmod p$. ◀

For each $i \in [m]$, we find a vector $\lambda(i)$ as given by Lemma 2 and let $b'_i$ be the corresponding 0-1 vector given by $\sum_{t=1}^{s} \lambda(i)_t b_t$. For each $i \in [m]$, this can be done by trying all possibilities of $\lambda(i)_1, \ldots, \lambda(i)_s$ and takes only $\mathcal{O}(p^s) = \mathcal{O}(2^{k\log k+k})$ time. Hence, finding all the $\lambda(i)$ for all $i \in [m]$ takes only $\mathcal{O}(2^{k\log k+k}m)$ time. We would like to highlight that the choice of $\lambda(i)$ is done independently for each $i \in [m]$. Note that since we do not yet know the value of

$b_i$ for $i > s$, the $\lambda(i)$ that we fix is not guaranteed to be the one exhibited in the proof of Lemma 2, i.e., $b'_i$ could differ from $b_i$, but they are the same for $i \leq s$.

Let $B'$ be the matrix whose rows are $b'^T_1, \ldots, b'^T_m$. Let $\tilde{B}$ be the matrix $B$ restricted to its first $s$ rows, and let $\tilde{A}$ be the matrix $A$ restricted to its first $s$ rows. For each $j \in [n]$, we find a vector $C'_j \in \{0,1\}^k$ such that $\tilde{B}C'_j = \tilde{A}_j$. For each $j \in [n]$, this can be done by iterating over all possible binary vectors of length $k$, which only takes $\mathcal{O}(2^k)$ time. Note that such a $C'_j$ should exist for all $j \in [n]$ if $A$ has binary rank $k$, provided that our guess about the $s$ linearly independent rows in $B$ is correct. Again, we would like to highlight that the choice of $C'_j$ is done independently for each $j \in [n]$. Let $C'$ denote the matrix whose columns are $C'_1, C'_2, \ldots, C'_n$.

The following lemma is the core idea that our algorithm uses. The lemma essentially says that although we choose the $b'_i$'s and $C'_j$'s independently, they actually combine to give a binary decomposition of $A$ over $GF(p)$.

▶ **Lemma 3.** *For each $i \in [m], j \in [n]$, $b'^T_i C'_j \equiv a_{ij} \pmod p$.*

**Proof.** $b'^T_i C'_j \equiv \left(\sum_{t=1}^{s} \lambda(i)_t b_t\right)^T C'_j \pmod p \equiv \sum_{t=1}^{s} \lambda(i)_t a_{tj} \pmod p \equiv a_{ij} \pmod p$ ◀

It only remains to eliminate the $GF(p)$ arithmetic, which is done by the following Lemma.

▶ **Lemma 4.** *For each $i \in [m], j \in [n]$, $b'^T_i C'_j = a_{ij}$.*

**Proof.** From Lemma 3, we have that $b'^T_i C'_j \equiv a_j \pmod p$. Since $b'_i$ and $C'_j$ are 0-1 vectors of length $k < p$, we have $b^T_i C'_j = (b'^T_i C'_j \mod p) = a_{ij}$. ◀

From Lemma 4, we have that $B'C' = A$. Since $B'$ is an $m \times k$ binary matrix and $C'$ is a $k \times n$ binary matrix, we have a binary factorization of $A$ with binary rank $k$. A pseudocode for the algorithm is given in Algorithm 1, and in Lemma 5, we prove that its running time is $\mathcal{O}^*(2^{2k^2+k\log k+k})$.

▶ **Lemma 5.** *The running time of Algorithm 1 is $\mathcal{O}^*(2^{2k^2+k\log k+k})$.*

**Proof.** The number of iterations of the outer loop is at most $k \cdot \binom{m}{s} \cdot 2^{sk} = \mathcal{O}^*(2^{2k^2})$, which follows from $m \leq 2^k$ and $s \leq k$. The two inner loops only have $n$ iterations at most. Step 5 takes $O^*(p^s) = O^*(2^{k+k\log k})$ time, and step 9 takes $O^*(2^k)$ time. All other steps take time polynomial in $m$ and $n$. Hence, the total time taken by the algorithm is $\mathcal{O}^*(2^{2k^2} \cdot (2^{k+k\log k} + 2^k)) = \mathcal{O}^*(2^{2k^2+k\log k+k})$. ◀

## 3 FPT and kernel lower bounds for BicliqueCover

In this section, we prove the following theorem, which has consequences for the complexity of BicliqueCover as stated in the corollaries below.

▶ **Theorem 6.** *There exists a polynomial time reduction that, given a 3-SAT instance $\psi$ on $n$ variables and $m$ clauses, produces a bipartite graph $G$ with $|U(G)| + |V(G)| = \mathcal{O}(n+m)$ such that there exists a positive integer $k = \mathcal{O}(\log n)$ for which $G$ has a biclique cover of size at most $k$ if and only if $\psi$ is satisfiable.*

▶ **Corollary 7.** BicliqueCover *cannot be solved in time $\mathcal{O}^*(2^{2^{o(k)}})$-time unless the Exponential Time Hypothesis is false.*

**Proof.** Follows directly from Theorem 6. ◀

**Algorithm 1:** FPT algorithm for BicliquePartition

**Input :** An $m \times n$ binary matrix $A$ and positive integer $k$ such that $m, n \leq 2^k$.
**Output :** Either report that binary rank of $A$ is greater than $k$ or output $m \times k$ binary matrix $B'$ and $k \times n$ binary matrix $C'$ such that $B'C' = A$

Find $p$, the smallest prime greater than $k$.
**foreach** $s \in [k]$,$\{i_1, i_2, \cdots, i_s\} \subseteq [m]$, *and* $\tilde{B} \in \{0,1\}^{s \times k}$ **do** `// loop 1`
    Permute the rows of $A$ such that rows $i_1, i_2, \cdots, i_s$ become the first $s$ rows of $A$. Let this permutation be $\sigma$;
    **for** $i \leftarrow 1$ **to** $m$ **do**
        Find a $\lambda(i) \in \{0, 1, \cdots, p-1\}^s$ such that $\sum_{t=1}^{s} \lambda(i)_t a_t \equiv a_i \pmod p$ and $\sum_{t=1}^{s} \lambda(i)_t b_t \pmod p$ is a 0-1 vector; **if** there is no such $\lambda(i)$, **then go to** the next iteration of loop 1;
        $b'_i \leftarrow \sum_{t=1}^{s} \lambda_t(i) \tilde{b}_t$;
    **end**
    **for** $j \leftarrow 1$ **to** $n$ **do**
        Find $C'_j \in \{0,1\}^k$ such that $\tilde{B}C'_j = \tilde{A}_j$ where $\tilde{A}$ is the matrix $A$ restricted to first $s$ rows; **if** there is no such $C'_j$, **then go to** the next iteration of loop 1;
    **end**
    **let** $B'$ be the matrix with $b'^T_1, b'^T_2, \cdots, b'^T_m$ as the rows and $C'$ be the matrix with $C'_1, C'_2, \cdots, C'_n$ as the columns;
    Apply the inverse permutation of $\sigma$ to the rows of $B'$;
    **output** $B'$ and $C'$ and **terminate**.
**end**
**report** that binary rank of $A$ is greater than $k$.

▶ **Corollary 8.** *There exists a constant $\delta > 0$ such that, unless $P = NP$, there is no polynomial time algorithm that produces a kernel for* BicliqueCover *of size less than $2^{\delta k}$.*

**Proof.** We give a proof sketch and refer to [2] for the details where the authors prove a similar statement for EdgeCliqueCover. By Theorem 6, we have an algorithm $A$ that takes an instance of 3-SAT and gives an equivalent instance of BicliqueCover with parameter $k = \mathcal{O}(\log n)$. Suppose there is a kernelization algorithm $B$ that produces a kernel with less than $2^{\delta k}$ size for some $\delta$ to be fixed later. Since BicliqueCover is NP-complete, there exists an algorithm $C$ that takes an instance of BicliqueCover and gives an equivalent instance of 3-SAT in polynomial time. By composing the algorithms $A$, $B$, and $C$ and fixing the parameter $\delta$ appropriately, we get an algorithm $D$ that, given a 3-SAT instance as input, produces an equivalent smaller 3-SAT instance as output. We can apply $D$ repeatedly to solve 3-SAT. Hence, algorithm $B$ cannot exist. ◀

The proof of Theorem 6 gives a reduction from 3-SAT to BicliqueCover, which is a modification of the one given in [2] from 3-SAT to EdgeCliqueCover.[2] The main difference is that we introduce an additional gadget consisting of $\log_2 n$ domino graph gadgets in order to make the reduction work for BicliqueCover, where $n$ is the number of variables in the input 3-SAT formula. This gadget replaces the independent set of size $\log_2 n$ used in [2], i.e.,

[2] A parameter preserving reduction from EdgeCliqueCover to BicliqueCover would have been better than having to redo the whole reduction from scratch. We could not find any such reduction.

we replace each vertex there by a domino graph here. We also modify some of the adjacencies in the construction such that the graph becomes bipartite. Moreover, we have simplified the reduction of [2] by using a simple trick. The trick is to make one of the domino graphs special by adding edges between this domino graph and clause gadgets so that the biclique covering this domino graph corresponds to a satisfying assignment. For EdgeCliqueCover, this corresponds to making one of the vertices in the independent set special by adding edges from it to the clause gadgets. We give the complete reduction for BicliqueCover here for being self-contained.

In [2], the authors use cocktail party graphs as the main gadget in their reduction. We use the bipartite analogue called *crown graphs*. A crown graph is basically a complete bipartite graph minus a perfect matching. It is formally defined as follows.

▶ **Definition 9** (Crown Graph, $H_r$)**.** A crown graph on $2r$ vertices denoted by $H_r$ is a bipartite graph with bipartitions $U(G) = \{u_1, u_2, \ldots, u_r\}$ and $V(G) = \{v_1, v_2, \ldots, v_r\}$ such that there is an edge from $u_i$ to $v_j$ iff $i \neq j$. In other words, the edges missing between $U(H_r)$ and $V(H_r)$ form a perfect matching given by $\{u_i v_i : i \in [r]\}$. (See $H = H_n$ in Figure 1.)

If we pick exactly one vertex from each of the edges of the missing perfect matching of the crown graph, then we get a maximal biclique provided that we pick at least one vertex from each of the bipartitions. The complement of this vertex set also forms a maximal biclique. These pair of bicliques are called duplex bicliques, formally defined as follows.

▶ **Definition 10** (Duplex Biclique[3])**.** A duplex biclique of a crown graph $H_r$ is defined as a pair of bicliques $\{B_1, B_2\}$ such that $U(B_1) \cap U(B_2) = \emptyset, V(B_1) \cap V(B_2) = \emptyset$, and $U(B_1) \cup U(B_2) = U(H_r)$, and $V(B_1) \cup V(B_2) = V(H_r)$.

We go on to define a duplex biclique cover as follows.

▶ **Definition 11** (Duplex Biclique Cover)**.** A duplex biclique cover of a crown graph is defined as a set of duplex bicliques that together cover all the edges of the graph. When we say size of a duplex biclique cover, we mean the number of bicliques in the cover, which is twice the number of duplex bicliques.

We prove the following two lemmas about crown graphs.

▶ **Lemma 12.** *$H_r$ has a duplex biclique cover of size $2\lceil \log r \rceil$ that can be found in time polynomial in $n$.*

**Proof.** We exhibit such a biclique cover. Let $\ell = \lceil \log r \rceil$. For any $x \in \{0, \ldots, 2^\ell - 1\}$ and $j \in [\ell]$, let $\langle x \rangle_j$ denote the $j$-th bit of the $\ell$-bit binary representation of $x$. For each $j \in [\ell]$, we define the $j$-th duplex biclique $\{T_j^1, T_j^2\}$ as follows: $T_j^1$ is the subgraph induced by $\{u_i : \langle i-1 \rangle_j = 1\} \cup \{v_i : \langle i-1 \rangle_j = 0\}$, and $T_j^2$ is the subgraph induced by $\{u_i : \langle i-1 \rangle_j = 0\} \cup \{v_i : \langle i-1 \rangle_j = 1\}$, where $u_i$ and $v_i$ are defined as in Definition 9. It is easy to see that $\{T_j^1, T_j^2\}$ is indeed a duplex biclique. It is also easy to see that any pair of vertices $u_i, v_j$ such that $i \neq j$ should be present in at least one of the $\ell$ duplex bicliques. ◀

▶ **Lemma 13.** *Given a duplex biclique $\{B_1, B_2\}$ of $H_r$ such that $|U(B_1)| = |V(B_1)|$, we can in polynomial time find a duplex biclique cover of $H_r$ with size $2\lceil \log_2 r \rceil$ such that $\{B_1, B_2\}$ is one of the duplex bicliques forming the biclique cover.*

[3] Duplex Bicliques correspond to Twin Cliques in [2]. We use this name to avoid confusion with twin vertices.

**Proof.** Using the definition of duplex bicliques and $|U(B_1)| = |V(B_1)|$, it follows that $[r]$ can be partitioned into 2 sets

$$J_1 = \{j \in [r] : u_j \in U(B_1) \wedge v_j \in V(B_2)\} \text{ and } J_2 = \{j \in [r] : u_j \in U(B_2) \wedge v_j \in V(B_1)\}$$

which are each of size $\frac{r}{2}$. We can reorder the indices of the vertices such that $J_1 = \left\{1, 2, \cdots \frac{r}{2}\right\}$ and $J_2 = \left\{\frac{r}{2}+1, \frac{r}{2}+2, \cdots, r\right\}$. Let $\ell = \log_2 r$. We define the duplex biclique $\{T_i^1, T_i^2\}$ for all $i \in [\ell]$ the same way as in the proof of Lemma 12. It is clear that the duplex biclique $\{B_1, B_2\}$ is the same as the duplex biclique $\{T_1^1, T_1^2\}$. Thus, the set of bicliques $\left\{T_1^1, T_2^1, \ldots, T_\ell^1\right\} \cup \left\{T_1^2, T_2^2, \ldots, T_\ell^2\right\}$ gives the required duplex biclique cover. ◀

Now, we state the following fact about twin vertices.

▶ **Fact 14.** *If $a_1$ and $a_2$ are twins and $E(G \setminus a_2)$ can be covered with $k$ bicliques, then $E(G)$ can be covered with $k$ bicliques by adding $a_2$ to all the bicliques containing $a_1$.*

When we say we apply **twin-reduction** to a pair of twin vertices, we mean the operation of deleting one of the twin vertices from the graph. When we say we apply twin-reduction to a graph, we mean to repeatedly apply twin-reduction until there are no more twins in the graph.

Let $\psi$ be the input 3-SAT formula with $n$ variables and $m$ clauses. Let $x_1, \ldots, x_n$ be the variables of $\psi$ and $C_1, \ldots, C_m$ be the clauses. Let $C_i^1, C_i^2$, and $C_i^3$ denote the 3 literals of clause $C_i$. For $1 \le a \le 3$, we say that $C_i^a = (x_j, 1)$ if the $a^{\text{th}}$ literal in clause $C_i$ is the variable $x_j$ appearing in positive form, and we say $C_i^a = (x_j, 0)$ if the $a^{\text{th}}$ literal in $C_i$ is the variable $x_j$ appearing in negated form.

**Assumptions about the input 3-SAT formula:** We assume that the number of variables is a power of 2. We also assume that if the instance is satisfiable, then there is a satisfying assignment $A$ such that half of the variables are assigned true in $A$ and the other half false. These assumptions can be handled easily by introducing some extra variables as shown in [2]. Note that this increases the number of variables by at most 4 times.

Let $\ell$ be such that $2^\ell = n$. We have that $\ell \in \mathbb{Z}$ since $n$ was assumed to be a power of 2. Before giving the reduction, we give the following useful definition.

▶ **Definition 15** (Bisimplicial Edge)**.** An edge $uv$ is said to be bisimplicial with respect to a biclique $B$ iff $N(u) \cup N(v) = U(B) \cup V(B)$.

Now, we give the reduction from 3-SAT to BicliqueCover.

**Construction:** Given $\psi$, we construct a bipartite graph $G$. See Figure 1 for an illustration of the construction. A vertex with superscript $u$ indicates that it belongs to $U(G)$, and a superscript $v$ indicates that it belongs to $V(G)$. The edges of $G$ are divided into two sets, a set of important edges $E^{imp}$ and a set of free edges $E^{free}$. The number of bicliques required to cover $E^{imp}$ will be different depending on whether $\psi$ is satisfiable or not, whereas the number of bicliques required to cover $E^{free}$ will depend only on the number of variables and clauses of $\psi$ but not on whether $\psi$ is satisfiable or not. There are 5 main gadgets in our construction of $G$ as given below.

1. A graph $H$ isomorphic to the crown graph $H_n$: Let the vertices of $U(H)$ be $h_1^u, h_2^u, \ldots, h_n^u$ and that of $V(H)$ be $h_1^v, h_2^v, \ldots, h_n^v$. The edges of $H$ are in $E^{imp}$. The vertices $h_i^u$ and $h_i^v$ correspond to the $i$-th variable of $\psi$. $h_i^u$ corresponds to the variable in positive form and the vertex $h_i^v$ corresponds to the variable in negative form.

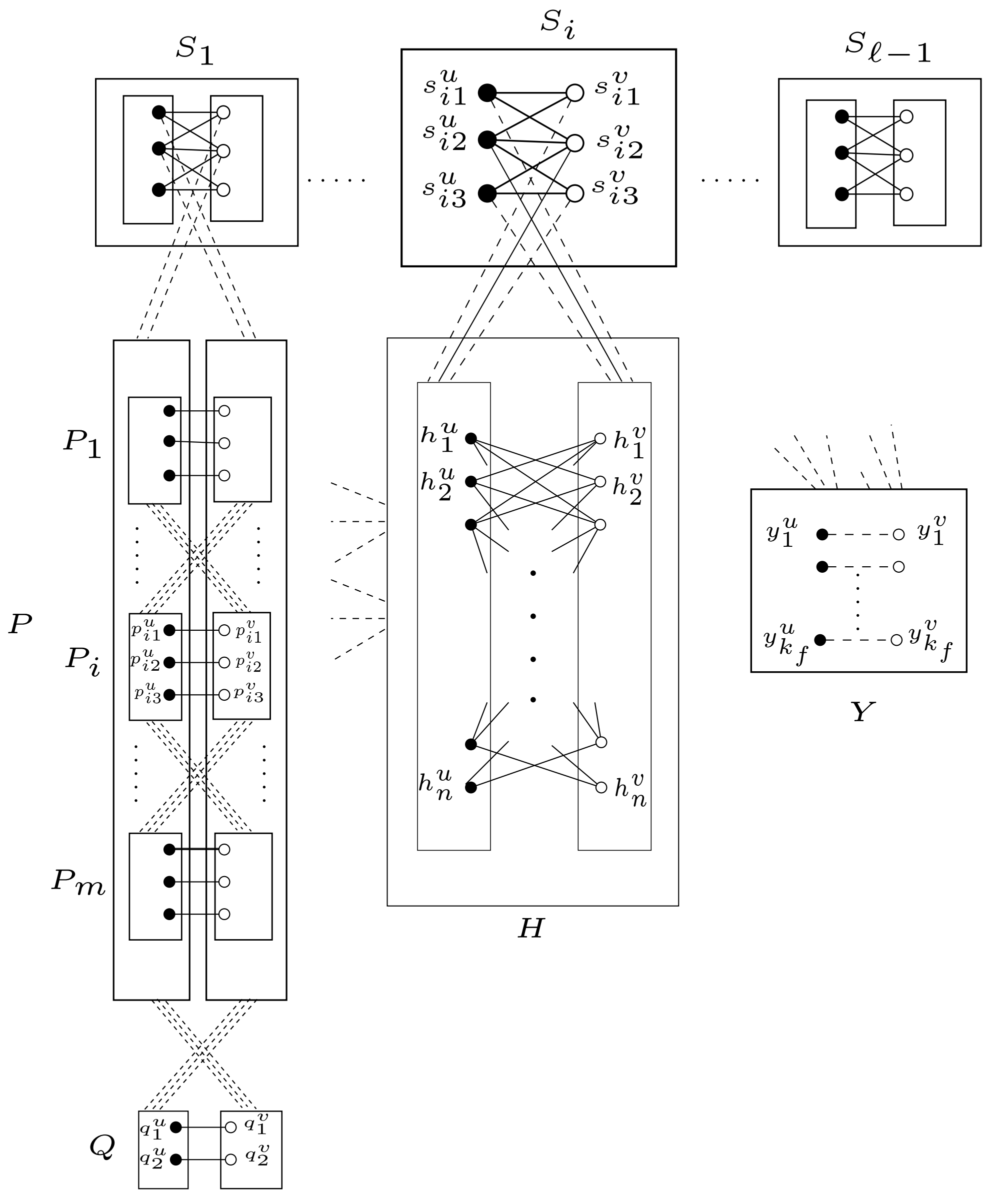


**Figure 1** Illustration of the construction of G. The black nodes denote the vertices in $U(G)$, and the white nodes denote the vertices in $V(G)$. The solid edges represent edges in $E^{imp}$, and the dashed edges denote edges in $E^{free}$. The edges between $P$ and $H$ and those between $Y$ and $G \setminus Y$ are not shown. The edges shown between $S_i$ and $H$ are present for all $i \in [\ell - 1]$, whereas the edges shown between $S_1$ and $P$ are only present for $S_1$ and not for any $S_i$ for $i \geq 2$.

2. A set $P$ of clause gadgets $P_1, \ldots, P_m$: Each $P_i$ is an induced matching of size 3. Let $U(P_i) = \{p^u_{i1}, p^u_{i2}, p^u_{i3}\}$ and $V(P_i) = \{p^v_{i1}, p^v_{i2}, p^v_{i3}\}$ and let the 3 edges of $P_i$ be $p^u_{i1}p^v_{i1}, p^u_{i2}p^v_{i2}$, and $p^u_{i3}p^v_{i3}$. These edges are in $E^{imp}$. For all $i \in [m]$, $P_i$ corresponds to the clause $C_i$ in $\psi$, and the 3 edges in $P_i$ correspond to the 3 literals in the clause, i.e., edge $p^u_{ia}p^v_{ia}$ corresponds to literal $C^a_i$ for $a \in \{1, 2, 3\}$.
3. A set $S$ of $\ell$ domino graphs $S_1, S_2, \ldots, S_\ell$ that are disconnected with each other: Let $U(S_i) = \{s^u_{i1}, s^u_{i2}, s^u_{i3}\}$ and $V(S_i) = \{s^v_{i1}, s^v_{i2}, s^v_{i3}\}$. The edges within each $S_i$ are in $E^{imp}$.

4. An induced matching $Y$ of size $k_f$ where $k_f = \mathcal{O}(\log n)$ will be fixed later in Lemma 16: $Y$ consists of edges $y_1^u y_1^v, \ldots, y_{k_f}^u y_{k_f}^v$, which are in $E^{free}$. For all $i \in [k_f]$, the edge $y_i^u y_i^v$ will be made bisimplicial with respect to the biclique $\tilde{B}_i^f$, which will be defined later. This is done to ensure that we need $k_f$ bicliques to cover the free edges.

We also have the following edges between gadgets.

- Between $H$ and $S$: For all $i \in [\ell]$ and $j \in [n]$, we add the following edges: $s_{i2}^u h_j^v$ and $s_{i2}^v h_j^u$ to $E^{imp}$; and, $s_{i1}^u h_j^v$, $s_{i3}^u h_j^v$, $s_{i1}^v h_j^u$, and $s_{i3}^v h_j^u$ to $E^{free}$.
- Between $P_i$ and $P_j$: For all $i \neq j \in [m]$, add edges between all pairs of vertices $u, v$ such that $u \in U(P_i)$ and $v \in V(P_j)$. These edges are in $E^{free}$.
- Between $P$ and $Q$: For all $i \in [m]$, add edges between all pairs of vertices $u, v$ such that $u \in U(Q)$ and $v \in V(P_i)$. Similarly, for all $i \in [m]$, add edges between all pairs of vertices $u, v$ such that $u \in U(P_i)$ and $v \in V(Q)$. These edges are in $E^{free}$.
- Between $H$ and $P$: For all $i \in [m]$ and $a \in [3]$, add edges between $p_{ia}^u$ and $h_j^v$ unless $C_i^a = (x_j, 1)$ and between $p_{ia}^v$ and $h_j^u$ unless $C_i^a = (x_j, 0)$. These edges are in $E^{free}$.
- Between $S$ and $P$: The only vertices in $S$ that will have edges to any $P_i$ are the 4 vertices $s_{11}^u, s_{12}^u, s_{11}^v$, and $s_{12}^v$. From $s_{11}^u$ and $s_{12}^u$, add edges to all vertices in $V(P_i)$ for all $i \in [m]$. Similarly, from $s_{11}^v$ and $s_{12}^v$, add edges to all vertices in $U(P_i)$ for all $i \in [m]$. These edges are in $E^{free}$.
- Between $Y$ and $G \setminus Y$: These edges are added in such a way that edge $y_i^u y_i^v$ is bisimplicial w.r.t. a biclique that will be defined later. We will give the exact description of these edges after we define the bicliques $B_i^f$ for $i \in [k_f]$. These edges belong to $E^{free}$.

**Summary of $\mathbf{E^{imp}}$:** All the edges within $H, S$, and $Q$; all edges within each $P_i$; edges $s_{i2}^u h_j^v$ and $s_{i2}^v h_j^u$ for all $i \in [\ell - 1], j \in [n]$.

**Summary of $\mathbf{E^{free}}$:** All the edges between $P_i$ and $P_j$ for $i \neq j$; all edges within $Y$; all edges between $Y$ and $G \setminus Y$; edges $s_{i1}^u h_j^v$, $s_{i3}^u h_j^v$, $s_{i1}^v h_j^u$, and $s_{i3}^v h_j^u$ for all $i \in [\ell - 1], j \in [n]$; all edges between $P$ and $Q$, between $H$ and $P$, and between $S_1$ and $P$.

First, we show how to take care of the edges in $E^{free}$ without interfering with the budget of $E^{imp}$. Let $E^y$ be the set of all edges of $G$ with at least one end point in $U(Y) \cup V(Y)$.

▶ **Lemma 16.** *The edges in $E^{free} \setminus E^y$ can be covered using $k_f = 4\log_2 n + 2\lceil \log_2 m \rceil + 6$ bicliques of $G$ such that none of these bicliques contains an edge from $E^{imp}$, and these bicliques can be found in time polynomial in $n + m$.*

**Proof.** According to our construction, there are the following types of edges in $E^{free} \setminus E^y$. For each of these types, we show how to cover it in polynomial time using bicliques that do not contain any edges from $E^{imp}$ such that the total number of bicliques used is at most $k_f$.

- Edges between $H$ and $S$: These edges can be covered with 2 bicliques, $B_1^{HS}$ and $B_2^{HS}$ defined as follows. $U(B_1^{HS}) = U(H)$, $V(B_1^{HS}) = \{s_{i1}^v : 1 \leq i \leq \ell\} \cup \{s_{i3}^v : 1 \leq i \leq \ell\}$, $U(B_2^{HS}) = \{s_{i1}^u : 1 \leq i \leq \ell\} \cup \{s_{i3}^u : 1 \leq i \leq \ell\}$, and $V(B_2^{HS}) = V(H)$. From the construction of $G$, it is easy to see that both $B_1^{HS}$ and $B_2^{HS}$ are indeed bicliques, and none of them contains an edge in $E^{imp}$.
- $E(P) \cap E^{free}$: These edges can be covered with $2\lceil \log_2 m \rceil$ bicliques. Consider the subgraph of $G$ given by the edges $E(P) \setminus E^{imp}$. Let this graph be $G_1$. The vertices $p_{i1}^u, p_{i2}^u$, and $p_{i3}^u$ are twins of each other in $G_1$ for all $i \in [m]$. Similarly, the vertices $p_{i1}^v, p_{i2}^v$, and $p_{i3}^v$ are twins of each other in $G_1$ for all $i \in [m]$. Let $G_2$ be the graph obtained by applying twin-reduction to $G_1$. It is clear that $G_2$ is isomorphic to the crown graph $H_m$. Hence,

$E(G_2)$ can be covered by $\lceil 2\log_2 m \rceil$ bicliques in polynomial time by Lemma 12. Then, by using Fact 14, we can find $\lceil 2\log_2 m \rceil$ bicliques that cover $E(G_1) = E(P) \setminus E^{imp}$.

- Edges between $P$ and $Q$: We can cover these edges with 2 bicliques $B_1^{PQ}$ and $B_2^{PQ}$, defined as $U(B_1^{PQ}) = U(P)$, $V(B_1^{PQ}) = V(Q)$; and, $U(B_2^{PQ}) = U(Q)$, $V(B_2^{PQ}) = V(P)$.
- Edges between $H$ and $P$: We cover these edges with $4\log_2 n$ bicliques. We will show how to cover the edges between $U(H)$ and $V(P)$ with $2\log_2 n$ bicliques. Symmetrically, the edges between $V(H)$ and $U(P)$ can be covered with another $2\log_2 n$ bicliques. Let $\langle j \rangle_i$ denote the $i^{\text{th}}$ bit in the binary representation of $j$. We will now give the description of a set of $2\log_2 n$ bicliques $\mathcal{B}^{HP} = \{B_1^{HP}, \ldots, B_{\log_2 n}^{HP}\} \cup \{\tilde{B}_1^{HP}, \ldots, \tilde{B}_{\log_2 n}^{HP}\}$ covering the edges between $U(H)$ and $V(P)$. We define $U(B_i^{HP}) = \{h_j^u : \langle j \rangle_i = 1\}$, $U(\tilde{B}_i^{HP}) = \{h_j^u : \langle j \rangle_i = 0\}$, $V(B_i^{HP}) = \bigcap_{u \in U(B_i^{HP})} N(u) \cap V(P)$, $V(\tilde{B}_i^{HP}) = \bigcap_{u \in U(\tilde{B}_i^{HP})} N(u) \cap V(P)$. It is clear that these are indeed bicliques from the definitions of $V(B_i^{HP})$ and $V(\tilde{B}_i^{HP})$. Since there are no edges of $E^{imp}$ between $U(H)$ and $V(P)$, we do not cover any edges in $E^{imp}$. We now show that we have covered every edge between $U(H)$ and $V(P)$. Suppose for the sake of contradiction that the edge $h_j^u p_{ia}^v$ was not covered. Let $p_{ia}^v$ correspond to variable $x_t$. Recall that the only vertex in $U(H)$ that can possibly not have edge to $p_{ia}^v$ is $h_t^u$. Edge $h_j^u p_{ia}^v$ not being covered by any biclique in $\mathcal{B}$ can happen only if every biclique in $\mathcal{B}^{HP}$ that contains $h_j^u$ also contains $h_t^u$ and if there is no edge between $h_t^u$ and $p_{ia}^v$. But if every biclique in $\mathcal{B}$ containing $h_j^u$ also contains $h_t^u$, then $j = t$. This means that there is no edge between $h_j^u$ and $p_{ia}^v$, which is a contradiction.
- Edges between $S$ and $P$: These edges can be covered with 2 bicliques $B_1^{PS}$ and $B_2^{PS}$, which is defined as follows. $U(B_1^{PS}) = U(P)$, $V(B_1^{PS}) = \{s_{11}^v, s_{12}^v\}$, $U(B_2^{PS}) = \{s_{11}^u, s_{12}^u\}$ and $V(B_2^{PS}) = V(P)$. ◀

We fix $k_f$ as given by Lemma 16. By Lemma 16, we know that there are $k_f$ bicliques that together cover all edges in $E^{free} \setminus E^y$ and do not cover any edges in $E^{imp}$. We will call these bicliques $B_1^f, \ldots, B_{k_f}^f$.

Now, we give the description of the edges from $Y$ to $G \setminus Y$. Recall that these edges are contained in $E^y \subset E^{free}$. For each $i \in [k_f]$, we add edges from $y_i^u$ to all the vertices in $V(B_i^f)$ and from $y_i^v$ to all vertices in $U(B_i^f)$. Observe that now the edge $y_i^u y_i^v$ is bisimplicial with respect to $B_i^f$. This together with Lemma 16 gives the following Lemma about the edge set $E^{free}$.

▶ **Lemma 17.** *Let $k_f = 4\log_2 n + 2\lceil \log_2 m \rceil + 6$.*

1. *The edge set $E^{free}$ can be covered using $k_f$ bicliques of $G$ such that none of these bicliques contains an edge from $E^{imp}$, and these bicliques can be found in time polynomial in $n+m$.*
2. *Any set of bicliques covering $E^{free}$ has $k_f$ bicliques that do not contain any edges from $E^{imp}$.*

**Proof.** From Lemma 16, we know that $E^{free} \setminus E^y$ can be covered by $k_f$ bicliques that do not cover any edges from $E^{imp}$. Given these $k_f$ bicliques $B_1^f, \ldots, B_{k_f}^f$, we extend them to the bicliques $\tilde{B}_i^f, \ldots, \tilde{B}_{k_f}^f$ as follows to cover all the edges of $E^{free}$ : $U(\tilde{B}_i^f) = U(B_i^f) \cup \{y_i^u\}$, and $V(\tilde{B}_i^f) = V(B_i^f) \cup \{y_i^v\}$. It is clear that $\tilde{B}_1^f, \ldots, \tilde{B}_n^f$ are all indeed bicliques, they together cover all edges of $E^{free}$, and do not cover any edges of $E^{imp}$.

Since the edges within $Y$ form an induced matching of size $k_f$, no two of them can be present in the same biclique. Hence, we need at least $k_f$ bicliques to cover the edges in $E^{free}$. We now show that if a biclique contains an edge from $E^{imp}$, it cannot have an edge from $Y$, which will complete the proof of the lemma. Suppose the edge $y_i^u y_i^v$ and an important edge $z^u z^v \in E^{imp}$ are in the same biclique for the sake of contradiction. But since $N(y_i^u) = V(B_i^f) \cup \{y_i^v\}$, we have that $z^v \in V(B_i^f)$. Symmetrically, we can argue that $z^u \in U(B_i^f)$. Then, however, $B_i^f$ contains the important edge $z^u z^v$, which is a contradiction. ◀

We set our budget $k$ as $k_f + 2\ell + 2$, which means a budget of $2\ell + 2$ for the edges in $E^{imp}$ due to Lemma 17. Now, we argue the completeness of the reduction in the following Lemma.

▶ **Lemma 18.** *If $\psi$ is satisfiable, then the edges in $E^{imp}$ can be covered by $2\ell + 2$ bicliques of $G$. (These bicliques might contain some edges from $E^{free}$ as well.)*

**Proof.** We know that there exists a satisfying assignment $A$ of $\psi$ such that exactly half of the variables are assigned *true* in $A$. Each clause $C_i$ has at least one literal which satisfies the clause. For each clause $C_i$, we fix one such literal. This literal corresponds to one of the 3 edges of $P_i$. Let us denote this edge by $e_i$.

We use two bicliques, $B_1^g$ and $B_2^g$, to cover the 2 guard edges of $Q$ and 2 edges from each $P_i$. Each of $B_1^g$ and $B_2^g$ covers 1 edge from $Q$ and 1 edge from each $P_i$. It is clear that this can be done. Now, each $P_i$ has one edge still to be covered. We will assign $B_1^g$ and $B_2^g$ such that the edge left uncovered in $P_i$ is $e_i$, i.e., the literal corresponding to this edge evaluates to *true* in $A$.

Let $B_1$ be the biclique defined as follows: $U(B_1) = \{h_i^u : A(x_i) = true, i \in [n]\}$ and $V(B_1) = \{h_i^v : A(x_i) = false, i \in [n]\}$. Also, define $\bar{B}_1$ as the biclique defined by the vertex sets $U(\bar{B}_1) = U(H) \setminus U(B_1)$ and $V(\bar{B}_1) = V(H) \setminus V(B_1)$. $B_1$ and $\bar{B}_1$ are indeed bicliques of $H$ because no literal evaluates to both *true* and *false*, and thus, the missing edges corresponding to the missing perfect matching in $H$ are avoided. Likewise, they are duplex bicliques due to the manner in which $\bar{B}_1$ is defined. Moreover, $|U(B_1)| = |V(B_1)|$ since $A$ has half of the variables assigned *true* and the other half $false$. Therefore, by Lemma 13, there exist $\ell - 1$ other duplex bicliques $\{B_2, \bar{B}_2\}, \ldots, \{B_\ell, \bar{B}_\ell\}$ such that $B_1, \bar{B}_1, B_2, \bar{B}_2, \ldots, B_\ell$, and $\bar{B}_\ell$ together cover $E(H)$. Now we extend these bicliques with additional vertices so that these bicliques together with $B_1^g$ and $B_2^g$ cover $E^{imp}$, which is done as follows. For $2 \le j \le \ell$, we define biclique $B'_j$ as $U(B'_j) = U(B_j) \cup \{s_{j1}^u, s_{j2}^u\}$ and $V(B'_j) = V(B_j) \cup \{s_{j1}^v, s_{j2}^v\}$. For $1 \le j \le \ell$, we define $\bar{B}'_j$ as $U(\bar{B}'_j) = U(\bar{B}_j) \cup \{s_{j2}^u, s_{j3}^u\}$ and $V(\bar{B}'_j) = V(B_j) \cup \{s_{j2}^v, s_{j3}^v\}$. $B'_1$ is defined as $U(B'_1) = U(B_1) \cup \{s_{11}^u, s_{12}^u\} \cup \bigcup_{i \in [m]} U(e_i)$ and $V(B'_1) = V(B_1) \cup \{s_{11}^v, s_{12}^v\} \cup \bigcup_{i \in [m]} V(e_i)$. It is clear that each $B'_i$ and $\bar{B}'_i$ is indeed a biclique of $G$ and that the bicliques $B'_1, B'_2, \ldots, B'_\ell, \bar{B}'_1, \bar{B}'_2, \ldots, \bar{B}'_\ell, B_1^g$, and $B_2^g$ together cover $E^{imp}$. Hence, $E^{imp}$ can be covered by $2\ell + 2$ bicliques of $G$. ◀

Now, we argue the soundness of our reduction in the next Lemma.

▶ **Lemma 19.** *If $E^{imp}$ can be covered by using $2\ell + 2$ bicliques of $G$, then $\psi$ is satisfiable.*

**Proof.** The edge set $M = \{q_1^u q_1^v, q_2^u q_2^v\} \cup \{s_{j1}^u s_{j1}^v : j \in [\ell]\} \cup \{s_{j3}^u s_{j3}^v : j \in [\ell]\}$ forms an induced matching of size $2\ell + 2$ in $G$. Recall that all these edges are in $E^{imp}$. Since no two edges of an induced matching can be contained in the same biclique, each edge in $M$ has to be covered by a distinct biclique. Let the biclique that covers $q_1^u q_1^v$ be $B_1^g$ and the one that covers $q_2^u q_2^v$ be $B_2^g$. Let $B_j$ and $\bar{B}_j$ be the bicliques covering the edges $s_{j1}^u s_{j1}^v$ and $s_{j3}^u s_{j3}^v$, respectively. Since we have already used our budget of $2\ell + 2$, all the edges in $E^{imp}$ must be covered by at least one biclique in $\mathcal{B} = \{B_g^1, B_g^2\} \cup \{B_1, B_2, \cdots B_\ell\} \cup \{\bar{B}_1, \bar{B}_2, \cdots, \bar{B}_\ell\}$. The only possible bicliques in $\mathcal{B}$ that can contain $s_{j2}^u$ or $s_{j2}^v$ are $B_j$ and $\bar{B}_j$. That means, edges between $s_{j2}^u$ and $H$ and edges between $s_{j2}^v$ and $H$ have to be covered by $\{B_j, \bar{B}_j\}$. Moreover, they have to be partitioned by $\{B_j, \bar{B}_j\}$ for the following reason: if the edge $s_{j2}^u h_i^v$ appears in both bicliques $B_j$ and $\bar{B}_j$, then the edge $s_{j2}^v h_i^u$ cannot appear in any of the two bicliques as there is no edge between $h_i^u$ and $h_i^v$; and symmetrically, if the edge

$s_{j2}^v h_i^u$ appears in both bicliques $B_j$ and $\bar{B}_j$, then the edge $s_{j2}^u h_i^v$ cannot appear in any of the two bicliques. Combined with the fact that $N(\{s_{j2}^u, s_{j2}^v\}) = U(H) \cup V(H)$, we get that $\{B_j', \bar{B}_j'\}$ is a duplex biclique, where $B_j'$ and $\bar{B}_j'$ are the intersection of the bicliques $B_j$ and $\bar{B}_j$, respectively, with $H$.

$B_1^g$ and $B_2^g$ can each cover at most 1 edge from each $P_i$. Hence, there is at least 1 edge of each $P_i$ that must be covered by $\mathcal{B} \setminus \{B_g^1, B_g^2\}$. Let us fix one such edge for each $P_i$ and call it $e_i$. Since, there are no edges from end points of $e_i$ to any $S_j$ for $j \geq 2$, we know that each $e_i$ must be covered by $B_1$ or $\bar{B}_1$. But since end points of $e_i$ are not adjacent to $s_{13}^u$ and $s_{13}^v$, $e_i$ cannot be covered by $\bar{B}_1$. Thus, each $e_i$ has to be covered by $B_1$.

Now, we construct an assignment $A$ according to $B_1$ as follows. For each $i \in [n]$, since $\{B_1, \bar{B}_1\}$ is a duplex biclique, $B_1$ contains exactly one among $h_i^u$ and $h_i^v$. If $h_i^u \in U(B_1)$, then we assign $x_i = true$ in the assignment $A$. Otherwise, i.e, if $h_i^v \in V(B_1)$, then we assign $x_i = false$ in $A$. We claim that $A$ must be a satisfying assignment for $\psi$, which can be observed as follows. Consider an arbitrary clause $C_j$. Let $x_i$ be the variable corresponding to $e_j$. Suppose $x_i$ occurs in positive form in $C_j$. From the construction of edges between $H$ and $P$, we know that there cannot be an edge from $h_i^v$ and end points of $e_j$. Hence, $B_1$ cannot contain $h_i^v$. But, since $B_1$ should contain one of $h_i^u$ and $h_i^v$, it should contain $h_i^u$. This means that we assigned $x_i = true$ in $A$ and hence $A$ satisfies clause $C_j$. Symmetrically, we can argue that if $x_i$ occurred in the negative form in $C_j$, then we would have assigned $x_i = false$ in $A$. Thus, $A$ satisfies all the clauses of $\psi$. ◀

The statement of Theorem 6 follows from Lemmas 17, 18, and 19.

## 4 Approximation of BicliqueCover and BicliquePartition

In this section, we use the exponential kernel given in [6] to get a polynomial time approximation algorithm for the optimization versions of BicliqueCover and BicliquePartition, achieving an approximation ratio of $\frac{n}{\log_2 n}$. In the optimization versions of the problems, we are required to find the biclique cover/partition with the smallest size. First, we give a useful definition and then proceed towards describing the algorithm. We give the complete algorithm including the reduction rule used for kernelization in [6].

▶ **Definition 20** (Star). A star of a vertex $w$ in a graph is the subgraph induced by $\{w\} \cup N(w)$.

**Algorithm:** Let $G$ be the input graph. If there exist twin vertices $w_1$ and $w_2$ in $G$, we remove one of them (say, $w_2$) and the edges incident on it and then recurse by finding the biclique cover/partition of the remaining graph $G \setminus w_2$. After finding the biclique cover/partition of $G \setminus w_2$, we add $w_2$ to all the bicliques in the solution that contain $w_1$. If $G$ contains no twins, we output the set of stars of all vertices in $U(G)$ as the biclique cover/partition. We prove in Theorem 21 that this algorithm achieves an approximation ratio of $n/\log_2 n$.

▶ **Theorem 21.** *The above algorithm correctly finds a biclique cover/partition of $G$ whose size is at most $\frac{n}{\log_2 n} \cdot k$, where $k$ is the size of the minimum biclique cover/partition and $n$ is the number of vertices of $G$. In other words, the algorithm is a polynomial time approximation algorithm for the optimization versions of* BicliqueCover *and* BicliquePartition*, giving an approximation ratio of $\frac{n}{\log_2 n}$.*

**Proof.** First let us prove the correctness, i.e, we indeed output a biclique cover/partition of $G$. The correctness of reducing twins is clear and is already proven in [6]. In the case when $G$ does not contain twins, the correctness follows because each star is a biclique, and each edge of $G$ is present in exactly one of the stars of the vertices in $U(G)$. Let $k$ be the number of bicliques in the optimal biclique cover/partition. Since we do not add any extra bicliques while reducing twins, we need only to consider the case when $G$ has no twins, in order to estimate the size of the biclique/cover partition output by our algorithm. In this case, we know that $|U(G)|, |V(G)| \leq 2^k$ from [6]. Hence, the number of bicliques in the cover/partition that we output is at most $\min\{n, 2^k\} = \frac{\min\{n,2^k\}}{k} \cdot k \leq \frac{n}{\log_2 n} \cdot k$. ◀

**Acknowledgements.** We are grateful to Erik Jan van Leeuwen for helpful discussions.

## References


1. Parinya Chalermsook, Sandy Heydrich, Eugenia Holm, and Andreas Karrenbauer. Nearly Tight Approximability Results for Minimum Biclique Cover and Partition. In *Algorithms – ESA 2014*, volume 8737 of *LNCS*, pages 235–246. Springer Berlin Heidelberg, 2014. `doi:10.1007/978-3-662-44777-2_20`.
2. Marek Cygan, Marcin Pilipczuk, and Michał Pilipczuk. Known algorithms for edge clique cover are probably optimal. *SIAM Journal on Computing*, 45(1):67–83, 2016. `doi:10.1137/130947076`.
3. Alina Ene, William Horne, Nikola Milosavljevic, Prasad Rao, Robert Schreiber, and Robert E. Tarjan. Fast exact and heuristic methods for role minimization problems. In *SACMAT'08: Proceedings of the 13th ACM symposium on Access control models and technologies*, pages 1–10, New York, NY, USA, 2008. ACM. `doi:10.1145/1377836.1377838`.
4. David Eppstein, Michael T. Goodrich, and Jeremy Yu Meng. Confluent layered drawings. *Algorithmica*, 47:439–452, 2007. `doi:10.1007/s00453-006-0159-8`.
5. Peter C. Fishburn and Peter L. Hammer. Bipartite dimensions and bipartite degrees of graphs. *Discrete Math.*, 160(1-3):127–148, 1996. `doi:10.1016/0012-365X(95)00154-O`.
6. Herbert Fleischner, Egbert Mujuni, Daniël Paulusma, and Stefan Szeider. Covering graphs with few complete bipartite subgraphs. *TCS*, 410(21-23):2045–2053, 2009. `doi:10.1016/j.tcs.2008.12.059`.
7. Floris Geerts, Bart Goethals, and Taneli Mielikäinen. Tiling databases. In *Discovery Science*, pages 278–289. Springer, 2004.
8. David A. Gregory, Norman J. Pullman, Kathryn F. Jones, and J. Richard Lundgren. Biclique coverings of regular bigraphs and minimum semiring ranks of regular matrices. *J. Comb. Theory, Ser. B*, 51(1):73–89, 1991. `doi:10.1016/0095-8956(91)90006-6`.
9. Hermann Gruber and Markus Holzer. Inapproximability of Nondeterministic State and Transition Complexity Assuming P≠NP. In Tero Harju, Juhani Karhumäki, and Arto Lepistö, editors, *Developments in Language Theory*, volume 4588 of *Lecture Notes in Computer Science*, pages 205–216. Springer, 2007. `doi:10.1007/978-3-540-73208-2_21`.
10. Tao Jiang and B. Ravikumar. Minimal NFA Problems are Hard. *SIAM Journal on Computing*, 22(6):1117–1141, 1993. `doi:10.1137/0222067`.
11. Dana S. Nau, George Markowsky, Max A. Woodbury, and D. Bernard Amos. A mathematical analysis of human leukocyte antigen serology. *Math. Biosciences*, 40(3-4):243–270, 1978. `doi:10.1016/0025-5564(78)90088-3`.
12. Igor Nor, Danny Hermelin, Sylvain Charlat, Jan Engelstadter, Max Reuter, Olivier Duron, and Marie-France Sagot. Mod/Resc parsimony inference: Theory and application. *Inf. Comput.*, 213:23–32, 2012. `doi:10.1016/j.ic.2011.03.008`.
13. James Orlin. Contentment in graph theory: Covering graphs with cliques. *Indagationes Mathematicae (Proceedings)*, 80(5):406–424, 1977. `doi:10.1016/1385-7258(77)90055-5`.