

Algorithmic and Complexity Issues of Three Clustering Methods in Microarray Data Analysis¹

Jinsong Tan,² Kok Seng Chua,³ Louxin Zhang,² and Song Zhu⁴

Abstract. The complexity, approximation and algorithmic issues of several clustering problems are studied. These non-traditional clustering problems arise from recent studies in microarray data analysis. We prove the following results. (1) Two variants of the Order-Preserving Submatrix problem are NP-hard. There are polynomial-time algorithms for the Order-Preserving Submatrix problem when the condition or gene sets are fixed. (2) Three variants of the Smooth Clustering problem are NP-hard. The Smooth Subset problem is approximable with ratio 0.5, but it cannot be approximable with ratio $0.5 + \delta$ for any $\delta > 0$ unless $NP = P$. (3) The inferring plaid model problem is NP-hard.

Key Words. Microarray data analysis, NP-hardness, Approximation, Polynomial-time algorithm, Plaid model, Smooth clustering, Order-preserving submatrix.

1. Introduction. Clustering analysis is a vital step in microarray experiment. It gives a readout of the distinct patterns of genes switched on or off in a cell and hence gives researchers a comprehensive snapshot of the cellular dynamics in a condition (such as tissues, environments) (e.g. [1], [9] and [20]). The analysis is divided into two steps for revealing common patterns of gene expressions across different conditions. The first step is to arrange gene-expression values into a matrix, in which the rows represent genes, the columns represent conditions and hence each entry is a measure of the expression strength of a gene in a condition. Based on this matrix, we may treat each gene as a vector (or point) in an n -dimensional metric space, where n is the number of conditions. The genes are then clustered into groups by a method that measures the distances between their corresponding vectors. Clustering analysis can also group conditions that show similar patterns of genome-wide gene expressions.

Traditional methods include k -means, self-organizing maps [20] and hierarchical clustering [9]. Many new methods have also been proposed for the following reasons. First, the traditional methods are best suited to determining relationships among a small number of variables, rather than deriving expression patterns involving thousands of genes. Secondly, microarray experiments have relatively low sensitivity. When rare diseases are studied, there are not enough samples and inevitably there are gene-expression data

¹ This work was partially supported by Grant BMRC01/1/21/19/140.

² Department of Mathematics, National University of Singapore, Singapore 117543. {matzlx, mattjs}@nus.edu.sg.

³ Institute of High Performance Computing, Singapore 117528. chuaks@ihpc.a-star.edu.sg.

⁴ ST Electronics (Satcom & Sensor Systems) Pte. Ltd., Singapore 609602. zhusong@agilis.stengg.com. His work was done in Kent Ridge Digital Labs.

missing data. For instance, in the gene-expression data with 4026 genes and 96 conditions used in [1], there are 47,639 missing values, which is about 12% of the total value.

Here, we study the complexity and algorithmic issues of non-traditional clustering methods that were recently proposed in [4], [11], [16] and [24]. Motivated by the fact that a subset of genes co-express under some but not all conditions, the authors of these works focused on finding local expression patterns on a subset of genes and/or experimental conditions.

In [4] Ben-Dor et al. studied the problem of identifying order-preserving submatrices, in which all genes co-express in the same magnitude under the conditions. In this paper we prove the NP-hardness of two versions of the order-preserving submatrix problem and present a quadratic-time algorithm for two practical subcases of the problem. We also validate the proposed algorithms on real data sets.

Based on the so-called smooth score, Zhang and Zhu proposed a clustering method aimed at overcoming data errors such as data missing [21] and data inconsistency [6] in the stage of clustering analysis. The smooth score is not defined as a pairwise dissimilarity measure like Euclidean distance; instead, it measures the deviation of the expression level of a gene from the average expression level of all concerned genes under a condition (see formula (1) for details). In [24] Zhang and Zhu proposed efficient greedy algorithms for the Smooth Clustering problem: given a set of conditions, find a largest cluster of genes with its smooth score below a threshold under the given conditions. They also looked for a largest smooth “bicluster”, grouping genes and conditions simultaneously as proposed in [7]. Here, we study the approximation issue of a variant of the Smooth Clustering problem. The Smooth Clustering problem is similar to the tiling problem with rectangles [5].

Finally, in their paper [16], Lazzeroni and Owen introduced the so-called plaid model for an exploratory analysis of microarray data in the statistical approach. The plaid model seeks the decomposition of a gene-expression matrix into submatrices with uniform entries. This is a very general statistical model. The decomposition methods related to it include singular-value decomposition, semidiscrete decomposition [14] and non-negative matrix factorization [17]. Here, we show that inferring an optimal plaid model is NP-hard. This answers an open problem posed in [16].

For basic notations and knowledge on NP-hardness and approximation algorithms, the reader is referred to [2], [10], [13] and [22].

2. The Order-Preserving Submatrix Problem. In the rest of this paper, we use $A = (a_{ij})$ to denote a gene-expression matrix with gene set X and condition set Y , in which a_{ij} denotes the expression value of the i th gene in the j th condition. We use $|X|$ and $|Y|$ to denote the numbers of genes in X and conditions in Y , respectively.

2.1. The Problem. Each gene (in the matrix $A = (a_{ij})$) induces an ordering of all the conditions in terms of its expression values. Two genes in rows i and j induce the same linear ordering if for any distinct $k, k' \in Y$, and $a_{ik} \neq a_{ik'}$, $a_{ik} - a_{ik'}$ has the same sign as $a_{jk} - a_{jk'}$.

An *order-preserving submatrix* of A corresponds to a subset X' of genes and a subset Y' of conditions such that, within conditions in Y' , the expression levels of all the genes

in X' have the same linear ordering. The problem of identifying a large order-preserving submatrix is formally defined as [4]:

Order-Preserving Submatrix (OPSM)

Instance: A gene-expression matrix $A = (a_{ij})$ with gene set X and condition set Y .

Question: Find an order-preserving submatrix $A(I, J)$, $I \subseteq X$ and $J \subseteq Y$, that maximizes $\min\{|I|, |J|\}$.

2.2. *NP-Completeness.* We first prove the following variant of the OPSM problem is also NP-complete:

Order-Preserving Submatrix II

Instance: A gene-expression matrix $A = (a_{ij})$ with gene set X and condition set Y , and a positive integer k .

Question: Find an OPSM $A(I, J)$ of k entries, where $I \subseteq X$ and $J \subseteq Y$.

THEOREM 2.1. *The Order-Preserving Submatrix II problem is NP-complete.*

PROOF. The NP-completeness proof is through a reduction from the Maximum Edge Biclique problem:

Instance: A bipartite graph $G = (V, E)$ (without loss of generality, it is assumed that no vertices in V are of degree 0), and positive integer $k \leq |E|$.

Question: Does G contain a complete bipartite subgraph with at least k edges?

This was proved to be NP-complete recently [19]. Since the reduction from the Balanced Complete Bipartite Subgraph to the OPSM problem in [4] cannot be applied here, we refine the reduction as follows.

Given a bipartite graph $G = (V_1 \cup V_2, E)$, we construct a $|V_1| \times (2|V_1||V_2|)$ matrix $A = (a_{ij})$, where $a_{ij} = j$ if $j - 1 \equiv j' \pmod{|V_2|}$ and j' satisfies $(v_i, w_{j'+1}) \in E$, $v_i \in V_1$, $w_{j'+1} \in V_2$, and $a_{ij} = -1$ otherwise. (Here we number both the rows in A and vertices in G from 1 onwards.) By definition, A consists of $2|V_1|$ identical blocks. Finally, we add an extra column to A consisting of -1 entries. Thus, the resulting matrix A has $|V_1|$ rows and $2|V_1||V_2| + 1$ columns. The reduction follows from the fact that G contains a complete bipartite subgraph with at least k edges if and only if A contains an order-preserving submatrix with $2|V_1|k$ entries. The rest of the proof focuses on the proof of this fact.

The “only if” direction follows by construction. If G contains a complete bipartite subgraph $G' = (V'_1 \cup V'_2, E')$ with k edges. Let $V'_1 = \{v_{i_1}, v_{i_2}, \dots, v_{i_s}\}$ and $V'_2 = \{w_{j_1}, w_{j_2}, \dots, w_{j_t}\}$. Then $st = k$. The submatrix in s rows i_1, i_2, \dots, i_s and $2|V_1|t$ columns $m|V_2| + j_1, m|V_2| + j_2, \dots, m|V_2| + j_t$ ($m = 0, 1, \dots, 2|V_1| - 1$) is an order-preserving submatrix with $2|V_1|k$ entries.

To see the “if” direction, we assume that there exists an order-preserving submatrix B with $2|V_1|k$ entries. Note that at most one column of B can contain a “ -1 ” entry since, otherwise, we contradict the order-preserving property. Thus, we may assume B

contains the $(2|V_1||V_2| + 1)$ th entry if it has one column with a “-1” entry. For each $0 \leq m \leq 2|V_1| - 1$, set B_m as the submatrix of B induced in the columns $m|V_2| + 1, m|V_2| + 2, \dots, (1+m)|V_2|$. The largest B_m must contain at least k entries. Otherwise, B contains at most $2|V_1|(k-1) + r(B) \leq 2|V_1|(k-1) + |V_1| < 2|V_1|k$ entries, where the term $r(B)$, the number of rows in B , is due to the fact that B may contain the last column with -1 entries. This contradicts the assumption. It is easy to see that the largest B_m corresponds to a complete bipartite subgraph with at least k entries in A . \square

We then consider the following problem that is closely related to the OPSM problem. It arises from finding genetic features for binary classification.

Maximum Differential Gene Subset

Instance: A gene expression matrix $A = (a_{ij})$ with gene set X and condition set Y , and two positive integers $k \leq |X|$ and $s \leq |Y|/2$.

Question: Are there a gene subset $X' \subseteq X$ and two disjoint condition subsets $Y', Y'' \subset Y$ such that $|X'| = k$, $|Y'| = |Y''| = s$ and such that, for each $i \in X', j' \in Y', j'' \in Y''$, $a_{ij'} < a_{ij''}$?

THEOREM 2.2. *The Maximum Differential Gene Subset problem is NP-complete.*

PROOF. We prove the NP-completeness by a reduction from the Balanced Complete Bipartite Subgraph (BCBS) problem. The BCBS problem is, given a bipartite graph $G = (V_1 \cup V_2, E)$ and a positive integer k , to find two disjoint subsets $V'_1 \subseteq V_1, V'_2 \subseteq V_2$ such that $|V'_1| = |V'_2| = k$ and such that $v_1 \in V'_1$ and $v_2 \in V'_2$ implies that $(v_1, v_2) \in E$. Such a problem is NP-complete (listed as GT24 in [10]).

Given an instance bipartite graph $G = (V_1 \cup V_2, E)$ of the BCBS problem, we construct a $|V_1| \times (2|V_2|)$ matrix $A = (a_{ij})$, where $a_{ij} = 1$ if $(u_i, v_j) \in E, i \leq |V_1|$ and $j \leq |V_2|$ and 0 otherwise. The reduction is derived from the following fact.

FACT. *G contains a balanced complete bipartite subgraph with $2k$ vertices if and only if there are a row subset I and two disjoint subsets $J \subseteq \{1, 2, \dots, |V_2|\}$ and $J' \subseteq \{|V_2| + 1, |V_2| + 2, \dots, 2|V_2|\}$ such that $|I| = |J| = |J'| = k$ and for each $i \in I, j \in J$ and $j' \in J', a_{ij} > a_{ij'}$.*

PROOF. If G contains a balanced complete bipartite subgraph G' with $2k$ vertices, then, by construction, there exists a submatrix $A'_{k \times k} = (1)$ in the first $|V_2|$ columns whose rows and columns correspond to the vertices in V_1 and V_2 , respectively. Let the subsets of rows and columns in A' be I and J , respectively. Again, by definition, the submatrix in rows I and columns $|V_2| + 1, \dots, |V_2| + k$ is a zero submatrix. Therefore, I, J and $J' = \{|V_2| + 1, \dots, |V_2| + k\}$ has the desired property. \square

On the other hand, if row subset I and column subsets J and J' satisfy the property that $|I| = |J| = |J'| = k$ and $a_{ij} > a_{ij'}$ for any $i \in I, j \in J, j' \in J'$, then, the submatrix $A_{I \times J} = (1)$ and $A_{I \times J'} = (0)$. By definition, $A_{I \times J}$ corresponds to a balanced complete bipartite subgraph with $2k$ vertices. \square

2.3. *Efficient Algorithms for Special Cases.* Since the OPSM problem is NP-hard, it is unlikely that it is polynomial-time solvable. In this subsection we present an efficient algorithm for two practical cases of this problem, which leads to a feasible approach for microarray data analysis.

THEOREM 2.3. *For a given gene-expression matrix $A = (a_{ij})$ with gene set X and condition set Y , the following two variants of the OPSM problem are linear-time and quadratic-time solvable, respectively.*

- (i) *Given a subset $J \subseteq Y$, find a largest subset $I \subseteq X$ such that $A(I, J)$ is order-preserving.*
- (ii) *Given a subset $I \subseteq X$, find a largest subset $J \subseteq Y$ such that $A(I, J)$ is order-preserving.*

PROOF. (i) The idea of the proof in this case is simple. Each gene induces a linear ordering on the condition subset J . Note that $A(I, J)$ is order-preserving if and only if all the genes in I induce the same ordering. Hence, we can sort the orderings induced by all the genes in X and take the largest subset of genes that incur the same ordering. With radix sort [8], this algorithm can be implemented in $O(|X|)$ time since the size of J is fixed.

(ii) It is less obvious how to find a largest subset $J \subseteq Y$, given $I \subseteq X$, such that $A(I, J)$ is order-preserving in polynomial time. Here, we reduce it to the problem of finding a longest path in acyclic graphs. Given a gene-expression matrix $A = (a_{ij})$ with gene set X and condition set Y and a gene subset $I \subseteq X$, we define a directed graph $D_A = (V_A, E_A)$, where V_A contains $|Y|$ vertices each corresponding to a condition in Y , and there is an arc $(u, v) \in E_A$ from u to v if their corresponding conditions j_u and j_v satisfy that, for any $i \in I$, $a_{ij_u} < a_{ij_v}$. Obviously, D_A is acyclic and can be constructed in quadratic time. Furthermore, for any condition subset J , $A(I, J)$ is order-preserving if and only if the vertices corresponding to conditions in J form a directed path in D_A . This implies that we only need to find a longest path in the directed acyclic graph D_A , which is solvable in linear time $O(|V_A| + |E_A|)$ (see for example [15]). \square

REMARKS. In contrast, the Maximum Differential Gene Subset problem still remains NP-hard even when the gene set is given. The proof is via the following reduction from the BCBS problem. Given a bipartite graph $G = (V_1 \cup V_2, E)$ such that $E \subseteq V_1 \times V_2$, we define a $(|V_1| + 1) \times (|V_2| + |V_1|)$ matrix $A = (a_{ij})$ as

$$a_{ij} = \begin{cases} 1, & i \neq j \text{ and } j \leq |V_1|, \\ 2, & 1 \leq i = j \leq |V_1|, \text{ or } 1 \leq i \leq |V_1|, \quad j > |V_1| \text{ and } (i, j - |V_1|) \notin E, \\ 3, & \text{otherwise.} \end{cases}$$

Then it is not hard to show that there are two disjoint k -column subsets C and C' such that, for any $1 \leq i \leq |V_1| + 1$, $a_{ij} < a_{ij'}$ for any $j \in C$, $j' \in C'$ if and only if the graph G has a balanced complete bipartite subgraph of $2k$ vertices.

2.4. *Validation Experimental Test.* Most of the microarray data usually contain about 10–30 conditions. Given an integer $k \leq 20$, there are only about 1 million different

k -condition subsets. Therefore, by applying Theorem 2.3 on each such condition subset, we are able to find a k -condition subset $J \subseteq Y$ and $I \subseteq X$ such that $A(I, J)$ is order-preserving and $|I|$ is maximal. We implemented a program based on the first algorithm in Theorem 2.3 and validated it through a real microarray dataset.

On an input expression matrix A with gene set X and condition set Y , for each value $k \leq |Y|$, the program identifies all the largest order-preserving submatrices induced by some k -condition subset by enumerating all the k -condition subsets. Then the statistical significance of each obtained submatrix is evaluated for acceptance.

We tested the program on the breast tumor dataset reported in [12] on a Linux machine with 2.4 GHz Pentium 4 CPU. The dataset consists of 3226 genes and 22 conditions. We compared our program with the original OPSM program in [4] and another program in [18]. We found that our algorithm found not only all the biologically meaningful clusters reported in [4] and [18], but also many more other statistically significant clusters. The testing details can be obtained from the authors.

3. The Smooth Clustering Problems

3.1. *Definitions.* Any subsets $I \subseteq X$ and $J \subseteq Y$ specify a submatrix $A(I, J)$. We associate it with a smooth score

$$(1) \quad s(I, J) = \max_{j \in J} \left(\max_{i \in I} a_{ij} - \frac{1}{|I|} \sum_{k \in I} a_{kj} \right),$$

where $(1/|I|) \sum_{k \in I} a_{kj}$ denotes the average expression value of a gene in I under condition j . The smooth score $s(I, J)$ is actually a refinement of the L_∞ -distance $d_\infty(\cdot, \cdot)$, a popular metric in functional analysis. Recall that, for any two n -dimensional vectors $\mathbf{x} = (x_i)$ and $\mathbf{y} = (y_i)$, $d_\infty(\mathbf{x}, \mathbf{y}) = \max_i |x_i - y_i|$. If a gene-expression level is considered as a function with a variable condition, then the clustering process aims to classify genes into groups, each containing genes with similar expression functions. Thus, the smooth score was proposed for gene-expression analysis in [24]. If $A(I, J)$ has the smooth score $s(I, J)$, then, for any rows v and v' in $A(I, J)$, $d_\infty(v, v') \leq 2s(I, J)$.

Given a small number $\varepsilon > 0$, $A(I, J)$ is an ε -smooth cluster if $s(I, J) \leq \varepsilon$. We formulate the following clustering problem:

Smooth Clustering Problem [24]

Instance: A gene-expression matrix $A = (a_{ij})$ with gene set X and condition set Y , a subset $J \subseteq Y$ and a number $\varepsilon > 0$.

Question: Find a largest subset $I \subseteq X$ such that $A(I, J)$ is an ε -smooth cluster.

To facilitate the study of genes with multiple functions that may or may not be co-active under all conditions, we also seek two-sided clusters as papers [7] and [16]. Thus, the following problem was also studied in [24]:

Smooth Biclustering Problem

Instance: A gene-expression matrix A with gene set X and condition set Y , and a number $\varepsilon > 0$.

Question: Find an ε -smooth submatrix $A(I, J)$, $I \subseteq X$ and $J \subseteq Y$, that maximizes $\min\{|I|, |J|\}$.

3.2. *NP-Hardness Results.* In this subsection we present three NP-hardness results that were also reported in [23].

THEOREM 3.1. *The Smooth Clustering problem is NP-hard.*

PROOF. We prove the theorem by a reduction from the INDEPENDENT SET problem, which is a basic NP-complete problem [10]. Recall that the INDEPENDENT SET problem is, given a graph $G = (V, E)$ and an integer $k > 0$, to find whether G contains a subset of $V' \subseteq V$ such that no two vertices in V' are adjacent and such that $|V'| \geq k$.

Given an instance (G, k) of the INDEPENDENT SET problem, we construct an instance of the Smooth Clustering problem as follows. For simplicity, we assume that G has vertex set $V = \{1, 2, \dots, n\}$ and edge set $E = \{e_1, e_2, \dots, e_m\}$. We define a matrix $A_{n \times m} = (a_{ij})$ as

$$a_{ij} = \begin{cases} 0 & \text{if } i \text{ is not an endpoint of } e_j, \\ -1 & \text{if } e_j = (i, i'), \quad i < i', \\ 1 & \text{if } e_j = (i', i), \quad i' < i. \end{cases}$$

Note that each column of A corresponds to an edge and has exactly two non-zero entries -1 and 1 . Let $J = \{1, 2, \dots, m\}$ and $\varepsilon = 1 - 1/n$.

Let $V' = \{i_1, i_2, \dots, i_k\}$. For any column j of matrix A , if its corresponding edge e_j has two endpoints in V' , then it follows that $\sum_{i \in V'} a_{ij} = 0$, and $\max_{i \in V'} |a_{ij} - (1/k) \sum_{l \in V'} a_{lj}| = 1$; Similarly, if e_j has only one endpoint in V' , $\max_{i \in V'} |a_{ij} - (1/k) \sum_{l \in V'} a_{lj}| = 1 - (1/k)$; and if e_j has no endpoint in V' , then $\max_{i \in V'} |a_{ij} - (1/k) \sum_{l \in V'} a_{lj}| = 0$. This concludes that V' is an independent set of G if and only if V' is a feasible solution to the instance $(A_{n \times m}, J, \varepsilon)$ of the Smooth Clustering problem. Hence, if we can find an optimal solution to $(A_{n \times m}, J, \varepsilon)$, we can easily decide whether the graph G has an independent set of size k or not. This proves that the Smooth Clustering problem is NP-hard. \square

Next, we consider a variant of the Smooth Clustering problem, which is of interest itself.

Square Smooth Clustering Problem

Instance: A gene-expression matrix $A = (a_{ij})$ with gene set X and condition set Y , a subset $J \subseteq Y$ and a number $\varepsilon > 0$.

Question: Find a largest subset $I \subseteq X$ such that, for every $j \in J$,

$$\sum_{i \in I} \left(a_{ij} - \frac{1}{|I|} \sum_{l \in I} a_{lj} \right)^2 \leq \varepsilon.$$

THEOREM 3.2. *The Square Smooth Clustering problem is NP-hard.*

PROOF. Again, we prove the NP-hardness via a reduction from the INDEPENDENT SET problem. Given an instance (G, k) of the INDEPENDENT SET problem, we construct an instance (A, ε) of the Square Smooth Clustering problem as follows. We let $A = (a_{ij})$ be the adjacency matrix of the graph G , where each column of A corresponds to an edge and each row to a vertex. By definition, a_{ij} is 1 if the i th vertex is an end vertex of the j th column and it is 0 otherwise. Since each edge has two endpoints, each column contains exactly two entries of value 1 (called 1-entries). Finally, we set $\varepsilon = 1$.

Assume $V' = \{v_{i_1}, v_{i_2}, \dots, v_{i_k}\}$ is a subset of vertices and its index subset $I' = \{i_1, i_2, \dots, i_k\}$. The reduction is derived from that, if $|V'| = k > 4$, then V' is an independent set if and only if

$$(2) \quad \sum_{i \in I'} \left(a_{ij} - \frac{1}{|I'|} \sum_{l \in I'} a_{lj} \right)^2 \leq 1$$

for every column j in A .

Now we prove the above claim. If V' is an independent set, then there is at most one 1-entry among the a_{ij} 's, $i \in I'$, for every column j . Therefore, for every j , $\sum_{i \in I'} (a_{ij} - (1/|I'|) \sum_{l \in I'} a_{lj})^2 = 0$ if all the a_{ij} 's ($i \in I'$) are zero. Otherwise, $(1/|I'|) \sum_{l \in I'} a_{lj} = 1/k$ since $|I'| = k$ and there is one 1-entry among the a_{ij} 's ($i \in I'$); furthermore,

$$\sum_{i \in I'} \left(a_{ij} - \frac{1}{|I'|} \sum_{l \in I'} a_{lj} \right)^2 = \left(1 - \frac{1}{k} \right)^2 + (k-1) \cdot \left(0 - \frac{1}{k} \right)^2 = 1 - \frac{1}{k} < 1.$$

Conversely, if (2) is true for I' , then there is at most one 1-entry among the a_{ij} 's ($i \in I'$) for every column j . Assume the fact is not true for some column j' . Since the j' th column has exactly two 1's, all these two 1-entries are among the $a_{ij'}$'s, $i \in I'$. Hence, since $|I'| = k$, $(1/|I'|) \sum_{l \in I'} a_{lj'} = 2/k$; since $k > 4$,

$$\sum_{i \in I'} \left(a_{ij'} - \frac{1}{|I'|} \sum_{l \in I'} a_{lj'} \right)^2 = 2 \left(1 - \frac{2}{k} \right)^2 + (k-2) \left(0 - \frac{2}{k} \right)^2 = \frac{2(k-2)}{k} > 1.$$

This contradicts (2). Therefore, there is at most one 1-entry among the a_{ij} 's ($i \in I'$) for every column j and hence V' is an independent set.

We have proved that V' is an independent set of G if and only if its index subset I' is a solution to the instance $(A, 1)$ of the Square Smooth Clustering problem assuming $|V'| = |I'| > 4$. This implies that if we can find a solution of size k for $(A, 1)$, we can easily find an independent set of the same size in G . Since the INDEPENDENT SET problem is NP-complete, the Square Smooth Clustering problem is NP-hard. \square

Finally, the Smooth Biclustering problem can be considered as a generalization of finding a largest balanced complete bipartite subgraph of a bipartite graph. Thus it is also NP-hard.

THEOREM 3.3. *The Smooth Biclustering problem is NP-hard.*

PROOF. The problem is equivalent to finding a largest ε -smooth square submatrix $A(I, J)$, i.e. $|I| = |J|$. We proved the theorem by using a reduction from the BCBS problem. Recall that this problem is, given a bipartite graph $G = (V, E)$ and a positive integer $k \leq |V|$, to find two disjoint subsets $V_1, V_2 \subseteq V$ such that $|V_1| = |V_2| = k$ and such that $v_1 \in V_1$ and $v_2 \in V_2$ imply that $(v_1, v_2) \in E$.

Given a bipartite graph $G = (V, E)$, we construct an instance of the Smooth Biclustering problem as follows. Without loss of generality, we may assume that $V = \{1, 2, \dots, n\}$, where $n = |V|$. We define an $n \times n$ matrix $A = (a_{ij})$ by assigning $a_{ij} = a_{ji} = 0$ if $(i, j) \in E$, and $a_{ij} = i$ and $a_{ji} = j$ otherwise. Hence, each row/column of A corresponds to a vertex of the graph. Let $\varepsilon < \frac{1}{2}$. Then we claim that, for any subsets $I, J \subseteq V$ such that $|I| = |J| \geq 2$, the square submatrix $A(I, J)$ is ε -smooth if and only if $I \times J \subseteq E$ and thus the induced subgraph on $I \cup J$ is a balanced complete bipartite subgraph. This implies that the Smooth Biclustering problem is NP-hard.

Now we conclude the proof by proving the claim. Assume that $I \times J \subseteq E$. By definition, for any $i \in I, j \in J, a_{ij} = 0$. Thus, $A(I, J)$ is a zero submatrix and hence ε -smooth. Conversely, if $A(I, J)$ is ε -smooth for $\varepsilon < \frac{1}{2}$, where $|I| = |J|$, it must be a zero matrix. Otherwise, let the j th column have non-zero entries for some $j \in J$. Assume that $a_{lj} = \min_{i \in I} a_{ij}$ and $a_{mj} = \max_{i \in I} a_{ij}$. Since $|I| \geq 2$, by definition, $a_{lj} < a_{mj}$. Since all entries are integers, $|a_{lj} - (1/|I|) \sum_{i \in I} a_{ij}| \geq \frac{1}{2}$, or $|a_{mj} - (1/|I|) \sum_{i \in I} a_{ij}| \geq \frac{1}{2}$. This contradicts the fact that $A(I, J)$ is ε -smooth. Since $A(I, J)$ is a zero submatrix, by definition, I and J induce a balanced complete bipartite subgraph. \square

3.3. Approximation Results. Since the Smooth Clustering problem is NP-hard, it is desirable to develop efficient approximation algorithms for it. However, this task is also difficult in general. In fact, the proof of Theorem 3.1 gives an approximation-ratio-preservation reduction from the INDEPENDENT SET problem (see [13]). Therefore, there is an $\varepsilon > 0$ such that approximating the Smooth Clustering problem within a factor n^ε is NP-hard, where n is the number of rows in the input gene-expression matrix. In the rest of this section we focus on matrices with only one column, where the Smooth Clustering problem is equivalent to

SMOOTH SUBSET PROBLEM. Given a finite set S , a weight $w(s) \geq 0$ for each $s \in S$, and a positive number ε , find a largest ε -smooth subset $S' \subseteq S$, i.e. $|w(s) - (1/|S'|) \sum_{t \in S'} w(t)| \leq \varepsilon$ for every $s \in S'$.

THEOREM 3.4. *Let $k(S, \varepsilon)$ be the size of a largest ε -smooth subset of S for a weighted set S and $\varepsilon > 0$. There is a quadratic-time algorithm that always outputs an ε -smooth subset of size at least $k(S, \varepsilon)/2$ on any input S .*

PROOF. Let (S, ε) be an instance of the Smooth Subset problem. For simplicity, we assume that $S = \{a_1, a_2, \dots, a_n\}$, where $w(a_i) \leq w(a_{i+1})$ for $1 \leq i \leq n-1$. Otherwise, we can sort S in terms of its weight function in polynomial time. Let S' be a largest ε -smooth subset of S . Then, by assumption, $k(S, \varepsilon) = |S'|$. For any $a, b \in S'$, by triangle inequality,

$$|w(a) - w(b)| \leq |w(a) - m| + |w(b) - m| \leq 2\varepsilon,$$

where $m = (1/|S'|) \sum_{x \in S'} w(x)$. This concludes that S' is contained in an interval of length 2ε . Hence,

$$k(S, \varepsilon) = |S'| \leq \max_{1 \leq i \leq n} |\{a \in S \mid w(a) \in [w(a_i), w(a_i) + 2\varepsilon]\}|.$$

Let $X_i = \{a \in S \mid w(a) \in [w(a_i), w(a_i) + \varepsilon]\}$ for $i = 1, 2, \dots, n$. Clearly, each X_i is an ε -smooth subset of S since $|w(a) - (1/|X_i|) \sum_{x \in X_i} w(x)| \leq w(a_i) + \varepsilon - w(a_i) = \varepsilon$ for every $a \in X_i$. We choose the largest subset X over all X_i 's, $1 \leq i \leq n$. Since

$$k(S, \varepsilon) \leq \max_{1 \leq i \leq n} |\{a \in S \mid w(a) \in [w(a_i), w(a_i) + 2\varepsilon]\}| \leq 2|X|,$$

X is an ε -smooth subset of size at least $\frac{1}{2}k(S, \varepsilon)$. Moreover, X can be found in quadratic time. \square

The above theorem gives a simple $\frac{1}{2}$ -approximation algorithm for the Smooth Subset problem. Surprisingly, we can also show there is no approximation algorithm with a ratio better than $\frac{1}{2}$ unless $\text{NP} = \text{P}$. This improves the inapproximability result in [23].

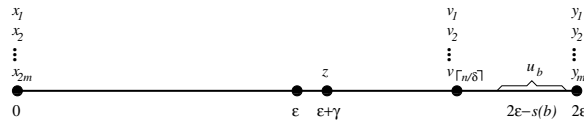
THEOREM 3.5. *Let $k(S, \varepsilon)$ be the size of the largest ε -smooth subsets of S for any weighted set S and $\varepsilon > 0$. For any small constant $\delta > 0$, there is no polynomial-time algorithm that can always output an ε -smooth subset of size at least $(\frac{1}{2} + \delta)k(S, \varepsilon)$ on an input S unless $\text{NP} = \text{P}$.*

PROOF. Let δ be a small positive constant. Suppose \mathcal{A} is a polynomial-time approximation algorithm with approximation factor $\frac{1}{2} + \delta$ for the Smooth Subset problem. We will show that \mathcal{A} can be used to derive a polynomial-time algorithm for the PARTITION problem, which implies $\text{NP} = \text{P}$ since PARTITION is NP-complete [10]. Recall that the PARTITION problem is, given a finite set B and an integer size $s(b) > 0$ for each $b \in B$, to decide if there is a subset $B' \subseteq B$ such that $\sum_{b \in B'} s(b) = \sum_{b \in B - B'} s(b)$.

For a weighted set B as an instance of the PARTITION problem, we set

$$\begin{aligned} \sigma &= \sum_{b \in B} s(b), \\ \gamma &= (\lceil n/\delta \rceil + \frac{1}{2})\sigma, \\ m &= n + \lceil n/\delta \rceil. \end{aligned}$$

We construct an instance (D, ε) of the Smooth Subset problem from B as follows. First, we set $\varepsilon = (2m + 3)\sigma$. The set D contains $2m$ x_i 's of weight 0, m y_i 's of weight 2ε and $\lceil n/\delta \rceil$ v_i 's of weight $2\varepsilon - \sigma$; for each $b \in B$, D contains a unique element u_b of weight $2\varepsilon - s(b) > 0$; in addition, D also contains an element z of weight $\varepsilon + \gamma$. In total, D contains $4m + 1$ elements as illustrated below:



FACT. *If there is a solution to the PARTITION instance B , then D has an ε -smooth subset of size at least $2m + 2\lceil n/\delta \rceil + 1$. Otherwise, any ε -smooth subset of D has size at most $2m + 1$.*

PROOF. Suppose B has a subset B' such that $\sum_{b \in B'} s(b) = \sum_{b \in B - B'} s(b) = \frac{1}{2}\sigma$. Then $D' = \{x_i, y_j \mid i \leq m + \lceil n/\delta \rceil + |B'|, j \leq m\} \cup \{v_k, u_b \mid k \leq \lceil n/\delta \rceil, b \in B'\} \cup \{z\}$ is a desired ε -smooth subset since it contains $2(m + \lceil n/\delta \rceil + |B'|) + 1$ elements and

$$\begin{aligned} \sum_{d \in D'} w(d) &= \sum_{j \leq m} w(y_j) + \sum_{k \leq \lceil n/\delta \rceil} w(v_k) + \sum_{b \in B'} w(u_b) + w(z) \\ &= 2m\varepsilon + (2\varepsilon - \sigma)\lceil n/\delta \rceil + (2\varepsilon|B'| - \frac{1}{2}\sigma) + (\varepsilon + \gamma) \\ &= (2(m + \lceil n/\delta \rceil + |B'|) + 1)\varepsilon + (\gamma - \sigma(\lceil n/\delta \rceil + \frac{1}{2})) \\ &= (2(m + \lceil n/\delta \rceil + |B'|) + 1)\varepsilon, \end{aligned}$$

where $w(d)$ denotes the weight of the element d . This proves the first part of the fact.

If there is no solution to the PARTITION instance B , then, $\sum_{b \in B'} s(b) \neq \frac{1}{2}\sigma$ for any subset $B' \subset B$. Let D'' be a largest ε -smooth subset of D . Recall that, for each $d \in D$, we use $w(d)$ to denote its weight. Let $\mu = (1/|D''|) \sum_{d \in D''} w(d)$. We consider the following three cases.

Case 1: $\mu > \varepsilon$. Then D'' does not contain any of the $2m$ elements of weight 0 and hence $|D''| \leq 2m + 1$.

Case 2: $\mu < \varepsilon$. First, it does not contain any of the y_j 's of weight 2ε . We further show that either it does not contain more than m elements with weight 0 or it does not contain any of the elements $u_b, b \in B$, and $v_k, 1 \leq k \leq \lceil n/\delta \rceil$. This implies that $|D''| \leq 2m + 1$.

Assume $|D''| \geq 2m + 2$ and there are $m + l$ ($l \geq 1$) elements of weight 0 in D'' . Since D'' does not contain any of the elements with weight 2ε and $|D''| \geq 2m + 2$,

$$\begin{aligned} \mu &= \frac{1}{|D''|} \sum_{d \in D''} w(d) \\ &\leq \frac{1}{|D''|} \left[w(z) + \sum_{1 \leq k \leq \lceil n/\delta \rceil} w(v_k) + \sum_{b \in B} w(u_b) \right] \\ &= \frac{1}{|D''|} [(\varepsilon + \gamma) + (2\varepsilon - \sigma)\lceil n/\delta \rceil + (2n\varepsilon - \sigma)] \\ &= \frac{1}{|D''|} [(2m + 1)\varepsilon - \frac{1}{2}\sigma] \\ &\leq \frac{2m + 1}{2m + 2} \varepsilon. \end{aligned}$$

For each $b \in B$, u_b has weight $2\varepsilon - s(b)$. Since

$$(2\varepsilon - s(b)) - \mu \geq \varepsilon - s(b) + \frac{1}{2m + 2} \varepsilon = \varepsilon + \left(\frac{2m + 3}{2m + 2} \sigma - s(b) \right) > \varepsilon,$$

u_b is not in D'' . Similarly, we can show that all the v_k ($1 \leq k \leq \lceil n/\delta \rceil$) are not in D'' .

Case 3: $\mu = \varepsilon$. In this case we have $|D''| \leq 2m$ by proving the fact that D'' does not contain any elements from $\{z, u_b, v_k | b \in B, 1 \leq k \leq \lceil n/\delta \rceil\}$. If the fact is not true, we assume D'' contains l more weight-0 elements x_i than weight- (2ε) elements y_j . Let $D_1 = D'' \cap \{u_b | b \in B\}$ and $D_2 = D'' \cap \{v_i | 1 \leq i \leq \lceil n/\delta \rceil\}$. We consider the following two subcases.

Case 3.1: $z \notin D''$. Then, since $\mu = \varepsilon$, we have

$$\begin{aligned} (l + |D_1| + |D_2|)\varepsilon &= \sum_{u_b \in D_1} w(u_b) + \sum_{v \in D_2} w(v) \\ &= 2\varepsilon(|D_1| + |D_2|) - \sum_{b: u_b \in D_1} s(b) - |D_2|\sigma. \end{aligned}$$

This implies

$$\sum_{b: u_b \in D_1} s(b) + |D_2|\sigma = (|D_1| + |D_2| - l)\varepsilon,$$

a contradiction since the left side is non-zero but smaller than ε .

Case 3.2: $z \in D''$. Similarly, we have

$$\begin{aligned} (1 + l + |D_1| + |D_2|)\varepsilon &= \varepsilon + \gamma + \sum_{u_b \in D_1} w(u_b) + \sum_{v \in D_2} w(v) \\ &= \varepsilon + (\lceil n/\delta \rceil + \frac{1}{2})\sigma + 2\varepsilon(|D_1| + |D_2|) - \sum_{b: u_b \in D_1} s(b) - |D_2|\sigma \end{aligned}$$

or equivalently

$$(|D_1| + |D_2| - l)\varepsilon = (|D_2| - \lceil n/\delta \rceil)\sigma + \left(\sum_{b: u_b \in D_1} s(b) - \frac{1}{2}\sigma \right).$$

This implies that $\lceil n/\delta \rceil = |D_2|$ and $\frac{1}{2}\sigma = \sum_{b: u_b \in D_1} s(b)$, contradicting that there is no solution to the PARTITION instance B . This finishes the proof of the fact. \square

By assumption, \mathcal{A} is a polynomial-time algorithm with approximation factor $\frac{1}{2} + \delta$ for the Smooth Subset problem. Now we apply \mathcal{A} to the instance D . If \mathcal{A} outputs an ε -smooth subset of size at most $2m + 1$, then the largest ε -smooth subset has size at most

$$\frac{2m + 1}{1/2 + \delta} = \frac{2n + 2\lceil n/\delta \rceil + 1}{1/2 + \delta} < 2n + 4\lceil n/\delta \rceil + 1 = 2m + 2\lceil n/\delta \rceil + 1$$

since \mathcal{A} is a $(\frac{1}{2} + \delta)$ -approximation algorithm. Thus, we conclude that there is no solution to the PARTITION instance B by the fact proved above. If \mathcal{A} outputs an ε -smooth subset of size at least $2m + 2$, then, by the fact, there is a solution to the PARTITION instance B . Therefore, we derive a polynomial time algorithm for the PARTITION problem using \mathcal{A} , which implies that $\text{NP} = \text{P}$. \square

4. Inferring Plaid Model Problem

4.1. *The Plaid Model.* With present microarray technology, a gene-expression matrix can contain many thousands of entries. Therefore, even visualization of a microarray data is challenging. One natural way to do this is first to form a color image of the data $A = (a_{ij})$ on an $|X|$ by $|Y|$ grid, with each cell colored according to the value of a_{ij} . Then it proceeds with re-ordering the rows and columns so that similar rows and columns are grouped together and hence an image with blocks of a similar color is formed. For instance, the rows and columns can be re-ordered after running a hierarchical clustering method on genes [9].

An ideal re-ordering of the array would produce an image with K rectangular blocks, each being nearly uniformly colored. Mathematically, this ideal corresponds to the existence of a disjoint K' -partition of genes and a disjoint K'' -partition of conditions such that $K'K'' = K$ and

$$a_{ij} = c_0 + \sum_{k'=1}^{K'} \sum_{k''=1}^{K''} \lambda_{k'k''}(i, j) c_{k'k''},$$

where c_0 is a background color, $c_{k'k''}$ is the color in the block specified by the k' th gene-block $X_{k'}$ and the k'' th condition block $Y_{k''}$, and $\lambda_{k'k''}(i, j)$ is 1 if $i \in X_{k'}$ and $j \in Y_{k''}$, and 0 otherwise.

However, it is more likely that the blocks will overlap in some places in real data. By removing the constraints that K color blocks are disjoint, we obtain the **plaid model** that represents the microarray data as a sum of possible overlapping “constant” layers. Algebraically, the plaid model corresponds to the decomposition of A into K “uniform” matrices $B_k = (b_{ij})$'s that are defined over gene subset X_k and condition subset Y_k such that

$$a_{ij} = \sum_{k=1}^K c'_k(i, j),$$

where $c'_k(i, j) = b_{ij}$ if $i \in X_k$ and $j \in Y_k$, and 0 otherwise. Let ρ_{ik} be 1 if i is in the gene subset X_k and 0 otherwise. To capture biological interests of identifying genes that had identically, though not constantly, co-expressed in a subset of conditions, the model allows each matrix $B_k = (b_{ij})$ to take one of the following forms [16]:

$$\begin{aligned} b_{ij} &= c_k, \\ b_{ij} &= c_k + \alpha_{ik}, \\ b_{ij} &= c_k + \beta_{kj}, \\ b_{ij} &= c_k + \alpha_{ik} + \beta_{jk}, \end{aligned}$$

where if α_{ik} is used, we request that $\sum_i \rho_{ik} \alpha_{ik} = 0$ to avoid overparameterization, with a similar condition on β_{jk} .

The clustering problem under the plaid model is to seek a model that best fits the data, i.e. with the smallest value of

$$\sum_i \sum_j \left[a_{ij} - \sum_{k=1}^K c'_k(i, j) \right]^2.$$

In this paper we study the following decision version of the above clustering problem. For simplicity, we say a matrix is *uniform* if all its entries are identical. Furthermore, we use (b) to denote the uniform matrix whose entries are b .

Plaid Model Fitting

Instance: A gene-expression matrix $A = (a_{ij})$ with gene set X and condition set Y , and a positive integer $K \leq |X||Y|$.

Question: Are there K uniform submatrices $B_k = (b_k)$ ($b_k \geq 0$, $1 \leq k \leq K$) with gene subset X'_k and condition subset Y'_k such that for each i and j , $a_{ij} = \sum_{k=1}^K \lambda_k(i, j)b_k$, where $\lambda_k(i, j)$ is 1 if $i \in X'_k$ and $j \in Y'_k$, and 0 otherwise?

4.2. NP-Completeness

THEOREM 4.1. *The Plaid Model Fitting problem is NP-complete.*

We prove the above theorem via a reduction from the following graph decomposition problem:

Complete Bipartite Subgraph Decomposition

Instance: A weighted bipartite graph $G = (V, E)$ in which each edge has a positive weight, and integer K .

Question: Can G be decomposed into K positively weighted completed bipartite graphs G_i such that all the edges in G_i have same weight, and for each edge in E , its weight in G is equal to the sum of its weights in the G_i 's?

Such a reduction is more or less straightforward since a matrix $A = (a_{ij})$ with a non-negative submatrix gives a unique weighted bipartite graph, in which the edge from the vertex corresponding to row i to the vertex corresponding to column j is assigned weight a_{ij} . To finish the proof of Theorem 4.1, we only need to prove the following lemma.

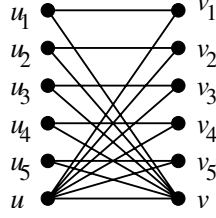
LEMMA 4.1. *The Complete Bipartite Subgraph Decomposition problem is NP-complete.*

PROOF. Obviously, the problem is in NP. We prove it is NP-hard by a reduction from the PARTITION problem. Recall that the PARTITION problem is, given a finite set P and an integer size $s(p) > 0$ for each $p \in P$, to decide if there is a subset $P' \subseteq P$ such that $\sum_{p \in P'} s(p) = \sum_{p \in P - P'} s(p)$.

Given an instance $P = \{p_1, p_2, \dots, p_n\}$ of the PARTITION problem, let $c = \sum_{i=1}^n s(p_i)$. We construct a weighted bipartite graph $G = (V_1 \cup V_2, E)$ with

$$\begin{aligned} V_1 &= \{u_1, u_2, \dots, u_n, u\}, & V_2 &= \{v_1, v_2, \dots, v_n, v\}, \\ E &= \{(u_i, v_i), (u, v), (u, v_i), (u_i, v) \mid 1 \leq i \leq n\} \end{aligned}$$

(as shown for $n = 5$ below)



where the edge (u, v) has weight $c/2$; for each $1 \leq i \leq n$, the edge (u_i, v_i) has weight $s(p_i)$, and both (u, v_i) and (u_i, v) have weight $2^{i-2} \cdot 3c + \sum_{j=1}^i 2^{i-j} s(p_j)$. Let $w(e)$ denote the weight assigned to the edge e . It is easy to see that $w((u, v_i)) = s(p_i) + 2w((u, v_{i-1}))$ and $w((u_i, v)) = s(p_i) + 2w((u_{i-1}, v))$ for each $i \geq 2$. The reason for assigning such a large weight to each “diagonal” edge in the graph is to force that at least one extra complete bipartite graph is required to contain edge (u, v_i) or (u_i, v) in any desired decomposition of G . The reduction is derived from the following fact.

FACT. *If there is a solution to the PARTITION instance P , then the weighted graph G can decompose into $3n$ weighted complete bipartite subgraphs; otherwise, any complete bipartite graph decomposition of G contains at least $3n+1$ complete bipartite subgraphs.*

PROOF. Assume there is a solution to the instance P . For simplicity, we assume that the first m elements form a solution, i.e. $\sum_{i=1}^m s(p_i) = \sum_{i=m+1}^n s(p_i) = c/2$. Then the graph G can be decomposed into the following $3n$ induced subgraphs:

$$G_i = \begin{cases} G(\{u_i, v_i, u, v\}), & i = 1, 2, \dots, m, \\ G(\{u_i, v_i\}), & i = m+1, \dots, n, \end{cases}$$

$$G_{n+i} = G(\{u, v_i\}), \quad i = 1, 2, \dots, n,$$

$$G_{2n+i} = G(\{u_i, v\}), \quad i = 1, 2, \dots, n,$$

where the uniform weight assigned to the edges in G_i is $s(p_i)$, $i \leq n$, and the weight assigned to the edges in G_{n+i} and G_{2n+i} is $w((u, v_i)) - s(p_i)$ for $i \leq m$ and $w((u, v_i))$ for $i \geq m+1$.

Assume the PARTITION instance P has no solution. Note that there are only three types of complete bipartite subgraphs that contain “horizontal” edges (u_i, v_i) , $i = 1, 2, \dots, n$:

- Type-1—subgraphs induced on vertex subsets $\{u_i, v_i\}$, $i = 1, 2, \dots, n$.
- Type-2—subgraphs induced on vertex subsets $\{u, v, u_i, v_i\}$, $i = 1, 2, \dots, n$.
- Type-3—subgraphs induced on vertex subsets $\{u_i, v_i, v\}$ or $\{u_i, v_i, u\}$, $i = 1, 2, \dots, n$.

Obviously, no complete bipartite subgraph contains two different edges (u_i, v_i) and (u_j, v_j) ; and only type-2 subgraphs contain both (u, v) and (u_i, v_i) . Hence, since P is not solvable, any complete bipartite subgraph decomposition of G includes at least $n+1$ weighted complete bipartite subgraphs that contains (u, v) or (u_i, v_i) as an edge, $i = 1, 2, \dots, n$. Let S be the set of weighted complete bipartite subgraphs that form a decomposition of G and let $S_0 \subseteq S$ be the set of subgraphs that contains (u, v) or (u_i, v_i) for some $i \leq n$. For each $i \leq n$, we also let $S_i \subseteq S$ be the set of subgraphs that contains (u, v_i) and let $S_{n+i} \subseteq S$ be the set of subgraphs that contains (u_i, v) . We have shown that

$|S_0| \geq n + 1$. Obviously, for any i, j , $(S_i - S_0)$ is disjoint from $S_{n+j} - S_0$. To prove that $|S| \geq 3n + 1$, we need only prove that $|\bigcup_{i=1}^n S_i - S_0| \geq n$ and $|\bigcup_{i=1}^n S_{n+i} - S_0| \geq n$.

The sum of weights assigned to graphs in S_0 is at most $c + c/2 = 1.5c$. For each $i \geq 1$, the sum of weights assigned to graphs in S_i is exactly $w(u, v_i)$. Since $w(u, v_1) = 1.5c + s(p_1) > 1.5c$, $S_1 - S_0$ is non-empty. For any $j \geq 2$, we have $w(u, v_j) = s(p_j) + 2w(u, v_{j-1}) > \sum_{k=1}^{j-1} w(u, v_k) + 1.5c$, since the right-hand side is

$$\begin{aligned} & \sum_{k=1}^{j-1} \left(2^{k-2} 3c + \sum_{i=1}^k 2^{k-i} s(p_i) \right) + 1.5c \\ &= \left(3c \sum_{k=1}^{j-1} 2^{k-2} + 1.5c \right) + \sum_{k=1}^{j-1} \sum_{i=1}^k 2^{k-i} s(p_i) \\ &= (2^{j-1}) 1.5c + \sum_{i=1}^{j-1} \sum_{k=i}^{j-1} 2^{k-i} s(p_i) \\ &= 2^{j-2} 3c + \sum_{i=1}^{j-1} 2^{j-i} s(p_i) = 2w(u, v_{j-1}). \end{aligned}$$

It follows that we have $S_j - \bigcup_{i=0}^{j-1} S_i$ is non-empty for all $j \geq 1$. Therefore, we have that $|\bigcup_{i=1}^n S_i - S_0| \geq n$. By symmetry, we have $|\bigcup_{i=1}^n S_{n+i} - S_0| \geq n$. This finishes the proof of the fact and hence Lemma 4.1. \square

5. Conclusion. We conclude this work by giving two more remarks. In this paper we have studied the complexity, approximation and algorithmic issues of three clustering problems arising from microarray data analysis. We show that all the clustering problems arising in [4], [16] and [24] are NP-hard. This justifies the random and greedy algorithms proposed for these problems in [4], [16] and [24], respectively.

In Section 2 we give an approximation-ratio-preserving reduction from the BCBS problem to the OPSM problem. However, since inapproximability of the BCBS problem is unknown, it is interesting to know whether the OPSM problem can be approximated within a constant ratio in polynomial time.

References

- [1] A. Alizadeh, M. B. Eisen, R. E. Davis, C. Ma, I. S. Lossos, A. Rosenwald, J. C. Boldrick, H. Sabet, T. Tran, X. Yu, J. I. Powell, L. Yang, G. E. Marti, T. Moore, J. Jr. Hudson, L. Lu, D. B. Lewis, R. Tibshirani, G. Sherlock, W. C. Chan, T. C. Greiner, D. D. Weisenburger, J. O. Armitage, R. Warnke, R. Levy, W. Wilson, M. R. Grever, J. C. Byrd, D. Botstein, P. O. Brown and L. M. Staudt, Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling, *Nature* **403** (2000), 503–510.
- [2] G. Ausiello, P. Crescenzi, G. Gambosi, V. Kann, A. Marchetti-Spaccamela and M. Protasi, *Complexity and Approximation*, Springer-Verlag, New York, 1999.
- [3] A. Ben-Dor and Z. Yakhini, Clustering gene expression patterns, in *Proceedings of RECOMB '99*, pp. 33–42.
- [4] 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 RECOMB '02*, pp. 49–57.

Three Clustering Methods in Microarray Data Analysis

- [5] P. Berman, B. DasGupta, S. Muthukrishnan and S. Ramaswami, Efficient approximation algorithm for tiling and packing problems with rectangles, *J. Algorithms* **41** (2001), 443–470.
- [6] Y. Chen, E. Dougherty and M. Bitter, Ratio-based decisions and the quantitative analysis of cDNA microarray images, *J. Biomed. Optics* **2** (1997), 364–374.
- [7] Y. Cheng and G. Church, Biclustering of expression data, in *Proceedings of ISMB 2000*, pp. 93–103.
- [8] T. H. Cormen, C. E. Leiserson, R. L. Rivest and C. Stein, *Introduction to Algorithms* (2nd edn.), McGraw-Hill, New York, 2001.
- [9] M. B. Eisen, P. T. Spellman, P. O. Brown, and D. Botstein, Clustering analysis and display of genome-wide expression pattern, *Proc. Natl. Acad. Sci. USA* **95** (1998), 14863–14868.
- [10] M. R. Garey and D. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*. Freeman, San Francisco, CA, 1979.
- [11] E. Hartuv, A. O. Schmitt, J. Lange, S. Meier-Ewert, H. Lehrach and R. Shamir, An algorithm for clustering cDNAs for gene expression analysis, in *Proceedings of Recomb '99*, pp. 188–197.
- [12] I. Hedenfalk, D. Duggan, Y. Chen, M. Radmacher, M. Bittner, R. Simon, P. Meltzer, B. Gusterson, M. Esteller, O. P. Kallioniemi, B. Wilfond, A. Borg, J. Trent, M. Raffeld, Z. Yakhini, A. Ben-Dor, E. Dougherty, J. Kononen, L. Bubendorf, W. Fehrle, S. Pittaluga, S. Gruvberger, N. Loman, O. Johannsson, H. Olsson, and G. Sauter, Gene-expression profiles in hereditary breast cancer, *New England J. Medicine* **344** (2001), 539–548.
- [13] D. S. Hochbaum, *Approximation Algorithms for NP-hard Problems*, PWS, Boston, MA, 1995.
- [14] T. G. Kolda and D. P. O’Leary, A semidiscrete matrix decomposition for latent semantic indexing in information retrieval, *ACM Trans. Inform. Systems* **16** (1998), 322–346.
- [15] E. L. Lawler, *Combinatorial Optimization: Networks and Matroids*, Holt, Rinehart and Winston, New York, 1976.
- [16] L. Lazzeroni and A. Owen, Plaid models for gene expression data, *Statist. Sinica* **12** (2002), 61–86. See <http://www-stat.stanford.edu/~owen> for more about Plaid model.
- [17] D. D. Lee and H. S. Seung, Learning the parts of objects by non-negative matrix factorization, *Nature* **401** (1999), 788–791.
- [18] J. Liu, J. Yang and W. Wang, Biclustering in gene expression data by tendency, in *Proceedings of CSB '04*, pp. 182–193.
- [19] R. Peeters, The maximum edge biclique problem is NP-complete, *Discrete Appl. Math.* **131** (2003), 651–654.
- [20] P. Tamayo, D. Slonim, J. Mesirov, Q. Zhu, S. Kitareewan, E. Dmitrovsky, E. Lander and T. Golub, Interpreting patterns of gene expression with self-organizing maps: methods and application to hematopoietic differentiation, *Proc. Natl. Acad. Sci. USA* **96** (1999), 2907–2912.
- [21] O. Troyanskaya, M. Cantor, G. Sherlock, P. Brown, T. Hastie, R. Tibshirani, D. Botstein and R. B. Altman, Missing value estimation methods for DNA microarrays, *Bioinformatics* **17** (2001), 520–525.
- [22] M. Yannakakis, Node- and edge-deletion NP-complete problems, in *Proceedings of the 10th Annual STOC*, pp. 253–264, 1978.
- [23] L. Zhang and S. Zhu, Complexity study on two clustering problems, in *Proceedings of the Annual International Symposium on Algebra and Computing*, pp. 660–669, 2001.
- [24] L. Zhang and S. Zhu, A new approach to clustering gene expression data, in *Proceedings of IEEE Symposium on Bioinformatics*, pp. 268–275, 2002.