Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2014, Article ID 934630, 13 pages
http://dx.doi.org/10.1155/2014/934630

Research Article

Partition of a Binary Matrix into k ($k \geq 3$) Exclusive Row and Column Submatrices Is Difficult

Peiqiang Liu,[1,2,3] Daming Zhu,[3] Jinjie Xiao,[1,2] Qingsong Xie,[1,2] and Yanyan Mao[1]

[1] *School of Computer Science and Technology, Shandong Institute of Business and Technology, Yantai 264005, China*
[2] *Key Laboratory of Intelligent Information Processing in Universities of Shandong (Shandong Institute of Business and Technology), Yantai 264005, China*
[3] *School of Computer Science and Technology, Shandong University, Jinan 250101, China*

Correspondence should be addressed to Peiqiang Liu; liupeiqiang@gmail.com

Received 25 March 2014; Revised 26 May 2014; Accepted 27 May 2014; Published 3 July 2014

Academic Editor: Anders Eriksson

A biclustering problem consists of objects and an attribute vector for each object. Biclustering aims at finding a bicluster—a subset of objects that exhibit similar behavior across a subset of attributes, or vice versa. Biclustering in matrices with binary entries ("0"/"1") can be simplified into the problem of finding submatrices with entries of "1." In this paper, we consider a variant of the biclustering problem: the k-submatrix partition of binary matrices problem. The input of the problem contains an $n \times m$ matrix with entries ("0"/"1") and a constant positive integer k. The k-submatrix partition of binary matrices problem is to find exactly k submatrices with entries of "1" such that these k submatrices are pairwise row and column exclusive and each row (column) in the matrix occurs in exactly one of the k submatrices. We discuss the complexity of the k-submatrix partition of binary matrices problem and show that the problem is NP-hard for any $k \geq 3$ by reduction from a biclustering problem in bipartite graphs.

1. Introduction

The problems considered in this paper are biclustering problems. Biclustering is an important optimization problem with applications in many fields including bioinformatics (especially in gene expression data analysis), identifying web communities, network information security analysis, and many more [1–3]. Biclustering is also known as block clustering, coclustering, or two-way clustering. The earliest biclustering algorithm that can be found in the literature is the so-called direct clustering by Hartigan in the 1970s [4, 5]. Since then, many approaches to biclustering have been proposed, such as the direct clustering algorithm [4], the node-deletion algorithm [6], the FLOC algorithm [7], the biclustering via spectral bipartite graph partitioning algorithm [8], the biclustering via GIBBS sampling algorithm [9], and the algorithm for finding an order-preserving submatrix [10]. For more on biclustering, see [3, 11, 12].

The basic model for biclustering is as follows. Let a dataset of m objects and n attributes be given as a matrix $A = [a_{ij}]_{m*n}$, where the value of a_{ij} is the value of the jth attribute of the ith object; the simplest aim of biclustering is to find a subset of rows (objects) that exhibit similar behavior across a subset of columns (attributes), or vice versa. In this case, the combination of the subset of objects and the subset of attributes is called a bicluster. A bicluster forms a contiguous rectangle after an appropriate reordering of rows and columns; that is, a bicluster is a submatrix of A.

In some applications, the main goal of biclustering is to simultaneously find many submatrices (biclusters) in a matrix. Madeira and Oliveira discussed this issue and summarized eight biclustering patterns [11]. Five of these patterns are presented in Figure 1: (1) exclusive row and column biclusters (Figure 1(a)), with each row (column) occurring in exactly one bicluster; (2) exclusive row biclusters (Figure 1(b)), with each row occurring in exactly one bicluster and each column occurring in at least one bicluster; (3) exclusive column biclusters (Figure 1(c)), with each column occurring in exactly one bicluster and each row occurring in at least one bicluster; (4) checkerboard structure (Figure 1(d)), with each entry of the matrix occurring in exactly one bicluster; and (5) arbitrarily positioned overlapping biclusters

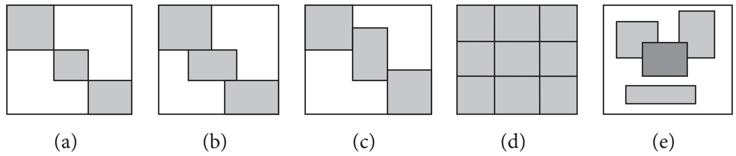

Figure 1: Biclustering patterns: (a) exclusive row and column biclusters, (b) exclusive row biclusters, (c) exclusive column biclusters, (d) checkerboard structure, and (e) arbitrarily positioned overlapping biclusters.

(Figure 1(e)), with no limiting condition of rows (columns) overlapping or entries overlapping.

In many applications, a biclustering problem consists of a matrix that has entries of "1" or "0," which is also called a binary matrix. The goal of biclustering in binary matrices is to find submatrices with entries of "1." For example, when applying biclustering to text mining, a dataset of m documents and n words is arranged in a binary matrix $A = [a_{ij}]_{m*n}$, where rows correspond to documents and columns correspond to words. If an entry (i, j) of the matrix is "1," then word j is present in document i. If the entry is "0," then the word is not present. The question is whether we can find k submatrices with entries of "1" such that these submatrices are pairwise row and column exclusive, and each row (column) occurs in exactly one submatrix. Clearly, if the answer is "yes," then these documents can be partitioned into k groups, and documents in the same group have a good chance of belonging to the same domain.

The text mining problem described above can be abstracted as the k-submatrix partition of binary matrices problem (k-SPBM). Given an $n \times m$ binary matrix and a constant positive integer k, k-SPBM is to find k submatrices with entries "1" such that these k submatrices are pairwise row and column exclusive and each row (column) of the matrix occurs in exactly one of these submatrices. The bicluster pattern of k-SPBM belongs to pattern (a) in Figure 1. To the best of our knowledge, the hardness of k-SPBM remains an open problem, for each $k \geq 3$.

We will show that k-SPBM is NP-complete by reduction from the partition of a bipartite graph into k bicliques problem (k-PBB) that is a variant of biclustering problems in bipartite graphs; that is, an instance of k-PBB is a bipartite graph. A bipartite graph is a graph whose vertex set can be partitioned into two disjoint sets such that no two graph vertices within the same set are adjacent. For a biclustering problem in bipartite graphs, the goal is to find bicliques according to some scoring criterion. A biclique, which is also called a complete bipartite graph, is a special type of bipartite graph for which every pair of vertices in the two sets are adjacent.

In recent years, much study has focused on algorithms and complexity of biclustering problems in bipartite graphs. Peeters, Dawande et al., and Amit proved that the maximum edge biclique problem [13], the maximum edge weight biclique problem [14], the bicluster graph editing problem [15], the exact cardinality biclique problem [16], and the minimum edge deletion biclique problem [16], among others, are NP-complete.

When Heydari et al. studied the biclustering of an attack graph problem in information security, they first proposed the partition of a bipartite graph into bicliques problem (PBB). Heydari et al. showed that PBB is NP-complete [17]. Furthermore, Bein et al. discussed the k-PBB problem, where k is a constant positive integer. Here, k-PBB is a parameterized version of PBB; it aims at partitioning the vertex set of a bipartite graph into k subsets such that each vertex subset can induce a biclique. k-PBB defines a family of problems for any $k \geq 3$. Bein et al. first proposed the k-PBB problem and indicated that the question of whether k-PBB is NP-complete for $k \geq 3$ remains open [18].

Contribution of this paper is that it focuses on the complexity of several biclustering problems. The main result shows that 3-PBB, k-PBB ($k > 3$), and k-SPBM ($k \geq 3$) are all NP-complete.

Organization of the paper is as follows: in Section 2, we introduce the k-PBB and k-SPBM problems. In Section 3, we first show that 3-PBB is NP-complete by reduction from a variant of the monotone one-in-three 3SAT problem (MO3), which is a well-known NP-complete problem [19, 20], and, then, we show that k-PBB ($k > 3$) is NP-complete by reduction from 3-PBB. In Section 4, we prove that k-SPBM ($k \geq 3$) is NP-complete by reduction from k-PBB. Finally, in Section 6, we present our conclusions.

2. Preliminaries

In this paper, we study two problems: the k-SPBM problem and the k-PBB problem. Next, we present the formal descriptions of k-SPBM and k-PBB.

(1) The k-submatrix partition of binary matrices problem (k-SPBM).

The input to the k-SPBM problem is typically a binary matrix. Let $A = [a_{ij}]_{m*n}$ be an $n \times m$ binary matrix. Denote the set of row vectors and the set of column vectors by $R = \{1, \ldots, m\}$ and $C = \{1, \ldots, n\}$, respectively. Suppose $R_1 \sqsubseteq R$ and $C_1 \sqsubseteq C$; then the public entries of row vectors $\{\alpha_i \mid a_{ij} \in \alpha_i, i \in R_1, j \in C\}$ and column vectors $\{\beta_j \mid a_{ij} \in \beta_j, i \in R, j \in C_1\}$ form a matrix $[a_{ij} : i \in R_1, j \in C_1]$ that is called a submatrix of A induced by R_1 and C_1, which is denoted by $A[R_1, C_1]$. Clearly, $A = A[R, C]$. Let $A_1 = A[R_1, C_1]$, $A_2 = A[R_2, C_2]$ be submatrices of A. If $R_1 \cap R_2 = \emptyset$, then A_1 and A_2 are row exclusive; if $C_1 \cap C_2 = \emptyset$, then A_1 and A_2 are column exclusive. k-SPBM is to find exactly k exclusive row and column submatrices with entries of "1" in a binary matrix.

The k-SPBM problem can be stated formally as follows.

> Instance: an $m \times n$ binary matrix A, and a constant positive integer k.
>
> Question: are there k submatrices with entries "1" $A[R_1, C_1], \ldots, A[R_k, C_k]$ of A such that the k submatrices are pairwise row and column exclusive, and $R_1 \cup \cdots \cup R_k = \{1, 2, \ldots, m\}$, $C_1 \cup \cdots \cup C_k = \{1, 2, \ldots, n\}$?

$A[R_1, C_1], \ldots, A[R_k, C_k]$ are called a k-submatrix partition of A.

(2) The partition of a bipartite graph into k-bicliques problem (k-PBB).

An instance of k-PBB is a bipartite graph. All bipartite graphs in the paper are simple bipartite graphs, that is, do not contain parallel edges or self-loops. Let $G = (X, Y, E)$ be a bipartite graph. For convenience in writing, vertices in X are called left-vertices, and vertices in Y are called right-vertices of G. In other words, X and Y are the left-vertex set and right-vertex set of G, respectively. We denote by $E(G)$ and $V(G)$ its set of edges and its set of vertices, respectively. For a vertex $v \in V(G)$, we denote the set of neighbors of vertex v by $\Gamma(v)$. A biclique in G corresponds to a subset of $V(G)$, say, $C = A \cup B$, such that $A \subseteq X$, $B \subseteq Y$, and for each $u \in A$, $v \in B$ the edge $(u, v) \in E$.

We say that there exists a k-biclique partition for a bipartite graph G if $V(G)$ can be partitioned into exactly k disjoint sets $V_1, V_2, \ldots, V_k$ such that, for $1 \leq i \leq k$, the subgraph induced by V_i is a biclique. The k-PBB problem is the problem of determining whether there is a k-biclique partition for a bipartite graph G, where k is a constant positive integer. The k-PBB problem can be stated formally as follows.

> Instance: a finite bipartite graph $G = (X, Y, E)$ and a constant positive integer $k \leq \min\{|X|, |Y|\}$.
>
> Question: does there exist a k-biclique partition for G?

3. The Complexity of k-PBB

In this section, we first show the NP-completeness of k-PBB when $k = 3$ (i.e., 3-PBB). We then show that k-PBB is NP-complete for any constant integer k $(k > 3)$ by reduction from 3-PBB. Finally, we conclude that k-PBB is NP-complete for any constant integer k $(k \geq 3)$.

3.1. The NP-Completeness of 3-PBB. In order to prove the hardness of 3-PBB, we first introduce the monotone one-in-three 3SAT problem (MO3), which was proved to be NP-complete by Schaefer in 1978 [19]. Then, we show that a variant of MO3 is NP-complete. Finally, we prove that 3-PBB is NP-complete by reduction from MO3.

Below we define the terms we will use in describing MO3. Let $U = \{u_1, u_2, \ldots, u_n\}$ be a set of Boolean variables. If $u_i \in U$, then u_i and $\overline{u}_i$ are literals over U. u_i is called a positive variable, and $\overline{u}_i$ is called a negative variable. A truth assignment for U is a function $t :\to T, F$. For $u_i \in U$, if $t(u_i) = F$, we say that u_i is "TRUE" under t; if $f(u_i) = F$, we say that u_i is "FALSE."

The MO3 problem, which is a variant of 3SAT, is specified as follows.

> Instance: set $U = \{u_1, u_2, \ldots, u_n\}$ of Boolean variables, collection $C = \{c_1, c_2, \ldots, c_m\}$ of clauses over U, where each clause $c \in C$ has $|c| = 3$, and c does not contain a negative variable; that is, $c_i = \{u_x, u_y, u_z\}$, u_x, u_y, $u_z \in U$, $1 \leq i \leq m$.
>
> Question: is there a truth assignment for U such that each clause in C has exactly one true literal?

In the MO3 problem, a clause over U contains only positive variables. For an MO3 instance, a clause over U is satisfied by a truth assignment if and only if it has exactly one "TRUE" literal (and thus exactly two "FALSE" literals) under the assignment. A collection C of clauses over U is satisfiable if and only if there exists a truth assignment for U that simultaneously satisfies all the clauses in C.

For example, we are given Boolean variable set $U = \{u_1, u_2, u_3, u_4\}$, and a clause collection $C = \{c_1, c_2, c_3, c_4\}$, where $c_1 = \{u_1, u_2, u_3\}$, $c_2 = \{u_2, u_3, u_4\}$ and $c_3 = \{u_1, u_2, u_4\}$. Let $[a(u_1), a(u_2), a(u_3), a(u_4)] = [F, T, F, F]$; then, the values of the variables in c_1, c_2, and c_3 are (F, T, F), (T, F, F), and (F, T, F), which means that c_1, c_2, and c_3 are satisfied. Therefore, $a(\cdot)$ is a feasible solution of this MO3 instance.

For an arbitrary MO3 instance, we can assume that the three literals in each clause are not from the same variable, in which case the clause is not satisfied. Moreover, a clause in which two literals are from the same variable can be transformed into six clauses with pairwise different variables. The approach is as follows.

Suppose that $c_k = \{u_i, u_i, u_j\}$ is a clause of an MO3 instance. We create four new variables u_{i1}, u_{i2}, u_{i3}, and u_{i4}. Then, we construct six clauses over u_i, u_j, and the four new variables: $c_k[1] = \{u_{i1}, u_{i2}, u_{i3}\}$, $c_k[2] = \{u_{i1}, u_{i2}, u_i\}$, $c_k[3] = \{u_{i2}, u_{i3}, u_i\}$, $c_k[4] = \{u_{i1}, u_{i2}, u_{i4}\}$, $c_k[5] = \{u_{i2}, u_{i3}, u_{i4}\}$, and $c_k[6] = \{u_i, u_{i4}, u_j\}$. Clearly, the clause $\{u_i, u_i, u_j\}$ is satisfied if and only if $a(u_i) = F$ and $a(u_j) = T$. Moreover, a truth assignment for the variables u_{i1}, u_{i2}, u_{i3}, and u_{i4} exists such that each clause in $c_k[1{\sim}6]$ is satisfied if and only if $a(u_i) = F$ and $a(u_j) = T$.

Thus, an arbitrary MO3 instance can be transformed into an MO3 instance with pairwise different variables in each clause in polynomial time. Therefore, we have Theorem 1.

Theorem 1. *MO3 with pairwise different variables in each clause is NP-complete.*

Throughout this paper, we assume without loss of generality that, for an instance of MO3, the three literals of each clause are pairwise different. Next, we discuss the complexity of 3-PBB; that is, we prove Theorem 2.

Theorem 2. *3-PBB is NP-complete.*

The proof of Theorem 2 consists of two steps. First, let a variable set $U = \{u_1, u_2, \ldots, u_n\}$ and a clause collection $C = \{c_1, c_2, \ldots, c_m\}$ be an instance of MO3; then we build a bipartite graph $B = (X[B], Y[B], E[B])$ that is an instance of 3-PBB.

Second, we show that C is satisfied if and only if there exists a 3-biclique partition for B.

3.1.1. The Construction of a Bipartite Graph B from an MO3 Instance. Given an instance of MO3, we build a bipartite graph B that is an instance of 3-PBB in three steps. In the first step, we construct three components T_{i1}, T_{i2}, and T_{i3} from the clause c_i $(1 \le i \le m)$. In the second step, we merge T_{i1}, T_{i2}, and T_{i3} into a bipartite graph B_i. In the final step, we merge m B_i's into a bipartite graph B.

Step 1. For each clause $c_i \in C$, we construct three components that are associated with the three literals in c_i. Each of these components is a bipartite graph.

Suppose that $c_i = \{u_x, u_y, u_z\} \in C$. Thus, we construct the components T_{i1}, T_{i2}, and T_{i3}. The three components contain vertices u_x, u_y, and u_z, which correspond to the variables u_x, u_y, and u_z of c_i, respectively. In the following, we will indiscriminately use the notation u_x, u_y, or u_z to represent a vertex or a variable.

The key idea used in this step of construction is that each of the three components contains a bipartite subgraph isomorphic to B_c illustrated in Figure 2. Moreover, for an arbitrary 3-biclique partition of T_{ij} $(j \in \{1, 2, 3\})$, the structure of T_{ij} ensures that

(1) $\{l_0, r_0\}$, $\{l_1, r_1\}$, and $\{l_2, r_2\}$ are always partitioned into different bicliques,

(2) u_x, u_y, or u_z only belongs to those bicliques that contain $\{l_0, r_0\}$ or $\{l_1, r_1\}$.

This is our basic way of encoding the idea that $u_i \in U$ can be set to either T or F; if u_i belongs to a biclique that contains $\{l_0, r_0\}$, we set $u_i = F$, and if u_i belongs to a biclique that contains $\{l_1, r_1\}$, we set $u_i = T$.

$T_{i1} = (L_{i1}, R_{i1}, E_{i1})$ contains 13 vertices and 21 edges, as shown in Figure 3(a). Figures 3(b)–3(d) show three 3-biclique partitions of T_{i1}. In Figures 3(b)–3(d), the vertices with the same color induce a biclique. In fact, there exist exactly three 3-biclique partitions for T_{i1}, as shown in Figures 3(b)–3(d).

Lemma 3. *For an arbitrary 3-biclique partition of* T_{i1}, $\{l_0, r_0\}$, $\{l_1, r_1\}$, *and* $\{l_2, r_2\}$ *are always partitioned into different bicliques. (For the sake of readability, we defer the proof to the Appendix. The complete proof is in Appendix A.)*

Based on Lemma 3, each vertex in T_{i1} is assigned a value for denoting a 3-biclique partition of T_{i1} by the assignment function $f : (T_{i1}) \rightarrow \{0, 1, 2\}$. According to a 3-biclique partition of T_{i1}, the function $f(\cdot)$ is defined as

$$f(v) = \begin{cases} 0 & v \text{ and } (l_0, r_0) \text{ belong to the same biclique} \\ 1 & v \text{ and } (l_1, r_1) \text{ belong to the same biclique} \\ 2 & v \text{ and } (l_1, r_1) \text{ belong to the same biclique.} \end{cases} \quad (1)$$

Lemma 4. *There exist exactly three 3-biclique partitions for* T_{i1}. *Accordingly, the values of the vertices* u_x, d_{i2}, *and* d_{i3} *are* $(f(u_x), f(d_{i2}), f(d_{i3})) \in \{(1, 2, 1), (0, 2, 2), (0, 0, 1)\}$. *(The proof is in Appendix B.)*

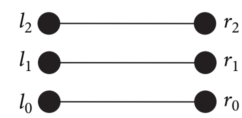

Figure 2: B_c.

$T_{i2} = (L_{i2}, R_{i2}, E_{i2})$ is presented in Figure 4(a). T_{i2} contains 12 vertices and 17 edges. Figures 4(b) and 4(c) show two 3-biclique partitions of T_{i2}. In Figures 4(b) and 4(c), the vertices with the same color induce a biclique. In fact, there exist exactly two 3-biclique partitions for T_{i2}, as shown in Figures 4(b) and 4(c).

Lemma 5. *For an arbitrary 3-biclique partition of* T_{i2}, $\{l_0, r_0\}$, $\{l_1, r_1\}$, *and* $\{l_2, r_2\}$ *are always partitioned into different bicliques. (The proof is in Appendix C.)*

Based on Lemma 5, the same approach that was used for T_{i1} is used to assign values to the vertices of T_{i2}. Again, we suppose that $f : V(T_{i2}) \rightarrow \{0, 1, 2\}$ is the assignment function for T_{i2}. The assignment method for $f(\cdot)$ is the same as that in Formula (1).

Lemma 6. *There exist exactly two 3-biclique partitions for* T_{i2}. *Accordingly, the values of the vertices* u_y *and* d_{i2} *are* $(f(u_y), f(d_{i2})) \in \{(0, 2), (1, 0)\}$. *(The proof is in Appendix D.)*

$T_{i3} = (L_{i3}, R_{i3}, E_{i3})$ is isomorphic to T_{i2}. To obtain T_{i3} in Figure 5, we only need to rename the vertices d_{i2}, u_y, m_{i1}, l_2, l_1, l_0, n_{i1}, n_{i2}, n_{i3}, r_2, r_1, and r_0 of T_{i2} as d_{i3}, u_z, o_{i1}, l_2, l_0, l_1, p_{i1}, p_{i2}, p_{i3}, r_2, r_0, and r_1, respectively. We present Lemmas 7 and 8 on T_{i3} without proof. The proofs are similar to those of Lemmas 5 and 6.

Lemma 7. *For an arbitrary 3-biclique partition of* T_{i3}, $\{l_0, r_0\}$, $\{l_1, r_1\}$, *and* $\{l_2, r_2\}$ *are always partitioned into different bicliques.*

Again, we assign the vertices of T_{i3} using Formula (1).

Lemma 8. *There exist exactly two 3-biclique partitions for* T_{i3}. *Accordingly, the values of the vertices* u_z *and* d_{i3} *are* $(f(u_z), f(d_{i3})) \in \{(1, 2), (0, 1)\}$.

Step 2. We merge T_{i1}, T_{i2}, and T_{i3} into a bipartite graph B_i $(1 \le i \le m)$ that is associated with the clause $c_i \in C$.

For the bipartite graphs $T_{11}, T_{12}, T_{13}, \ldots, T_{m1}, T_{m2}$, and T_{m3} $(1 \le i \le m)$ constructed as before, we first merge T_{i1}, T_{i2}, and T_{i3} into B_i before building an instance B of 3-PBB. Suppose that $B_i = (X[B_i], Y[B_i], E[B_i])$ and $B = (X[B], Y[B], E[B])$.

The left and right vertex sets of B_i are obtained by merging the left and right vertex sets of T_{i1}, T_{i2}, and T_{i3}:

$$\begin{aligned} X[B_i] &= L_{i1} \cup L_{i2} \cup L_{i3}, \\ Y[B_i] &= R_{i1} \cup R_{i2} \cup R_{i3}. \end{aligned} \quad (2)$$

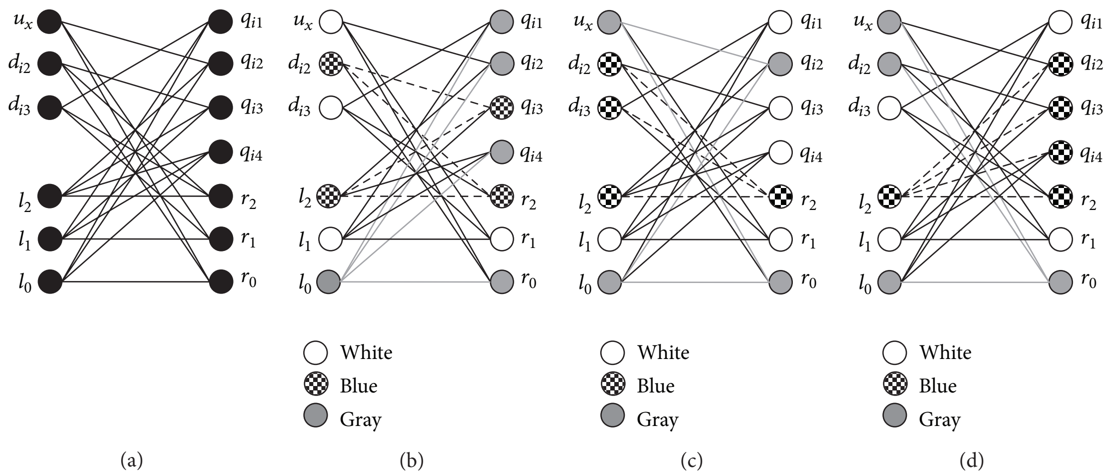

Figure 3: T_{i1} and the three 3-biclique partitions of T_{i1}.

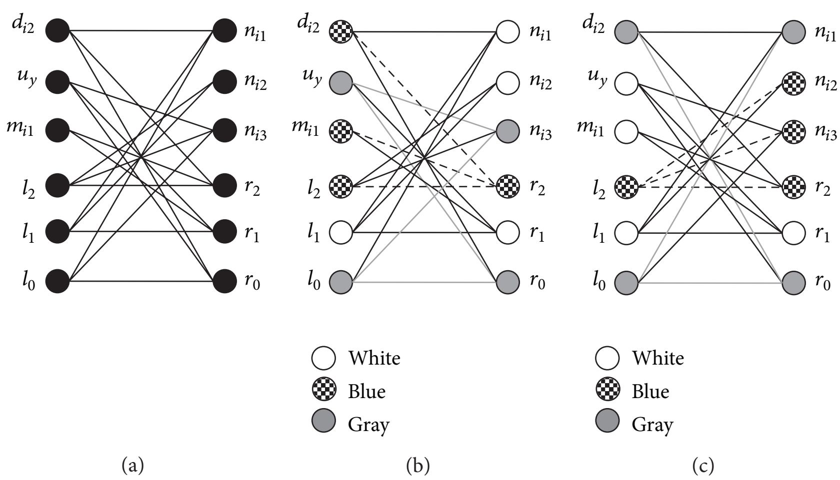

Figure 4: T_{i2} and the two 3-biclique partitions of T_{i2}.

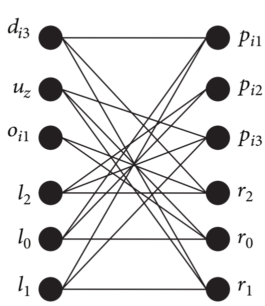

Figure 5: T_{i3}.

In words, each vertex of $V(B_i)$ belongs to $V(T_{i1})$, $V(T_{i2})$, or $V(T_{i3})$, and vice versa, and vertices with the same vertex label in T_{i1}, T_{i2}, and T_{i3} are merged into one vertex in B_i as follows: the vertices with the same label, including l_0, r_0, l_1, r_1, l_2, and r_2 in T_{i1}, T_{i2}, and T_{i3}, are merged into one group of vertices labeled l_0, r_0, l_1, r_1, l_2, and r_2 in B_i; two vertices d_{i2} in T_{i1} and d_{i2} in T_{i2} are merged into one vertex labeled d_{i2} in B_i; and two vertices d_{i3} in T_{i1} and d_{i3} in T_{i3} are merged into one vertex labeled d_{i3} in B_i. In T_{i2} and T_{i3}, no other vertices exist with the same label except for l_0, r_0, l_1, r_1, l_2, and r_2.

$E(B_i)$ has two portions. Let $E[B_i] = E_1[B_i] \cup E_2[B_i]$. The first portion $E_1[B_i]$ can be obtained by merging $E(T_{i1})$, $E(T_{i2})$, and $E(T_{i3})$:

$$E_1[B_i] = \{(l,r) \mid (l,r) \in E_{i1} \text{ or } (l,r) \in E_{i2} \text{ or } (l,r) \in E_{i3}\}. \tag{3}$$

Clearly, the edges with the same vertex label in T_{i1}, T_{i2}, and T_{i3} are merged into one edge of $E_1[B_i]$, respectively, and T_{i1}, T_{i2}, and T_{i3} are bipartite subgraphs of B_i. To ensure that there exists a 3-biclique partition for B_{i1}, we require the addition of more edges as the other portion of $E(B_i)$ as follows: the edges of T_{i1} and T_{i2} among the nonpublic vertices are added, as denoted by $E_2[B_i,1,2]$; the edges of T_{i2} and T_{i3} among the nonpublic vertices are added, as denoted by $E_2[B_i,2,3]$; and the edges of T_{i3} and T_{i1} among the nonpublic vertices are added, as denoted by $E_2[B_i,3,1]$. For two graphs, if a vertex label occurs exactly one of the two graphs, then the vertex corresponding to this label is called a nonpublic vertex. These three additional edge sets are formally stated as follows:

$$\begin{aligned} E_2[B_i,1,2] = {} & \{(l,r) \mid l \in L_{i1} - (L_{i1} \cap L_{i2}), \\ & r \in R_{i2} - (R_{i1} \cap R_{i2})\} \\ & \cup \{(l,r) \mid l \in L_{i2} - (L_{i1} \cap L_{i2}), \\ & r \in R_{i1} - (R_{i1} \cap R_{i2})\}, \\ E_2[B_i,2,3] = {} & \{(l,r) \mid l \in L_{i2} - (L_{i2} \cap L_{i3}), \\ & r \in R_{i3} - (R_{i2} \cap R_{i3})\} \\ & \cup \{(l,r) \mid l \in L_{i3} - (L_{i2} \cap L_{i3}), \\ & r \in R_{i2} - (R_{i2} \cap R_{i3})\}, \\ E_2[B_i,3,1] = {} & \{(l,r) \mid l \in L_{i3} - (L_{i3} \cap L_{i1}), \\ & r \in R_{i1} - (R_{i3} \cap R_{i1})\} \\ & \cup \{(l,r) \mid l \in L_{i1} - (L_{i3} \cap L_{i1}), \\ & r \in R_{i3} - (R_{i3} \cap R_{i1})\}. \end{aligned} \tag{4}$$

Hence, the second portion of $E(B_i)$ can be obtained:

$$E_2[B_i] = E_2[B_i,1,2] \cup E_2[B_i,2,3] \cup E_2[B_i,3,1]. \tag{5}$$

For B_i and its bipartite subgraphs T_{i1}, T_{i2}, and T_{i3}, Proposition 9 holds.

Proposition 9. *A bipartite subgraph of B_i induced by $V(T_{ij})$ is isomorphic to T_{ij}, where $j \in \{1,2,3\}$. (The proof is in Appendix E.)*

Figure 6 illustrates the process of building B_i from T_{i1}, T_{i2}, and T_{i3}. The meaning of Figure 6 is as follows.

(1) Figure 6(a) shows the public vertices. The white vertex set is a public vertex set of T_{i1}, T_{i2}, and T_{i3}. The gray vertex d_{i2} is a public vertex of T_{i1} and T_{i2}. The blue vertex d_{i3} is a public vertex of T_{i1} and T_{i3}.

(2) Figure 6(b) depicts how to obtain $V(B_i)$ and $E_1[B_i]$. The white vertices of T_{i1}, T_{i2}, and T_{i3}, the gray vertex of T_{i1} and T_{i2}, and the blue vertex of T_{i1} and T_{i3} are merged together, respectively. Here, u_x, u_y, and u_z cannot be merged because they are pairwise different. As shown in Figure 6(b), the edge set is $E_1[B_i]$.

(3) Figure 6(c) displays the following additional edge sets: $E_2[B_i,1,2]$ (yellow edge set), $E_2[B_i,2,3]$ (black edge set), and $E_2[B_i,3,1]$ (red edge set). For the sake of clarity, $E_1[B_i]$ is not illustrated in Figure 6(c). If $E_1[B_i]$ is added to Figure 6(c), then B_i will be obtained.

Step 3. We merge $B_1, B_2, \ldots, B_m$ into B that is associated with an instance of MO3.

The steps used to merge B_i $(1 \le i \le m)$ are similar to those in merging T_{i1}, T_{i1}, and T_{i3} as above. $V(B)$ is obtained by merging $V(B_i)$'s $(1 \le i \le m)$:

$$\begin{aligned} X[B] &= \bigcup_{i=1}^{m} X[B_i], \\ Y[B] &= \bigcup_{i=1}^{m} Y[B_i]. \end{aligned} \tag{6}$$

In words, each vertex of $V(B)$ belongs to $V(B_i)$ $(1 \le i \le m)$ and vice versa, and vertices with the same vertex label in B_i's are merged into one vertex in B as follows: the m group vertices labeled $\{l_0, r_0, l_1, r_1, l_2, r_2\}$ in $B_1, B_2, \ldots, B_m$ are merged into one group in B and are still labeled $\{l_0, r_0, l_1, r_1, l_2, r_2\}$, and if a variable $u_i \in U$ appears t times in the clause collection C, then in B, the t vertices labeled u_i in t B_i's are merged into one vertex u_i. Therefore, each variable corresponds to exactly one vertex in B.

$E(B)$ has two portions. Let $E[B] = E_1[B] \cup E_2[B]$. The first portion $E_1[B]$ can be obtained by merging $E(B_1)$, $E(B_2), \ldots, E(B_m)$; that is,

$$E_1[B] = \{(l,r) \mid (l,r) \in E[B_i], 1 \le i \le m\}. \tag{7}$$

Similarly, the edges with the same vertex label in B_i's $(1 \le i \le m)$ are merged into one edge of $E_1[B]$, and B_i's are bipartite subgraph of B. To ensure that there exists a 3-biclique partition for B, we require the addition of more edges to be the other portion of $E(B)$: the edges among the nonpublic vertices of B_i and B_j are added as the edge set $E_2[B,i,j]$, where $i \ne j$. These additional edge sets are formally stated as follows:

$$\begin{aligned} E_2[B,i,j] = {} & \{(l,r) \mid l \in X[B_i] - (X[B_i] \cap X[B_j]), \\ & r \in Y[B_j] - (Y[B_i] \cap Y[B_j])\} \\ & \cup \{(l,r) \mid l \in X[B_j] - (X[B_i] \cap X[B_j]), \\ & r \in Y[B_i] - (Y[B_i] \cap Y[B_j])\}. \end{aligned} \tag{8}$$

Consequently, the second portion of $E(B)$ can be obtained:

$$E_2[B] = \bigcup_{i=1}^{m-1} \bigcup_{j=j+1}^{m} E_2[B,i,j]. \tag{9}$$

This completes the construction of the bipartite graph B. B obtained by merging m B_i's has at most $23 * m$ vertices and $85 \times m + C_m^2 \times 140$ edges. Therefore, B can be constructed in polynomial time.

For B, B_i, and T_{ij} $(1 \le i \le m, 1 \le j \le 3)$, Proposition 10 holds.

Proposition 10. *A bipartite subgraph of B induced by $V(B_i)$ is isomorphic to B_i, and a bipartite subgraph of B induced by $V(T_{ij})$ is isomorphic to T_{ij}, where $1 \le i \le m, 1 \le j \le 3$. (The proof is in Appendix F.)*

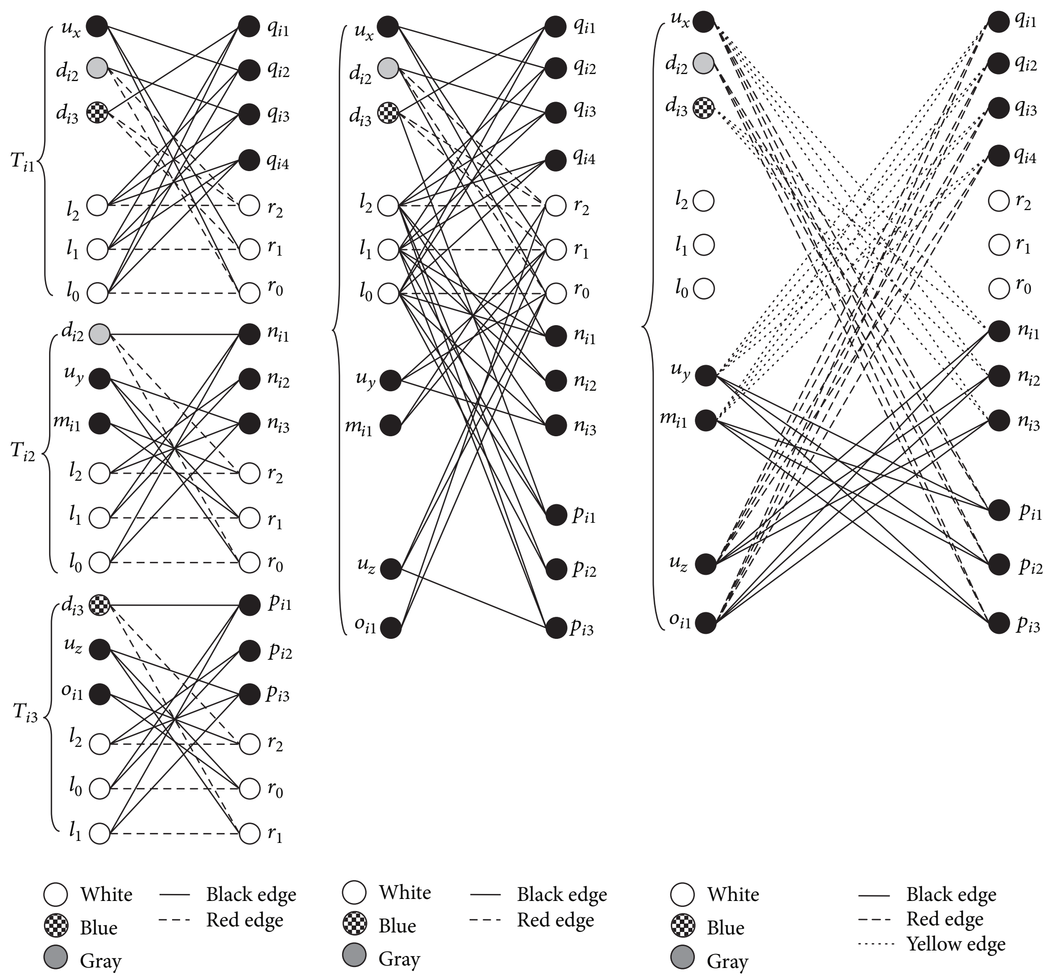

Figure 6: The construction of B_i by merging T_{i1}, T_{i2}, and T_{i3}.

Next, we show that there does not exist a 2-biclique partition for B; that is, if there exists a k-biclique partition for B, then $k \geq 3$.

Lemma 11. *If there exists a k-biclique partition for B, then $k \geq 3$.*

Proof. An arbitrary vertex $v \in L_{ij}$ is adjacent to at most two of r_0, r_1, and r_2 in T_{ij}. In the process of building B, there is no additional edge whose end vertex is in $\{l_0, l_1, l_2, r_0, r_1, r_2\}$. Therefore, an arbitrary vertex $v \in X[B]$ is adjacent to, at most, two of r_0, r_1, and r_2, such that r_0, r_1, and r_2 belong to at least two bicliques. If r_0, r_1, and r_2 are partitioned into two bicliques, then suppose that $\{r_x, r_y\}$ and $\{r_z\}$ are partitioned into different bicliques, where $x, y, z \in \{1, 2, 3\}$, $x \neq y$, $x \neq z$, $y \neq z$. Based on the process of building B, $(l_x, r_y) \notin E[B]$, and $(l_x, r_z) \notin E[B]$. Thus, l_x, r_x, r_y, and r_z of B belong to at least three bicliques, and the lemma follows. □

In the following, we prove that if there exists a 3-biclique partition for B, then Lemmas 12 and 13 hold.

Lemma 12. *If there exists at least one 3-biclique partition for B, then $\{l_0, r_0\}$, $\{l_1, r_1\}$, and $\{l_2, r_2\}$ will always be partitioned into three different bicliques for an arbitrary 3-biclique partition of B.*

Proof. There are only three edges (l_0, r_0), (l_1, r_1), and (l_2, r_2) between $\{l_0, l_1, l_2\}$ and $\{r_0, r_1, r_2\}$ in B. Therefore, if r_0, r_1, and r_2 are partitioned into three bicliques, then $\{l_0, r_0\}$, $\{l_1, r_1\}$, and $\{l_2, r_2\}$ must be partitioned into three bicliques. Moreover, because an arbitrary vertex $v \in X[B]$ is adjacent to at most two vertices of $\{r_0, r_1, r_2\}$, r_0, r_1, and r_2 belong to at least two bicliques in a 3-biclique partition of B. We next show that r_0, r_1, and r_2 do not belong to two bicliques using proof by contradiction.

Suppose that r_0, r_1, and r_2 belong to two bicliques. We can assume without loss of generality that $X[B] \cup Y[B] = V_{b1} \cup V_{b2} \cup V_{b3}$ is a 3-biclique partition of B, $\{r_x, r_y\} \subseteq V_{b1}$, $\{r_z\} \subseteq V_{b2}$, where $x, y, z \in \{0, 1, 2\}$, $x \neq y$, $y \neq z$, $x \neq z$. Because $(l_x, r_y) \notin E_B$, $(l_y, r_x) \notin E_B$, $(l_x, r_z) \notin E_B$, $(l_y, r_z) \notin E_B$, we have $\{l_x, l_y\} \subseteq V_{b3}$. Thus, there exists $T_{ij} = (L_{ij}, R_{ij}, E_{ij})$, $1 \leq i \leq m$, $j \in \{1, 2, 3\}$, such that $\{l_x, l_y, v_r\} \subseteq V_{b3}$, $v_r \in R_{ij} \setminus \{r_0, r_1, r_2\}$. Because $\{v_r, r_0, r_1, r_2\} \subseteq R_{ij}$, the vertices in R_{ij}

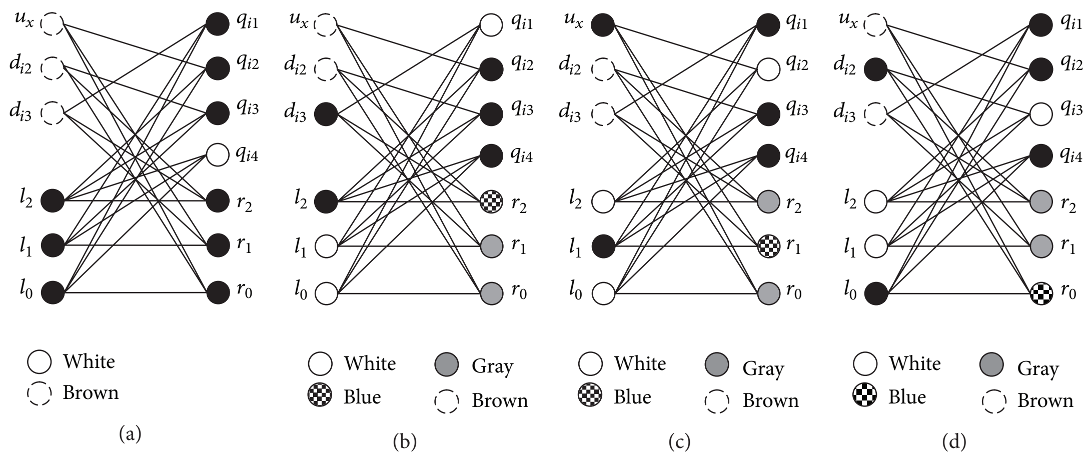

Figure 7: If T_{ij} is T_{i1}, then the vertices of L_{ij} will belong to three bicliques.

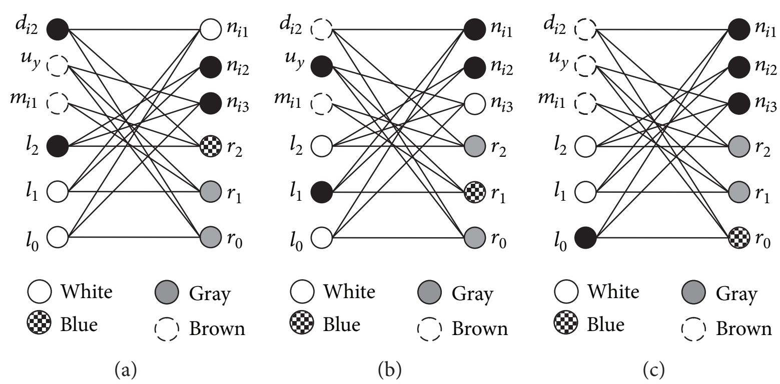

Figure 8: If T_{ij} is T_{i2}, then the vertices of L_{ij} will belong to three bicliques.

are partitioned into three bicliques in a 3-biclique partition of B. By Proposition 10, the edge subset of B induced by $V(T_{ij})$ is exactly E_{ij}. We next show that the vertices in L_{ij} also belong to three bicliques. Consider the following three cases: $T_{ij} = T_{i1}$, $T_{ij} = T_{i2}$, and $T_{ij} = T_{i3}$.

(1) If $T_{ij} = T_{i1}$, then $v_r \in \{q_{i1}, q_{i2}, q_{i3}, q_{i4}\}$. As shown in Figure 7(a), if v_r is q_{i4}, then there are no edges between $\{u_x, d_{i2}, d_{i3}\}$ and q_{i4}. Moreover, u_x, d_{i2}, and d_{i3} cannot simultaneously belong to either V_{b1} or V_{b2}. Therefore, the vertices in L_{ij} belong to three bicliques. As shown in Figures 7(b)–7(d), if $v_r \in \{q_{i1}, q_{i2}, q_{i3}\}$, we distinguish three cases. For an arbitrary $v_r \in \{q_{i1}, q_{i2}, q_{i3}\}$, v_r is not adjacent to two of u_x, d_{i2}, and d_{i3} (the brown vertices), and these two vertices cannot simultaneously belong to V_{b1} or V_{b2}. Therefore, the vertices of L_{ij} belong to three bicliques.

(2) If $T_{ij} = T_{i2}$, then $v_r \in \{n_{i1}, n_{i2}, n_{i3}\}$. As shown in Figures 8(a)–8(c), we distinguish three cases. For an arbitrary $v_r \in \{n_{i1}, n_{i2}, n_{i3}\}$, v_r is not adjacent to two of u_y, d_{i2}, and m_{i1} (the brown vertices), and these two vertices cannot simultaneously belong to V_{b1} or V_{b2}. Therefore, the vertices of L_{ij} belong to three bicliques.

(3) If $T_{ij} = T_{i3}$, then because T_{i2} and T_{i3} are isomorphic, the vertices of L_{ij} also belong to three bicliques.

By (1), (2), and (3), either the left or right vertices of T_{ij} are always partitioned into three bicliques in a 3-biclique partition of B. Thus, $V_{b1} \cap V(T_{ij})$, $V_{b2} \cap V(T_{ij})$, or $V_{b3} \cap V(T_{ij})$ induces a biclique in a 3-biclique partition of B, respectively. The three bicliques are a 3-biclique partition of T_{ij}. From Lemmas 3, 5, and 7, r_0, r_1, and r_2 must belong to three different bicliques, which contradicts the supposition that r_0, r_1, and r_2 belong to two bicliques. The lemma follows. □

Lemma 13. *Let $X[B] \cup Y[B] = V(B) = V_{b1} \cup V_{b2} \cup V_{b3}$ be a 3-biclique partition of B. Then, $L_{ij} \cup R_{ij} = V(T_{ij}) = [V_{b1} \cap V(T_{ij})] \cup [V_{b2} \cap V(T_{ij})] \cup [V_{b3} \cap V(T_{ij})]$ is a 3-biclique partition of T_{ij}.*

Proof. From Lemma 12, $\{l_0, r_0\}$, $\{l_1, r_1\}$, and $\{l_2, r_2\}$ are always partitioned into three different bicliques in a 3-biclique partition of B. Thus, for T_{ij} in B, the vertices of either its L_{ij} or R_{ij} all belong to three bicliques. By Proposition 10, the bipartite subgraph of B induced by $V(T_{ij})$ is T_{ij}. Therefore, the edges between L_{ij} and R_{ij} must belong to E_{ij} in a 3-biclique partition of B. From the definition of a biclique, the lemma follows. □

3.1.2. Completing the NP-Completeness Proof of 3-PBB. It is easy to see that 3-PBB $\in$ NP because, for a given bipartite graph B, a nondeterministic algorithm need only guess a partition with size 3 of $V(B)$ that partitions $V(B)$ into three groups and check in polynomial time whether the bipartite subgraph induced by each vertex group is a biclique.

Previously, we constructed a bipartite graph $B = (X[B], Y[B], E[B])$ from a variable set $U = \{u_1, u_2, \ldots, u_n\}$ and a clause collection $C = \{c_1, c_2, \ldots, c_m\}$. All that remains to be shown is that there exists a truth assignment for U such that C is satisfied if and only if there exists a 3-biclique partition for B.

($\rightarrow$) Assume that $A : U \rightarrow \{T, F\}$ is a truth assignment that satisfies C. We first assign each vertex of B in three steps and then show that there exists a 3-biclique partition for B.

(1) Let $c_i = (u_x, u_x, u_y) \in C$; then the value of c_i is $(A(u_x), A(u_y), A(u_z)) \in \{(T, F, F), (F, F, T), (F, T, F)\}$. The 3-biclique partitions of T_{i1}, T_{i2}, and T_{i3} are given from the values of $A(u_x)$, $A(u_y)$, and $A(u_z)$, as presented in Table 1. Based on Lemmas 3, 5, and 7, we set each vertex of T_{i1}, T_{i2}, and T_{i3} to "0," "1," or "2" by Formula (1) and Table 1.

(2) We assign a value to each vertex of $V(B_i)$ as follows: if a vertex $v \in B_i$ has the same label with a vertex $w \in T_{ij}$ $(1 \le j \le 3)$, then set v equal to the value of w. As shown in Table 1, a key observation is that vertices with the same label in T_{i1}, T_{i2}, and T_{i3} are assigned an identical value by a 3-biclique partitions of T_{i1}, T_{i2}, or T_{i3} and the true assignment of U. This ensures that each vertex of $V(B_i)$ cannot be assigned different values.

(3) Similarly as step (2), we assign a value to each vertex of $V(B)$ as follows: if a vertex $v \in V(B)$ has the same label with a vertex $w \in B_i$ $(1 \le j \le 3)$, then set v equal to the value of w. Clearly, by the truth assignment, even if a variable occurs in more than one clause of C, the variable has exactly one value; therefore, even if a variable corresponds to more than one vertex in different B_i's, these vertices corresponding to this variable are assigned an identical value, and it is not hard to see that each vertex of $\{l_0, r_0, l_1, r_1, l_2, r_2\}$ has an identical value in different B_i's by Formula (1). In addition, except for u_x, u_y, u_z, l_0, r_0, l_1, r_1, l_2, and r_2, there do not exist other vertices with the same label in different B_i's. It follows that vertices with the same label in different B_i's have an identical value. This ensures that each vertex of $V(B)$ cannot be assigned different values.

Next, to prove that there exists a 3-biclique partition for B, it suffices to show that vertices with an identical value form a biclique of B. In other words, we only need to show that if v and w belong to the left and right vertex sets, respectively, and their values are identical, then $(v, w) \in E[B]$. If v and w belong to the same T_{ij}, and their values are identical, then v and w certainly belong to a biclique, and $(v, w) \in E_{ij} \in E[B]$ must hold. If v and w belong to different T_{ij}'s, then the edge (v, w) must be added in the process of merging T_{ij}'s into B_i or merging B_i's into B; that is, $(v, w) \in E[B]$. Therefore, the vertices of B with an identical value certainly form a biclique of B.

($\leftarrow$) Suppose that $V(B) = V_{b1} \cup V_{b2} \cup V_{b3}$ is a 3-biclique partition of B. Based on Lemma 12, a 3-biclique partition of B always partitions $\{l_0, r_0\}$, $\{l_1, r_1\}$, and $\{l_2, r_2\}$ into three different bicliques. By Formula (1), each vertex of B is set to "0," "1," or "2." We next show that the vertices that correspond to a clause $c_i = \{u_x, u_y, u_z\}$ are assigned $(f(u_x), f(u_y), f(u_z)) \in \{(1, 0, 0), (0, 0, 1), (0, 1, 0)\}$.

Based on Lemma 13, $[V_{b1} \cap V(T_{ij})] \cup [V_{b2} \cap V(T_{ij})] \cup [V_{b3} \cap V(T_{ij})]$ is a 3-biclique partition of T_{ij}. Therefore, we can directly consider obtaining the assignment of u_x, u_y, and u_z from a 3-biclique partition of T_{ij}.

When T_{ij} is T_{i1}, based on Lemma 4, we have $(f(u_x), f(d_{i2}), f(d_{i3})) \in \{(1, 2, 1), (0, 2, 2), (0, 0, 1)\}$. Because d_{i2} of T_{i1} and d_{i2} of T_{i2} are of the same vertex, and d_{i3} of T_{i1} and d_{i3} of T_{i3} are of the same vertex in B, then the assignment of d_{i2} in T_{i1} is the same as that of d_{i2} in T_{i2}, and the assignment of d_{i3} in T_{i1} is the same as that of d_{i3} in T_{i3}. Therefore, the assignments of d_{i2} in T_{i2} and d_{i3} in T_{i3} must satisfy $(f(d_{i2}), f(d_{i3})) \in \{(2, 1), (2, 2), (0, 1)\}$. When T_{ij} is T_{i2} or T_{i3}, based on Lemmas 6 and 8, we have $(f(u_y), f(d_{i2})) \in \{(0, 2), (1, 0)\}$, $(f(u_z), f(d_{i3})) \in \{(1, 2), (0, 1)\}$. Therefore, to ensure that $(f(d_{i2}), f(d_{i3})) \in \{(2, 1), (2, 2), (0, 1)\}$ holds, we must have $(f(u_y), f(u_z)) \in \{(0, 0), (0, 1), (1, 0)\}$ hold. It follows that if there is a 3-biclique partition for B, then $(f(u_x), f(u_y), f(u_z)) \in \{(1, 0, 0), (0, 0, 1), (0, 1, 0)\}$ holds.

Because each variable corresponds to exactly one vertex in B, it is easy to obtain a truth assignment for all the variables: $A : U \rightarrow \{T, F\}$ from the vertex values of B. We merely set $A(u_t) = T$ if the assignment of u_t is $f(u_t) = 1$ in B and set $A(u_t) = F$ if the assignment of u_t is $f(u_t) = 0$ in B. After this assignment is made, an arbitrary clause $c_i = \{u_x, u_y, u_z\}$ of an MO3 instance is set to $(A(u_x), A(u_y), A(u_z)) \in \{(T, F, F), (F, F, T), (F, T, F)\}$, which satisfies the clause collection C of the MO3 instance.

3.2. The NP-Completeness of k-PBB (k > 3). To prove the NP-completeness of k-PBB for any $k > 3$, we provide a reduction from 3-PBB as follows.

Theorem 14. *k-PBB ($k > 3$) is NP-complete, where k is a constant positive integer.*

Proof. It is easy to see that k-PBB $\in$ NP because a nondeterministic algorithm need only guess a partition with size k of $V(G)$, which partitions $V(G)$ into k groups for a given bipartite graph G, and check in polynomial time whether the bipartite subgraph that is induced by each vertex group is a biclique.

We provide a reduction from 3-PBB. Given an input instance $G_1 = (X_1, Y_1, E_1)$ of 3-PBB, we form an instance $G_2 = (X_2, Y_2, E_2)$ of k-PBB ($k > 3$) as follows: $X_2 = X_1 \cup \{l[i] \mid 1 \le i \le k-3\}$; $Y_2 = Y_1 \cup \{r[i] \mid 1 \le i \le k-3\}$; $E_2 = E_1 \cup \{(l[i], r[i]) \mid 1 \le i \le k-3\}$. That is, we add $2(k-3)$ vertices and $(k-3)$ independent edges to G_1 for building G_2. Then $G_2 = (X_2, Y_2, E_2)$ becomes an instance of k-PBB ($k > 3$). The subgraph formed by these additional vertices and edges

Table 1: The relationship between the clause $c_i = \{u_x, u_y, u_z\}$ and the vertex values of B_i.

$(A(u_x), A(u_y), A(u_z))$	(T, F, F)	(F, T, F)	(F, F, T)
3-biclique partitions of T_{i1}	$\{l_0, r_0, q_{i1}, q_{i2}, q_{i4}\}$/ $\{l_1, r_1, u_x, d_{i3}\}$/ $\{l_2, r_2, d_{i2}, q_{i3}\}$	$\{l_0, r_0, u_x, d_{i2}\}$/ $\{l_1, r_1, d_{i3}, q_{i1}\}$/ $\{l_2, r_2, q_{i2}, q_{i3}, q_{i4}\}$	$\{l_0, r_0, u_x, q_{i2}\}$/ $\{l_1, r_1, q_{i1}, q_{i3}, q_{i4}\}$/ $\{l_2, r_2, d_{i2}, d_{i3}\}$
3-biclique partitions of T_{i2}	$\{l_0, r_0, u_y, n_{i3}\}$/ $\{l_1, r_1, n_{i1}, n_{i2}\}$/ $\{l_2, r_2, d_{i2}, m_{i1}\}$	$\{l_0, r_0, d_{i2}, n_{i1}\}$/ $\{l_1, r_1, u_y, m_{i1}\}$/ $\{l_2, r_2, n_{i2}, n_{i3}\}$	$\{l_0, r_0, u_y, n_{i3}\}$/ $\{l_1, r_1, n_{i1}, n_{i2}\}$/ $\{l_2, r_2, d_{i2}, m_{i1}\}$
3-biclique partitions of T_{i3}	$\{l_0, r_0, u_z, o_{i1}\}$/ $\{l_1, r_1, d_{i3}, p_{i1}\}$/ $\{l_2, r_2, p_{i2}, p_{i3}\}$	$\{l_0, r_0, u_z, o_{i1}\}$/ $\{l_1, r_1, d_{i3}, p_{i1}\}$/ $\{l_2, r_2, p_{i2}, p_{i3}\}$	$\{l_0, r_0, p_{i1}, p_{i2}\}$/ $\{l_1, r_1, u_z, p_{i3}\}$/ $\{l_2, r_2, d_{i3}, o_{i1}\}$

consists of $k-3$ disjoint bicliques, and each biclique contains only one edge.

We have that there exists a 3-biclique partition for G_1 if and only if there exists a k-biclique partition for G_2 by the observation of G_1 and G_2. The theorem follows. □

By Theorems 2 and 14, we get that Corollary 15 holds.

Corollary 15. *k-PBB is NP-complete for $k \geq 3$, where k is a constant positive integer.*

4. The Complexity of k-SPBM

Next, we discuss the complexity of k-SPBM. We show that k-SPBM is NP-complete for any $k \geq 3$.

Theorem 16. *k-SPBM is NP-complete for an arbitrary $k \geq 3$, where k is a constant positive integer.*

Proof. It is easy to see that k-SPBM belongs to NP, given a binary matrix A, because a nondeterministic algorithm need only guess k submatrices with entries "1" of A and check in polynomial time whether these submatrices are a k-submatrix partition of A.

In what follows, we reduce k-PBB to k-SPBM. Assume that $B = (X, Y, E)$ is an instance of k-PBB, where $X = \{x[1], x[2], \ldots, x[m]\}$, $Y = \{y[1], y[2], \ldots, y[n]\}$. Thus, we construct an $m \times n$ binary matrix $A = [a_{ij}]_{m*n}$, and we assign "0" or "1" to each entry of A by the following:

$$a_{ij} = \begin{cases} 1, & (x[i], y[j]) \in E, \\ 0, & (x[i], y[j]) \notin E. \end{cases} \quad (10)$$

We next show that there exists a k-biclique partition for B if and only if A has a k-submatrix partition.

$(\rightarrow)$ Suppose that $X \cup Y = V_1 \cup V_2 \cup \cdots \cup V_k$ is a k-biclique partition of B. A submatrix A_i of A can be obtained from the vertex set V_i as follows. Let $V_i = X_i \cup Y_i$, and let $X_i = \{x[i_1], \ldots, x[i_p]\}$ and $Y_i = \{y[j_1], \ldots, y[j_q]\}$ be the left and right vertex sets of B, respectively. Then let $R_i = \{i_1, \ldots, i_p\}$, $C_i = \{j_1, \ldots, j_q\}$. Thus, a submatrix $A_i = A[R_i, C_i]$ of A is selected. Note that, because $V_1, V_2, \ldots, V_k$ are a k-biclique partition of B, $R_i \cap R_j = \emptyset$, $C_i \cap C_j = \emptyset$, where $i \neq j$, and $R_1 \cup \cdots \cup R_k = \{1, \ldots, m\}$, $C_1 \cup \cdots \cup C_k = \{1, 2, \ldots, n\}$. Moreover, for $j_s \in R_i$, $j_t \in C_i$. Because $(x[i_s], y[j_t]) \in E$, $a_{j_s j_t} = 1$; that is, each entry of A_i is "1." Thus, $A_1, A_2, \ldots, A_k$ are a k-submatrix partition of A.

$(\leftarrow)$ Assume that $A_1 = A[R_1, C_1], \ldots, A_k = A[R_k, C_k]$ are submatrices of A, where $R_i \cap R_j = \emptyset$ $(i \neq j)$, $C_i \cap C_j = \emptyset$ $(i \neq j)$, $R_1 \cup \cdots \cup R_k = \{1, \ldots, m\}$, $C_1 \cup \cdots \cup C_k = \{1, 2, \ldots, n\}$, and each entry of A_i is "1." Then, for the vertex set $V_i = \{x[i_s], y[j_t] \mid i_s \in R_i, j_t \in C_i\}$ obtained from R_i and C_i, where $1 \leq i \leq k$, the bipartite subgraph of B induced by V_i is a biclique because each entry of A_i is "1." Moreover, as $A_1, \ldots, A_k$ are pairwise row and column exclusive and each row (column) of A occurs in exactly one of these submatrices, $X \cup Y = V_1 \cup V_2 \cdots \cup V_k$ is a 3-biclique partition of B. □

5. Applications

Large binary matrices arise in many applications, for example, market-basket data analysis, text mining, and community detection. In addition, we can transform a real matrix into a binary matrix in biclustering for convenient analysis [11, 21–24]; the same approach can be used for clustering [25–27]. Recently, because of its prevalence and importance, the biclustering problem in binary matrices has been widely applied to many domains [3, 24, 28], such as the following.

(1) Market-basket analysis: this goal aims at finding groups of customers who have similar purchasing preferences toward a subset of products. We are given a binary matrix with rows that correspond to customers and columns that correspond to products. If entry (i, j) of the matrix is "1," then customer i purchased product j. If the entry is "0," then the customer did not purchase that product. Clearly, a submatrix with entries "1" formed by a subset of rows and a subset of columns can reveal that the corresponding customers have similar purchasing preferences [3].

(2) Gene expression data analysis: this analysis searches for groups of genes that have similar expression levels toward a subset of conditions. We are given a binary matrix with rows that correspond to genes and columns that correspond to conditions. If entry (i, j) of the matrix is "1," then gene i was switched on under condition j. If the entry is "0," then the gene was not switched on under the condition. A submatrix with entries "1" formed by a subset of

rows and a subset of columns can reveal that it is highly likely that these genes in the submatrix either perform similar functions or are involved in the same biological process [11].

(3) There are also many other applications, including community detection and text mining.

The model of k-SPBM can be used to analyze data that belong to different domains and can help extract previously unknown interesting patterns of biclusters.

6. Conclusions and Future Work

We have first proved that 3-PBB is NP-complete by reduction from MO3. Moreover, we have proved that k-PBB ($k > 3$) is NP-complete by reduction from 3-PBB, thus proving that k-PBB ($k \geq 3$) is NP-complete. Finally, we have shown that k-SPBM ($k \geq 3$) is NP-complete from the NP-completeness of k-PBB ($k \geq 3$).

Because k-SPBM ($k \geq 3$) is NP-complete, the problem has no polynomial time algorithm. Determining an efficient exact algorithm or an approximation algorithm is important, and it requires further research. We intend to study this problem in the future. Moreover, the complexity of some variants of finding bicliques in bipartite graphs is open, for example, the maximum ±1 edge weight biclique problem [15]. Additionally, we plan to study complexity and algorithms for these problems.

Appendices

A. Proof of Lemma 3

Proof. Obviously, for a 3-biclique partition of T_{i1}, q_{i1} and q_{i2} belong to 1 or 2 bicliques. If q_{i1} and q_{i2} belong to 2 bicliques, with $(d_{i2}, q_{i1}) \notin E_{i1}$ and $(d_{i2}, q_{i2}) \notin E_{i1}$, then d_{i2}, q_{i1}, and q_{i2} belong to three different bicliques. Moreover, $(q_{i4}, d_{i2}) \notin E_{i1}$; therefore, either q_{i4} and q_{i1} or q_{i4} and q_{i2} belong to the same biclique. Thus, if there exists a 3-biclique partition for T_{i1}, there are three cases to be considered: (1) q_{i1} and q_{i2} belong to 1 biclique; (2) q_{i1} and q_{i2} belong to 2 bicliques, and q_{i4} and q_{i1} belong to the same biclique; and (3) q_{i1} and q_{i2} belong to 2 bicliques and q_{i4} and q_{i2} belong to the same biclique. Below we discuss the three cases.

(1) In case 1, as shown in Figure 3(b), suppose that $V(T_{i1}) = V_{b1} \cup V_{b2} \cup V_{b3}$ is an arbitrary 3-biclique partition of T_{i1}. Because l_0 is a unique vertex that is adjacent to q_{i1} and q_{i2}, and $(l_1, q_{i2}) \notin E_{i1}$, we can assume without loss of generality that $\{l_0, q_{i1}, q_{i2}\} \subseteq V_{b1}$ and $\{l_1\} \subseteq V_{b2}$. Because $(r_2, l_1) \notin E_{i1}$ and $(r_2, l_0) \notin E_{i1}$, we have $\{r_2\} \subseteq V_{b3}$. Because $(u_x, q_{i1}) \notin E_{i1}$ and $(u_x, r_2) \notin E_{i1}$, and r_1 is a unique vertex that is adjacent to l_1 and u_x, thus, we have $\{l_1, u_x, r_1\} \subseteq V_{b2}$. Because $(l_2, q_{i1}) \notin E_{i1}$ and $(l_2, r_1) \notin E_{i1}$, $(d_{i2}, q_{i1}) \notin E_{i1}$ and $(d_{i2}, r_1) \notin E_{i1}$, and $(q_{i3}, l_0) \notin E_{i1}$ and $(q_{i3}, u_x) \notin E_{i1}$, we have $\{l_2, d_{i2}, r_2, q_{i3}\} \subseteq V_{b3}$. Because $(d_{i3}, q_{i2}) \notin E_{i1}$ and $(d_{i3}, q_{i3}) \notin E_{i1}$, $\{l_1, u_x, d_{i3}, r_1\} \subseteq V_{b2}$, $(r_0, l_2) \notin E_{i1}$ and $(r_0, l_1) \notin E_{i1}$, and $(q_{i4}, u_x) \notin E_{i1}$ and $(q_{i4}, d_{i2}) \notin E_{i1}$, thus, we have $\{l_0, r_0, q_{i4}, q_{i1}, q_{i2}\} \subseteq V_{b1}$. We conclude that, in case 1, each vertex set of $\{l_0, r_0, q_{i4}, q_{i1}, q_{i2}\}$, $\{l_1, u_x, d_{i3}, r_1\}$, and $\{l_2, d_{i2}, r_2, q_{i3}\}$ induces a biclique. It follows that, in case 1, $V(T_{i1}) = \{l_0, r_0, q_{i4}, q_{i1}, q_{i2}\} \cup \{l_1, u_x, d_{i3}, r_1\} \cup \{l_2, d_{i2}, r_2, q_{i3}\}$ is a unique 3-biclique partition of T_{i1}.

(2) In case 2, as shown in Figure 3(c), suppose that $V(T_{i1}) = V_{b1} \cup V_{b2} \cup V_{b3}$ is an arbitrary 3-biclique partition of T_{i1}. Because $\{q_{i4}, q_{i1}\}$, $\{q_{i2}\}$, and $\{d_{i2}\}$ belong to different bicliques, we can assume without loss of generality that $\{q_{i4}, q_{i1}\} \subseteq V_{b1}$, $\{q_{i2}\} \subseteq V_{b2}$, and $\{d_{i2}\} \subseteq V_{b3}$. Because $(d_{i3}, q_{i2}) \notin E_{i1}$, $(d_{i3}, q_{i4}) \notin E_{i1}$, and r_2 is a unique vertex that has edges to d_{i2} and d_{i3}, we have $\{d_{i2}, d_{i3}, r_2\} \subseteq V_{b3}$. Because $(l_1, q_{i2}) \notin E_{i1}$ and $(l_1, r_2) \notin E_{i1}$, and $(u_x, q_{i1}) \notin E_{i1}$ and $(u_x, r_2) \notin E_{i1}$, we have $\{l_1, q_{i4}, q_{i1}\} \subseteq V_{b1}$ and $\{u_x, q_{i2}\} \subseteq V_{b2}$. Because $(q_{i3}, u_x) \notin E_{i1}$ and $(q_{i3}, d_{i3}) \notin E_{i1}$, and $(r_0, l_1) \notin E_{i1}$ and $(r_0, d_{i3}) \notin E_{i1}$, we have $\{l_1, q_{i4}, q_{i1}, q_{i3}\} \subseteq V_{b1}$ and $\{u_x, r_0, q_{i2}\} \subseteq V_{b2}$. Because $(l_0, r_2) \notin E_{i1}$ and $(l_0, q_{i3}) \notin E_{i1}$, and $(l_2, r_0) \notin E_{i1}$ and $(l_2, q_{i1}) \notin E_{i1}$, we have $\{l_0, u_x, r_0, q_{i2}\} \subseteq V_{b2}$ and $\{l_2, d_{i2}, d_{i3}, r_2\} \subseteq V_{b3}$. Because $(r_1, l_0) \notin E_{i1}$ and $(r_1, l_2) \notin E_{i1}$, we have $\{l_1, r_1, q_{i4}, q_{i1}, q_{i3}\} \subseteq V_{b1}$. We conclude that, in case 2, each vertex set of $\{l_0, u_x, r_0, q_{i2}\}$, $\{l_1, r_1, q_{i4}, q_{i1}, q_{i3}\}$, and $\{l_2, d_{i2}, d_{i3}, r_2\}$ induces a biclique. It follows that, in case 2, $V(T_{i1}) = \{l_0, u_x, r_0, q_{i2}\} \cup \{l_1, r_1, q_{i4}, q_{i1}, q_{i3}\} \cup \{l_2, d_{i2}, d_{i3}, r_2\}$ is a unique 3-biclique partition of T_{i1}.

(3) In case 3, as shown in Figure 3(d), suppose that $V(T_{i1}) = V_{b1} \cup V_{b2} \cup V_{b3}$ is an arbitrary 3-biclique partition of T_{i1}. Because $\{q_{i1}\}$, $\{q_{i4}, q_{i2}\}$, and $\{d_{i2}\}$ belong to different bicliques, we can assume without loss of generality that $\{q_{i1}\} \subseteq V_{b1}$, $\{q_{i4}, q_{i2}\} \subseteq V_{b2}$, and $\{d_{i2}\} \subseteq V_{b3}$. Because $(u_x, q_{i1}) \notin E_{i1}$, $(u_x, q_{i4}) \notin E_{i1}$, and r_0 is a unique vertex that is adjacent to u_x and d_{i2}, we have $\{u_x, d_{i2}, r_0\} \subseteq V_{b3}$. Because $(l_1, r_0) \notin E_{i1}$ and $(l_1, q_{i2}) \notin E_{i1}$, $(d_{i3}, q_{i4}) \notin E_{i1}$ and $(d_{i3}, r_0) \notin E_{i1}$, $(r_1, d_{i2}) \notin E_{i1}$ and $\Gamma(r_1) \cap (\Gamma(u_x) \cap \Gamma(q_{i2})) = \emptyset$, and $(l_2, q_{i1}) \notin E_{i1}$ and $(l_2, r_0) \notin E_{i1}$, we have $\{l_1, d_{i3}, r_1, q_{i1}\} \subseteq V_{b1}$ and $\{l_2, q_{i4}, q_{i2}\} \subseteq V_{b2}$. Because $(r_2, u_x) \notin E_{i1}$ and $(r_2, l_1) \notin E_{i1}$, and $(q_{i3}, d_{i3}) \notin E_{i1}$ and $(q_{i3}, u_x) \notin E_{i1}$, we have $\{l_2, r_2, q_{i3}, q_{i4}, q_{i2}\} \subseteq V_{b2}$. Because $(l_0, r_1) \notin E_{i1}$ and $(l_0, r_2) \notin E_{i1}$, we have $\{l_0, u_x, d_{i2}, r_0\} \subseteq V_{b3}$. We conclude that, in case 3, each vertex set of $\{l_0, u_x, d_{i2}, r_0\}$, $\{l_1, d_{i3}, r_1, q_{i1}\}$, and $\{l_2, r_2, q_{i3}, q_{i4}, q_{i2}\}$ induces a biclique. It follows that, in case 3, $V(T_{i1}) = \{l_0, u_x, d_{i2}, r_0\} \cup \{l_1, d_{i3}, r_1, q_{i1}\} \cup \{l_2, r_2, q_{i3}, q_{i4}, q_{i2}\}$ is a unique 3-biclique partition of T_{i1}.

Thus, there exist exactly three 3-biclique partitions for T_{i1}. The lemma follows. □

B. Proof of Lemma 4

Proof. By Lemma 3, there exist exactly three 3-biclique partitions for T_{i1}. Therefore, the lemma follows. □

C. Proof of Lemma 5

Proof. For a 3-biclique partition of T_{i2}, there are two cases to be considered: (1) n_{i1} and n_{i2} belong to the same biclique, and (2) n_{i1} and n_{i2} belong to different bicliques. Below we discuss the two cases.

(1) In case 1, as shown in Figure 4(b), suppose that $V(T_{i2}) = V_{b1} \cup V_{b2} \cup V_{b3}$ is an arbitrary 3-biclique partition of T_{i2}. Because l_1 is a unique vertex that is adjacent to n_{i1} and n_{i2}, and m_{i1} has no edges to n_{i1} and n_{i2}, we can assume without loss of generality that $\{l_1, n_{i1}, n_{i2}\} \subseteq V_{b1}$ and $\{m_{i1}\} \subseteq V_{b2}$. Because $(l_0, n_{i2}) \notin E_{i2}$ and $\Gamma(l_0) \cap \Gamma(m_{i1}) = \emptyset$, $(r_0, m_{i1}) \notin E_{i2}$ and $(r_0, l_1) \notin E_{i2}$, and $(n_{i3}, m_{i1}) \notin E_{i2}$ and $(n_{i3}, l_1) \notin E_{i2}$, we have $\{l_0, r_0, n_{i3}\} \subseteq V_{b3}$. Because $(d_{i2}, n_{i2}) \notin E_{i2}$ and $(d_{i2}, n_{i3}) \notin E_{i2}$, $(l_2, n_{i1}) \notin E_{i2}$ and $(l_2, r_0) \notin E_{i2}$, and $(r_2, l_1) \notin E_{i2}$ and $(r_2, l_0) \notin E_{i2}$, we have $\{l_2, d_{i2}, m_{i1}, r_2\} \subseteq V_{b2}$. Because $(r_1, l_0) \notin E_{i2}$ and $(r_1, l_2) \notin E_{i2}$, we have $\{l_1, r_1, n_{i1}, n_{i2}\} \subseteq V_{b1}$. Because $(u_y, n_{i1}) \notin E_{i2}$ and $(u_y, r_2) \notin E_{i2}$, we have $\{l_0, u_y, r_0, n_{i3}\} \subseteq V_{b3}$. We conclude that, in case 1, each vertex set of $\{l_1, r_1, n_{i1}, n_{i2}\}$, $\{l_0, u_y, r_0, n_{i3}\}$, and $\{l_2, d_{i2}, m_{i1}, r_2\}$ induces a biclique. It follows that, in case 1, $V(T_{i2}) = \{l_1, r_1, n_{i1}, n_{i2}\} \cup \{l_0, u_y, r_0, n_{i3}\} \cup \{l_2, d_{i2}, m_{i1}, r_2\}$ is a unique 3-biclique partition of T_{i2}.

(2) In case 2, as shown in Figure 4(c), suppose that $V(T_{i2}) = V_{b1} \cup V_{b2} \cup V_{b3}$ is an arbitrary 3-biclique partition of T_{i2}. Because u_y and m_{i1} have no edges to n_{i1} and n_{i2}, and r_1 is a unique vertex that is adjacent to u_y and m_{i1}, it follows that $\{u_y, m_{i1}, r_1\}$, $\{n_{i1}\}$, and $\{n_{i2}\}$ must belong to different bicliques. We can assume without loss of generality that $\{u_y, m_{i1}, r_1\} \subseteq V_{b1}$ and $\{n_{i1}\} \subseteq V_{b2}$, $\{n_{i2}\} \subseteq V_{b3}$. Because $(d_{i2}, n_{i2}) \notin E_{i2}$ and $(d_{i2}, r_1) \notin E_{i2}$, $(r_0, m_{i1}) \notin E_{i2}$ and $\Gamma(r_0) \cap (\Gamma(n_{i2}) = \emptyset$, and $(l_0, n_{i2}) \notin E_{i2}$ and $(l_0, r_1) \notin E_{i2}$, we have $\{l_0, d_{i2}, r_0, n_{i1}\} \subseteq V_{b2}$. Because $(n_{i3}, m_{i1}) \notin E_{i2}$ and $(n_{i3}, d_{i2}) \notin E_{i2}$, $(r_2, l_0) \notin E_{i2}$ and $(r_2, u_y) \notin E_{i2}$, and $(l_2, r_1) \notin E_{i2}$ and $(l_2, n_{i1}) \notin E_{i2}$, we have $\{l_2, r_2, n_{i3}, n_{i2}\} \subseteq V_{b3}$. Because $(l_1, r_2) \notin E_{i2}$ and $(l_1, r_0) \notin E_{i2}$, we have $\{l_1, u_y, m_{i1}, r_1\} \subseteq V_{b1}$. We conclude that, in case 1, each vertex set of $\{l_1, u_y, m_{i1}, r_1\}$, $\{l_0, d_{i2}, r_0, n_{i1}\}$, and $\{l_2, r_2, n_{i3}, n_{i2}\}$ induces a biclique. It follows that, in case 2, $V(T_{i2}) = \{l_1, u_y, r_1, m_{i1}\} \cup \{l_0, d_{i2}, r_0, n_{i1}\} \cup \{l_2, r_2, n_{i3}, n_{i2}\}$ is a unique 3-biclique partition of T_{i2}.

Thus, there exist exactly two 3-biclique partitions for T_{i2}. The lemma follows. □

D. Proof of Lemma 6

Proof. By Lemma 5, there exist exactly two 3-biclique partitions for T_{i2}. Therefore, the lemma follows. □

E. Proof of Proposition 9

Proof. Suppose that a bipartite subgraph of B_i induced by $L_{ij} \cup R_{ij} = V(T_{ij})$ is $T_{ij}[B_i] = (L_{ij}, R_{ij}, E'_{ij}[B_i])$. It suffices to prove that $E'_{ij}[B_i] = E_{ij}$. By Formulae (4) and (5), for an edge $(u, v) \in E_2[B_i]$, u and v do not simultaneously belong to T_{ij}. That is, $E'_{ij}[B_i] \cap E_2[B_i] = \emptyset$. Therefore, we need only consider whether the edges in $E_1[B_i]$ can lead to a difference between $E'_{ij}[B_i]$ and E_{ij}. By Formula (3), we have $E_{ij} \sqsubseteq E'_{ij}[B_i]$. For $k \neq j$, we next show that, if any edge of T_{ik} does not belong to T_{ij}, then it cannot become an edge of $T_{ij}[B_i]$. To ensure this result, it suffices to show that the public vertices of T_{ij} and T_{ik} induce isomorphic bipartite subgraphs in T_{ij} and T_{ik}, respectively. In fact, the vertex set $(L_{i1} \cap L_{i2}) \cup (R_{i1} \cap R_{i2}) = \{l_0, l_1, l_2, r_0, r_1, r_2, d_{i2}\}$ induces isomorphic bipartite subgraphs in T_{i1} and T_{i2}; the vertex set $(L_{i1} \cap L_{i3}) \cup (R_{i1} \cap R_{i3}) = \{l_0, l_1, l_2, r_0, r_1, r_2, d_{i3}\}$ induces isomorphic bipartite subgraphs in T_{i1} and T_{i3}; the vertex set $(L_{i2} \cap L_{i3}) \cup (R_{i2} \cap R_{i3}) = \{l_0, l_1, l_2, r_0, r_1, r_2\}$ induces isomorphic bipartite subgraphs in T_{i2} and T_{i3}. Thus, $E'_{ij}[B_i] = E'_{ij}[B_i] \cap E_1[B_i] = E_{ij}$. □

F. Proof of Proposition 10

Proof. Suppose that the bipartite subgraph induced by $V(B_i)$ is $B_i[B] = (X[B_i], Y[B_i], E'_i[B])$; we show that $E'_i[B] = E[B_i]$ as follows. By Formulae (8) and (9), for an edge $(u, v) \in E_2[B]$, u and v do not simultaneously belong to the same B_i. In other words, $E'_i[B] \cap E_2[B] = \emptyset$. Thus, we need only to consider whether an edge in $E_1[B]$ can lead to a difference between $E'_i[B]$ and $E[B_i]$. From Formula (7), we have $E[B_i] \sqsubseteq E'_i[B]$. For $j \neq i$, we next show that, for an arbitrary edge of B_j, if it does not belong to B_i, then it cannot become an edge of $B_i[B]$. To ensure this result, it suffices to show that the public vertices of B_j and B_i induce isomorphic bipartite subgraphs in B_j and B_i, respectively. In fact, if $c_i \cap c_j = \emptyset$, then $(X[B_i] \cap X[B_j]) \cup (Y[B_i] \cap Y[B_j]) = \{l_0, l_1, l_2, r_0, r_1, r_2\}$. Obviously, $\{l_0, l_1, l_2, r_0, r_1, r_2\}$ induces isomorphic bipartite subgraphs in B_i and B_j. If $c_i \cap c_j \neq \emptyset$, then $(X[B_i] \cap X[B_j]) \cup (Y[B_i] \cap Y[B_j]) = (c_i \cap c_j) \cup \{l_0, l_1, l_2, r_0, r_1, r_2\}$. Note that any vertex belonging to $c_i \cap c_j$ is always adjacent to r_0 and r_1 in B_i or B_j. Hence, $(c_i \cap c_j) \cup \{l_0, l_1, l_2, r_0, r_1, r_2\}$ induces isomorphic bipartite subgraphs in B_i and B_j. Therefore, the bipartite subgraph of B induced by $V(B_i)$ is isomorphic to B_i. From Proposition 9, the bipartite subgraph of B induced by $V(T_{ij})$ is isomorphic to T_{ij}. □

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

The authors would like to thank the anonymous reviewers for their helpful suggestions and constructive comments. This paper was supported by (1) the PhD Station Foundation of Chinese Education Department (Grant no. 20090131110009), (2) the National Natural Science Foundation of China (Grants nos. 61070019, 61373079, and 61373079), (3) the Natural Science Foundation of Shandong Province of China (Grant no.

ZR2011FL004), (4) the Science and Technology Development Planning Item of Yantai (Grant no. 2010167), and (5) the Key Project of Chinese Ministry of Education (Grant no. 212101).

References

[1] J. Abello, P. Pardalos, and M. Resende, *Handbook of Massive Data Sets*, vol. 4, Springer, Amsterdam, The Netherlands, 2002.

[2] A. K. Jain, "Data clustering: 50 years beyond K-means," *Pattern Recognition Letters*, vol. 31, no. 8, pp. 651–666, 2010.

[3] S. Busygin, O. Prokopyev, and P. M. Pardalos, "Biclustering in data mining," *Computers & Operations Research*, vol. 35, no. 9, pp. 2964–2987, 2008.

[4] J. Hartigan, "Direct clustering of a data matrix," *Journal of the American Statistical Association*, pp. 123–129, 1972.

[5] B. Mirkin, *Mathematical Classification and Clustering*, vol. 11, Kluwer Academic Publishers, Dordrecht, The Netherlands, 1996.

[6] Y. Cheng and G. M. Church, "Biclustering of expression data," in *Proceedings of the International Conference on Intelligent Systems for Molecular Biology (ISMB '00)*, pp. 93–103, AAAI Press, 2000.

[7] J. Yang, W. Wang, H. Wang, and P. Yu, "δ-clusters: capturing subspace correlation in a large data set," in *Proceedings of the 18th International Conference on Data Engineering*, pp. 517–528, March 2002.

[8] I. S. Dhillon, "Co-clustering documents and words using bipartite spectral graph partitioning," in *Proceedings of the 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 269–274, August 2001.

[9] Q. Sheng, Y. Moreau, and B. De Moor, "Biclustering microarray data by Gibbs sampling," *Bioinformatics*, vol. 19, no. 2, pp. 196–205, 2003.

[10] A. Ben-Dor, B. Chor, R. Karp, and Z. Yakhini, "Discovering local structure in gene expression data: the order-preserving submatrix problem," in *Proceedings of the 6th Annual International Conference on Computational Biology*, pp. 49–57, Washington, DC, USA, April 2002.

[11] S. C. Madeira and A. L. Oliveira, "Biclustering algorithms for biological data analysis: a survey," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 1, no. 1, pp. 24–45, 2004.

[12] A. Tanay, R. Sharan, and R. Shamir, "Biclustering algorithms: a survey," in *Handbook of Computational Molecular Biology*, S. Aluru, Ed., pp. 261–274, Chapman & Hall/CRC, London, UK, 2006.

[13] R. Peeters, "The maximum edge biclique problem is NP-complete," *Discrete Applied Mathematics*, vol. 131, no. 3, pp. 651–654, 2003.

[14] M. Dawande, P. Keskinocak, J. M. Swaminathan, and S. Tayur, "On bipartite and multipartite clique problems," *Journal of Algorithms. Cognition, Informatics and Logic*, vol. 41, no. 2, pp. 388–403, 2001.

[15] N. Amit, *The bicluster graph editing problem [M.S. thesis]*, Tel Aviv University, 2004.

[16] M. Dawande, P. Keskinocak, and S. Tayur, "On the biclique problem in bipartite graphs," Gsia Working Paper, Carnegie Mellon University, 1996.

[17] M. H. Heydari, C. O. Shields Jr., L. Morales, and I. H. Sudborough, "Computing cross associations for attack graphs and other applications," in *Proceedings of the 40th Annual Hawaii International Conference on System Sciences (HICSS '08)*, p. 270b, IEEE Computer Society, Waikoloa, Hawaii, USA, January 2007.

[18] D. Bein, L. Morales, W. Bein, J. C. O. Shields, Z. Meng, and I. H. Sudborough, "Clustering and the biclique partition problem," in *Proceedings of the 41st Annual Hawaii International Conference on System Sciences (HICSS '08)*, p. 475, IEEE Computer Society Press, Waikoloa, Hawaii, USA, 2008.

[19] T. J. Schaefer, "The complexity of satisfiability problems," in *Proceedings of the 10th Annual ACM Symposium on Theory of Computing*, pp. 216–226, ACM, San Diego, Calif, USA, 1978.

[20] M. R. Garey and D. S. Johnson, *Computers and Intractability. A Guide to the Theory of NP-Completeness*, W. H. Freeman, San Francisco, Calif, USA, 1979.

[21] A. Prelić, S. Bleuler, P. Zimmermann et al., "A systematic comparison and evaluation of biclustering methods for gene expression data," *Bioinformatics*, vol. 22, no. 9, pp. 1122–1129, 2006.

[22] A. Tanay, R. Sharan, and R. Shamir, "Discovering statistically significant biclusters in gene expression data," *Bioinformatics*, vol. 18, supplement 1, pp. S136–S144, 2002.

[23] L. Wang, Y. Lin, and X. Liu, "Approximation algorithms for biclustering problems," *SIAM Journal on Computing*, vol. 38, no. 4, pp. 1504–1518, 2008.

[24] Z. Y. Zhang, T. Li, C. Ding, X. W. Ren, and X. S. Zhang, "Binary matrix factorization for analyzing gene expression data," *Data Mining and Knowledge Discovery*, vol. 20, no. 1, pp. 28–52, 2010.

[25] A. Figueroa, J. Borneman, and T. Jiang, "Clustering binary fingerprint vectors with missing values for DNA array data analysis," *Journal of Computational Biology*, vol. 11, no. 5, pp. 887–901, 2004.

[26] R. Gelbard, O. Goldman, and I. Spiegler, "Investigating diversity of clustering methods: an empirical comparison," *Data and Knowledge Engineering*, vol. 63, no. 1, pp. 155–166, 2007.

[27] P. Q. Liu, D. M. Zhu, Q. S. Xie, H. Fan, and S. H. Ma, "Complexity and improved heuristic algorithms for binary fingerprints clustering," *Journal of Software*, vol. 19, no. 3, pp. 500–510, 2008.

[28] D. Chakrabarti, D. S. Modha, S. Papadimitriou, and C. Faloutsos, "Fully automatic cross-associations," in *Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 79–88, ACM, August 2004.