Rough Overlapping Biclustering of Gene Expression Data

Ruizhi Wang Duoqian Miao Gang Li Hongyun Zhang
Department of Computer Science and Technology
Tongji University
Shanghai, P.R.China

Abstract—**A great number of biclustering algorithms have been proposed for analyzing gene expression data. Many of them assume to find exclusive biclusters whose subsets of genes are co-regulated under subsets of conditions without intersection. This is not consistent with a general understanding of biological processes that many genes participate in multiple different processes. Therefore nonexclusive biclustering algorithms are required. In this paper we present a novel approach (ROB) to find potentially overlapping biclusters in the framework of generalized rough sets. Our scheme mainly consists of two phases. First, we generate a set of highly coherent seeds (original biclusters) based on two-way rough k-means clustering. And then, the seeds are iteratively adjusted (enlarged or degenerated) by adding or removing genes and conditions based on a proposed criterion. We illustrate the method on yeast gene expression data. The experiments demonstrate the effectiveness of this approach.**

Keywords-biclustering; gene expression data; rough clustering; overlapping biclusters

I. INTRODUCTION

Clustering has been one of the most popular approaches of analyzing gene expression data and has proven to be successful in many applications, such as discovering gene pathway, gene classification, and function prediction. Traditional clustering methods, such as hierarchical clustering[1], K-means[2], or SOM[3] assume that genes in a cluster behave similarly over all the conditions. These methods produce reliable results for microarray experiments performed on homogeneous conditions. However, when the conditions of an experiment vary greatly, the assumption is no longer appropriate. In this case, it is desirable to develop approaches that can detect those relevant conditions under which the behavior similarity between genes of a potential group exists. This leads to a promising paradigm of clustering, biclustering. Unlike traditional clustering that reveals genes behaving similarly over all the conditions, biclustering of expression data captures a subset of genes exhibiting strikingly pattern similarity (coherent fluctuation) across a subset of conditions. So biclustering paradigm is more consistent with a general understanding of cellular processes that subsets of genes are co-regulated and co-expressed under certain experimental conditions, but behaves almost independently under other conditions.

The notion of biclustering (also known as direct clustering [4], co-clustering [7], box clustering[8]) was first introduced by Hartigan[4] in 1972 to describe simultaneously grouping of both row and column subsets in a data matrix. The term biclustering was first used by Cheng & Church [6] in gene expression data analysis in 2000. Since then, numerous microarray biclustering algorithms have been developed. Yang et al.[9]generalized the additive biclustering model proposed by Cheng & Church to incorporate null values and proposed a probabilistic algorithm(FLOC) that can discover a set of k possibly overlapping biclusters simultaneously. Getz et al.[10] developed the coupled two-way iterative clustering approach to identify biclusters. Tang et al.[11]presented the interrelated two-way clustering algorithm that combines the results of one-way clustering on both gene and sample dimensions to produce biclusters. Lazzeroni and Owen[12] introduced the plaid model which can be seen as a generalization of the additive model where the value of an element in the data matrix is viewed as a sum of terms called layers(biclusters). Segal et al.[13]proposed rich probabilistic models based on the language of probabilistic relational models which allows to include multiple types of information to identify similar objects. Its outcome can be interpreted as a collection of disjoint biclusters generated in a supervised manner. Tanay et al.[14]combined graph theoretic and statistical considerations and devised the SAMBA algorithm by modeling the expression data as a bipartite graph and transforming the biclustering problem to that of finding the heaviest subgraphs in a bipartite graph. This approach can find statistically significant biclusters in gene expression data. Kluger et al.[15]presented a spectral biclustering method to find distinctive checkerboard patterns in expression matrices. The checkerboard structures can be found in eigenvectors which can be readily identified by commonly used linear algebra approaches. Sheng et al.[16]implemented Gibbs sampling to biclustering discretized microarray data. In [17], Wu et al. proposed a simpler Gibbs sampling scheme and expand its application to biclustering continuous gene expression data. Cano et al.[5]recently proposed a possibilitic spectral biclustering algorithm (PSB) to obtain potentially overlapping biclusters, based on Fuzzy Technology and Spectral Clustering. A review of most biclustering methods can be found in [25]. Since the biclustering problem is proven to be NP-hard[6], most biclustering algorithms use heuristic approaches that are not guaranteed to find optimal solutions. To address the problem, some stochastic search techniques have been employed recently due to their potential to escape local minima. Bryan et al.[18] exploited Simulated Annealing to improve upon greedy techniques. Divina and Jesús[19]used

Supported by the National Natural Science Foundation of China (No. 60475019) and the Research Fund for the Doctoral Program of Higher Education (No. 20060247039).

Evolutionary Algorithms to search for biclusters following a sequential covering strategy.

Most of above algorithms find exclusive biclusters which is inappropriate in the biological context. Since biological processes are not independent of each other, many genes participate in multiple different processes. Each gene therefore should be assigned to multiple biclusters whenever biclusters are identified with processes.

We addressed the above concern in the framework of generalized rough sets [20][24] by associating each bicluster to a pair of distinct sets, a lower approximation and an upper approximation. The lower approximation is a subset of the upper approximation. The members (genes or conditions) of the lower approximation belong to and only belong to the bicluster. However, the members of the upper approximation may belong to more than one biclusters among which there are nonempty intersections. The boundary region between the lower and upper approximation forms an overlapping part among corresponding biclusters. Therefore it is expected that lower and upper approximation derived directly from expression data would better capture the overlapping feature among the co-regulated genes.

In this paper, we propose a novel biclustering approach based on generalized rough sets, named Rough Overlapping Biclustering(ROB for short), to find biclusters of maximum size, with stronger coherence, and particularly with a reasonable degree of overlapping. Our approach uses a similar bicluster model given in [19], which is a generalized bicluster model of Cheng & Church [6]. The main processes of our approach consist of two phases: seeds generating phase and iterative bicluster refining phase. In the seeds generating phase, we devise a two-way rough k-means clustering algorithm which can produce naturally a set of very tightly co-regulated submatrices or seeds from expression data. Starting with these initial submatrices(or seeds), our approach can avoid random interference suffered by replacing the missing value and masking discovered biclusters with random numbers in [6]. In the iterative bicluster refining phase, seeds are enlarged or degenerated when the biclusters membership of each row/column in expression data is adjusted iteratively. The rule of thumb is to guide the adjustment to optimize the overall quality of biclusters. To this end, we introduce a criterion by which the biclusters membership of each row/column can be decided. Base on the criterion, we propose a novel overlapping biclustering algorithm to discover a set of overlapping biclusters simultaneously. Our algorithm iterates adjusting the memberships of rows and/or columns in a random order until there exists a row or column whose potential operation would lead to the corresponding mean squared residue being bigger than a predefined threshold δ .

We implemented the proposed method to find 100 biclusters on the same yeast data containing 2884 genes and 17conditions with same parameters setting as in [6] and [9]. The results show that the biclusters produced by our approach, on average, have comparative size but a smaller mean squared residue than that of Cheng &Church's algorithm [6] and FLOC [9], partly because the proposed method avoids the random interference suffered by the Cheng & Church's algorithm and partly because the use of two-way clustering strategy to construct seeds (initial biclusters) well captures the intrinsic coherence of expression data. In addition, our approach can produce a set of δ -biclusters simultaneously with a reasonable degree of overlapping, due to introducing of the lower and upper approximation.

The remainder of the paper is organized as follows. Section 2 introduces the general model of bicluster. Section 3 presents the algorithm in detail. Section 4 gives the experimental results on yeast expression dataset. And finally, the conclusion is provided in Section 5.

II. THE MODEL OF BICLUSTER

In this section, we present a brief description of the bicluster model that is similar to the bicluster model in [19] and a merit function to assess the quality of a bicluster.

A bicluster is defined on a gene-expression data. Let $G = \{g_1, \cdots, g_N\}$ be a set of genes and $C = \{c_1, \cdots, c_M\}$ be a set of conditions. The data can be viewed as an $N \times M$ expression matrix D. D is a matrix of real numbers, with possible null values, where each entry d_{ij} corresponds to the logarithm of the relative abundance of the mRNA of a gene g_i under a specific condition c_j . Although each entry in expression matrix D may have a null value, each entry in a bicluster is supposed to be specified in this paper.

A bicluster essentially corresponds to a submatrix that exhibits some coherent tendency. Each bicluster can be uniquely identified by a set of relevant genes and conditions, which determine the submatrix. Thus, a bicluster is a matrix $I \times J$, represented by a pair (I, J) where $I \subseteq \{1, \cdots, N\}$ is a subset of genes (rows) and $J \subseteq \{1, \cdots, M\}$ is a subset of conditions [19].

Definition 2.1 The **volume** of a bicluster (I, J) is defined as the number of entries d_{ij} such that $i \in I$ and $j \in J$ [19].

In order to assess the quality of a bicluster, we use mean squared residue defined by Cheng & Church [6]. In the following, we give some definitions related to the measure, which are taken from [6] [9] [19].

Definition 2.2 For a given bicluster (I, J), the **base** of a gene g_i is defined as $d_{iJ} = \sum_{j \in J} d_{ij} / |J|$. Similarly, the base of a condition c_j is defined as $d_{Ij} = \sum_{i \in I} d_{ij} / |I|$. The base of a bicluster is the average value of all the entries contained in (I, J), $d_{IJ} = \dfrac{\sum_{i \in I, j \in J} d_{ij}}{|I| \cdot |J|}$.

Defintion 2.3 The **residue** of an entry d_{ij} in a bicluster (I, J) is $r_{ij} = d_{ij} - d_{iJ} - d_{Ij} + d_{IJ}$.

The residue is an indicator of the degree of coherence of an entry with the remaining entries in the bicluster, given the tendency of the relevant gene and the relevant condition. The

lower the residue is, the stronger the coherence. To evaluate the overall coherence of a bicluster, Cheng & Church [6] defined the mean squared residue of a bicluster (I,J) as the sum of the squared residue.

Definition 2.4 The **mean squared residue** of a bicluster (I,J) is $H(I,J)=\frac{1}{|I|\cdot|J|}\sum_{i\in I,j\in J}r_{ij}^2$

The mean squared residue well indicates the overall coherence of a bicluster. The lower the mean squared residue, the stronger the coherence exhibited by the bicluster, and the better quality of the bicluster. A bicluster is called δ-bicluster if $H(I,J)\le\delta$ for some $\delta\ge 0$. It has been proven that the problem of finding the largest square δ-biclusters is NP-hard [6].

In this paper, we are interested in finding biclusters of maximum size, with relative small mean squared residue (lower than a given δ), and particularly with a reasonable degree of overlapping. We define the overlapping degree of two biclusters as follows.

Definition 2.5 Given two biclusters A and B, the **overlapping degree** R of the two biclusters is defined as $R=|A\cap B|\ /\ |A\cup B|$, where $|A\cap B|$ is the volume of $A\cap B$, $|A\cup B|$ is the volume of $A\cup B$.

III. ROUGH OVERLAPPING BICLUSTERING

In this section, we present a description in detail of our Rough Overlapping Biclustering (ROB) approach, which can effectively and efficiently find a set of overlapping biclusters with relative lower average mean squared residue and average larger volume simultaneously. The ROB first generates a set of very tightly co-regulated submatrices or seeds from expression data. Starting with the seeds (initial biclusters), an iterative adjustment process is conducted to improve the overall quality of these biclusters. At each iteration, the bicluster membership of each gene (row) and/or condition (column) is adjusted to produce better biclusters in terms of lower average residue and larger volume. The algorithm terminates when there exists a row or column whose potential operation (add or remove) would lead to the corresponding mean squared residue being bigger than threshold δ.

In ROB approach, a pre-processing procedure is conducted first, and then a two-way rough *k*-means clustering algorithm as well as an overlapping biclustering algorithm is applied on the gene expression matrix .The whole procedure of Rough Overlapping Biclustering is presented in Fig.1.

A. Pre-processing of Data

In gene expression matrix, different genes have different range of intensity values. In order to eliminate the influence of different gene-dimensions, the data are normalized by the formula as follows [11].

$$d_{ij}^{'}=\frac{d_{ij}-\mu_i}{\mu_i},\quad \text{where}\quad \mu_i=\frac{1}{m}\sum_{j=1}^{m}d_{ij}\ , \qquad (1)$$

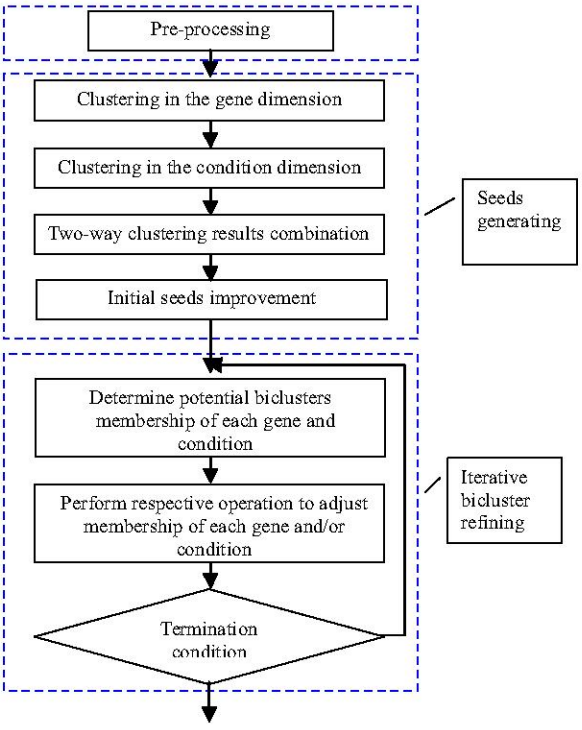

Figure 1. The structure of ROB

$d_{ij}^{'}$ represents normalized intensity value for gene i under condition j, d_{ij} denotes the original intensity value for gene i under condition j, m is the number of conditions, and μ_i is the mean of the intensity value of the original intensity values for gene i over all conditions.

Gene expression matrix usually has thousands of genes (rows). Among them, some genes have little reaction to the experiment conditions and contribute little in biclustering the data. We believe genes whose normalized intensity values keep invariant or fluctuate very little belong to this class. These genes which are called 'flat genes' in the remainder of the paper need to be removed. Cheng & Church [6] introduced the row variance as an accompanying score to reject trivial biclusters where there is no fluctuation. Their method can remove 'flat genes' in the trivial biclusters during each iteration of greedy node deletion algorithm, but increase computing complexity as well. In this paper, we consider to remove 'flat genes' in the preprocessing phase and reduce gene-dimension at the same time. We follow the method proposed in [11].

Let's assume we have n genes and m conditions. Each gene is represented by m-dimensional vector (after normalization) as follows.

$$g_i=(d_{i1}^{'},d_{i2}^{'},\cdots,d_{im}^{'}), \qquad (2)$$

where $i=1,2,\cdots,n$ for each gene. We use vector-cosine between each gene vector and a pre-defined stable pattern E to test whether a gene intensity value (after normalization) varies much over all conditions. The pattern can be denoted as $E=(e_1,e_2,\cdots,e_m)$, where all e_i are equal[11].

$$\cos(\theta) = \frac{< \vec{g}_i, \vec{E} >}{\|\vec{g}_i\| \cdot \|\vec{E}\|} = \frac{\sum_{j=1}^{m} d_{ij}^{'} \times e_j}{\sqrt{\sum_{j=1}^{m} d_{ij}^{'2}} \times \sqrt{\sum_{j=1}^{m} e_j^2}}, \qquad (3)$$

where θ is the angle between $\vec{g}_i$ and $\vec{E}_i$ in *m*-dimensional space. If the two vector patterns are more similar, the vector-cosine will be closer to 1. Therefore, we can choose a threshold η to remove genes matching pattern E (those genes' vector-cosine values with E are higher than the threshold η).

After the pre-processing procedure, we usually reduce twenty to thirty percent of genes, which facilitates biclustering in the next stage.

B. Two-way Rough K-means Clustering

To handle missing value and avoid random interference suffered by Cheng & Church's algorithm [6], our approach ROB starts with a set of seeds (original small biclusters) from gene expression data. Intuitively, seeds that demonstrate higher coherence will facilitate refining biclusters with lesser iteration steps in the next phase. We address these concerns within the framework of Two-Way Clustering (TWC). Generally, any standard clustering method can be used in the framework of TWC. The optimal algorithm for overlapping biclustering analysis of gene expression data should have the potential to produce overlapping clusters. Therefore, we choose rough *k*-means clustering algorithm [21][22] and use it on both gene and condition dimensions.

In rough *k*-means clustering, each cluster has two approximations, a lower and an upper approximation. Strictly speaking, the lower and upper approximations are not looked upon as Pawlak's rough sets [20][28], but rather interval sets [23].

The rough *k*-means algorithm begins by randomly choosing *k* objects as the centroids(means) of the *k* clusters. The objects are assigned to the lower approximation or upper approximation based on the following **ratio** [22].

For each object vector $\mathbf{v}$, let $d(\mathbf{v}, \mathbf{m}_j)$ be the distance between itself and the centroid of cluster $\mathbf{m}_j$, and $d(\mathbf{v}, \mathbf{m}_i) = \min_{1 \le j \le k} d(\mathbf{v}, \mathbf{m}_j),\ 1 \le i, j \le k.$ The **ratio** $d(\mathbf{v}, \mathbf{m}_i) / d(\mathbf{v}, \mathbf{m}_j)$ is used to determine the membership of $\mathbf{v}$. Let

$$T = \{ j \,|\, d(\mathbf{v}, \mathbf{m}_j) / d(\mathbf{v}, \mathbf{m}_i) \le \zeta \text{ and } i \ne j \},$$

1. If $T \ne \varnothing, \mathbf{v} \in \overline{A}(\mathbf{m}_i)$ and $\mathbf{v} \in \overline{A}(\mathbf{m}_j), \forall j \in T$.
2. Otherwise, if $T = \varnothing, \mathbf{v} \in \underline{A}(\mathbf{m}_i)$.

After the assignment of all the objects to various clusters, the new centriod (means) vectors of the clusters are calculated by:

if $\overline{A}(\mathbf{m}_j) - \underline{A}(\mathbf{m}_j) \ne \varnothing \wedge \underline{A}(\mathbf{m}_j) \ne \varnothing$

$$\omega_l \frac{\sum_{\mathbf{v} \in \underline{A}(\mathbf{m}_j)} \mathbf{v}}{\left| \underline{A}(\mathbf{m}_j) \right|} + \omega_b \frac{\sum_{\mathbf{v} \in (\overline{A}(\mathbf{m}_j) - \underline{A}(\mathbf{m}_j))} \mathbf{v}}{\left| \overline{A}(\mathbf{m}_j) - \underline{A}(\mathbf{m}_j) \right|}$$

else if $\overline{A}(\mathbf{m}_j) - \underline{A}(\mathbf{m}_j) = \varnothing \wedge \underline{A}(\mathbf{m}_j) \ne \varnothing$

$$\frac{\sum_{\mathbf{v} \in \underline{A}(\mathbf{m}_j)} \mathbf{v}}{\left| \underline{A}(\mathbf{m}_j) \right|}$$

else if $\overline{A}(\mathbf{m}_j) - \underline{A}(\mathbf{m}_j) \ne \varnothing \wedge \underline{A}(\mathbf{m}_j) = \varnothing$

$$\frac{\sum_{\mathbf{v} \in (\overline{A}(\mathbf{m}_j) - \underline{A}(\mathbf{m}_j))} \mathbf{v}}{\left| \overline{A}(\mathbf{m}_j) - \underline{A}(\mathbf{m}_j) \right|}$$

where $1 \le j \le m$. The parameters ω_l and ω_b correspond to the relative importance of lower approximation and boundary region of the cluster, and $\omega_l + \omega_b = 1$. The expression $\left| \underline{A}(\mathbf{m}_j) \right|$ indicates the number of objects in lower approximation of the cluster and $\left| \overline{A}(\mathbf{m}_j) - \underline{A}(\mathbf{m}_j) \right|$ is the number of objects in the boundary region.

The process stops when the centroids of clusters stabilize, i.e. the centroid vectors from the previous iteration are identical to those generated in the current iteration.

In the paper, we apply the rough *k*-means method on the gene(row) and condition(column) dimensions of the expression data matrix after pre-processing procedure separately, and then combine the results to obtain seeds(small co-regulated submatrices). Given a gene expression data matrix D, let k_g be the number of clusters on gene-dimension and k_c be the number of clusters on condition-dimension after rough k-means clustering. C^g is the family of gene clusters and C^c denotes the family of condition clusters. Let c_i^g be a subset of genes and $c_i^g \in C^g$, ($1 \le i \le k_g$). Let c_j^c be a subset of conditions and $c_j^c \in C^c$ ($1 \le j \le k_c$). The pair (c_i^g, c_j^c) denotes a submatrix (seeds) of D. Therefore, by combining the results of gene-dimensional rough *k*-clustering and condition-dimensional rough *k*-clustering, we obtain $k_g \times k_c$ seeds.

Two-way clustering methods apply standard clustering methods on the row and column dimension of data matrix separately and combine the results to obtain biclusters. The combined results exhibit similarity on either gene dimension or condition dimension; however, they may not well capture the overall coherence of both a subset of genes and a subset of conditions. To improve the seeds obtained by two-way rough k-means, we use the Single Node Deletion algorithm given in [6] to remove gene (row) or column (condition) such that the mean squared residue of each seed is less than or equal to a predefined threshold. Among these refined seeds, we are interested in those which exhibit relative higher coherence and larger size. Thus, we choose the largest $K < k_g \times k_c$ seeds as input to the following overlapping biclustering algorithm.

C. *Overlapping Biclustering Based on Generalized Rough Sets*

In this section, we present a novel overlapping biclustering algorithm based on generalized rough sets, which can effectively and efficiently approximate a set of overlapping biclusters simultaneously with relative lower mean squared residue. This algorithm starts from a set of seeds produced by a preceding two-way rough k-means clustering algorithm and iteratively conducts adjustments to approach the best solution.

In the framework of generalized rough sets[20][24], we view each bicluster as a generalized rough set which has two approximations, a lower and an upper approximation. The lower approximation is a subset of the upper approximation. The members (genes or conditions) of the lower approximation belong to and only belong to the bicluster. However, the members of the upper approximation may belong to the bicluster as well as other biclusters. Therefore, the boundary region between the lower and upper approximation forms an overlapping part among corresponding biclusters.

Given a gene expression data matrix D, for each object (gene or condition), there are possible three kinds of bicluster membership. They are as follows.

- It may not belong to any biclusters in D.
- Otherwise, if the object belongs to a bicluster, it is key step to determine whether an object belongs to the lower approximation of the bicluster or upper approximation of the bicluster in D.

To determine the bicluster membership, we give the following **criterion**.

For each object(gene or condition) vector $\mathbf{v}$, let $\Delta H(\mathbf{v}, X_j) = H'_{X_j} - H_{X_j}$ ($1 \le i, j \le k$, k is the number of biclusters) be the **gain** of the adjustment (insert or remove), where H_{X_j} and H'_{X_j} are mean squared residue of bicluster X_j (see definition 2.4) before and after the $\mathbf{v}$ is inserted to or remove from bicluster X_j, respectively. Let $\Delta H(\mathbf{v}, X_i) = \min_{1 \le j \le k} \Delta H(\mathbf{v}, X_j)$. The **difference** of gains $\Delta H(\mathbf{v}, X_j) - \Delta H(\mathbf{v}, X_i)$ is used to determine the membership of $\mathbf{v}$. Let

$$T = \{ j \,|\, \Delta H(\mathbf{v}, X_j) - \Delta H(\mathbf{v}, X_i) \le \varepsilon \text{ 且 } i \ne j \},$$

1. If $H'_{X_j} \ge \delta$, $\mathbf{v} \notin X_j \Rightarrow \mathbf{v} \notin \overline{X}_j, \mathbf{v} \notin \underline{X}_j$;
2. Otherwise, $\mathbf{v} \in X_j$. Furthermore,
   a) if $T \ne \varnothing$, $\mathbf{v} \in \overline{X}_i$ and $\mathbf{v} \in \overline{X}_j, \forall j \in T$;
   b) otherwise, if $T = \varnothing$, $\mathbf{v} \in \underline{X}_i$.

where δ and ε are predefined thresholds. The parameter δ is to guarantee that all biclusters discovered have mean squared residues less than δ. The parameter ε determines the degree of overlapping among these biclusters. By selecting proper ε, we can find a set of overlapping biclusters simultaneously.

Based on the **criterion**, we devise the overlapping biclustering algorithm as shown in fig.2. Given gene expression matrix D, n is the number of genes in D, and m is the number of conditions in D.

Input: $X_j, 1 \le j \le K$, a set of K seeds.

Output: $\overline{X}_j, 1 \le j \le k \le K$, a set of k overlapping biclusters.

Iteration:

1) generate a random sequence of genes and conditions;
   for each object(gene or condition) $\mathbf{v}$ along the sequence, do
   1.1) compute every gain $\Delta H(\mathbf{v}, X_j)$ for the possible insert/remove and find the minimal **gain.**
   1.2) compute the $\Delta H(\mathbf{v}, X_j) - \Delta H(\mathbf{v}, X_i)$ and find T, do
   If $H'_{X_j} \ge \delta$, remove $\mathbf{v}$ from $\overline{X}_j$;
   Else if $H'_{X_j} \le \delta$ and $T \ne \varnothing$, insert $\mathbf{v}$ to $\overline{X}_i$ and $\overline{X}_j$, for $\forall j \in T$;
   Else if $H'_{X_j} \le \delta$ and $T = \varnothing$, insert $\mathbf{v}$ to $\underline{X}_i$.
2) goto 1), until termination condition for adjustment is satisfied.
3) output the best solution.

Figure 2. Overlapping biclustering algorithm based on Rough Sets

Our heuristic algorithm is sensitive to the input order of genes and conditions in each iteration. Note for gene expression data, the number of genes is much larger than the number of conditions. It may be more acceptable to put genes in front of conditions. Therefore, in this paper, we generate a random sequence at the beginning of each iteration by the following strategy. We first generate a random list of genes and a random list of conditions respectively, and then combine them by arranging the gene list in front of the condition list. In the each iteration, genes and conditions are examined one by one in the proposed random sequence. The adjustment to a gene or condition will be conducted based on the proposed criterion. The process iterates until the termination condition is met. In order to explain the termination condition, we define the Average H/Volume ratio as follows.

$$\text{Avg.H/V} = \frac{1}{k} \sum_{j=1}^{k} \frac{\text{mean squared residue of bicluster } X_j}{\text{volume of bicluster } X_j} \quad (4)$$

Average H/Volume ratio reflects the overall quality of biclusters generated in the algorithm. The lower the Average H/Volume is, the higher the coherence of biclusters exhibits. Suppose the algorithm will obtain the best solution after many iterations, it can be expected that the Average H/Volume ratio

will stabilize from then on, i.e. the Average H/Volume ratio from the previous iteration is identical to the one generated in the current iteration. Thus, the iteration will be stopped when the Average H/Volume ratio stabilized.

IV. Experimental Results

To evaluate the performance of our ROB algorithm, we conduct experiments on yeast expression data [26], available at website [27]. The yeast expression matrix consists of 2884 genes and 17 experimental conditions. We compare our ROB with the Cheng & Cheng's algorithm [6] and FLOC[9] on the same yeast expression data set with the same parameter setting, i.e. $\delta = 300$, and for the same objective to find 100 largest biclusters with mean squared residue less than 300.

A. Parameters Setting

As most threshold soft clustering methods, the proposed ROB uses several parameter to approximate the optimal solution.

During data pre-processing procedure, by sorting genes using vector-cosine calculated from (3), we choose threshold 0.7(See Table I), then remove genes which vector-cosine with pattern E is higher than that threshold. 802 genes which intensity values vary little across the conditions are removed from 2884 genes.

In the two-way clustering phase, we use rough *k*-means clustering algorithm on both gene dimension and condition dimension. Euclidean distance is exploited as the distance measure. Weight pair (ω_l , ω_b) is set to (0.75, 0.25) as proposed in [21]. Various gene and condition radio threshold pair (ζ_g , ζ_c) values ranging [1.1, 1.5] are tried. It is found that when the values of pair (ζ_g , ζ_c) are set at (1.3,1.2), the resulting clusters leaded to good seeds with lower average mean squared residue. We group yeast expression data into 284 gene clusters and 5 condition clusters, and then combine them into 284*5 submatrices (seeds) each containing at least 10 close genes and at least 5 close conditions.

During the iterative overlapping biclustering procedure, we choose the same parameter setting δ as that reported by [6][9],i.e. $\delta = 300$, and then remove those genes or conditions that will cause corresponding mean squared residue larger than 300. As for ε, we try $\varepsilon = 0,1,2,\cdots,18$. It shows that when ε is chosen from 8 to 10, the average H/Volume of final biclusters approximates the global optimum, i.e. 0.12 (See Fig.3). From $\varepsilon \geq 10$, the average H/Volume increases a little and then stabilizes. Moreover, it is interesting to note that our algorithm delivers meaningful results over the range [7,9] of ε where the overlapping degree increase dramatically from 3% and reach its maximum 18%, as depicted in Fig.4. For ε smaller than 7, average degree of overlapping fluctuates little and nears a constant value 0.02. When ε is greater than 9, the average overlapping degree decreases quickly and convergences towards a stable value 0.1. The Fig.4 shows the overlapping degree of biclusters in dependency of ε. In order to produce biclusters of lower average mean squared residue, with larger volume and reasonable degree of overlapping, we set the threshold of ε to 8 in this paper.

TABLE I. Parameter Setting

Parameter		*Value*
Preprocessing	Vector-cosine threshold η	0.7
Two-Way Rough k-means clustering	Lower-approximation weigh ω_l	0.75
	Boundary-region weight ω_b	0.25
	Gene clusters num. k_g	284
	Condition clusters num. k_c	5
	Gene radio threshold ζ_g	1.3
	Condition radio threshold ζ_c	1.2
Overlapping biclustering	Mean squared residue threshold δ	300
	Difference threshold ε	8

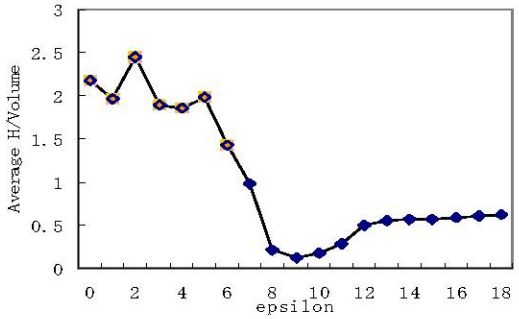

Figure 3. Relationship between Average H/Volume and ε

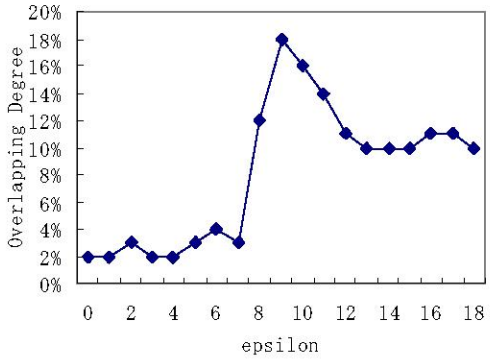

Figure 4. Relationship between overlapping degree and ε

B. Performance Comparisons

Table II shows the performance comparison of ROB with that of Cheng & Church's algorithm (henceforth CC)[6] and the algorithm FLOC[9] for what concerns the average mean squared residue and the average dimension of the biclusters found. We can see that ROB and FLOC are comparable in terms of finding biclusters characterized by a larger average volume than the ones detected by CC. This is largely due to the different strategies adopted by ROB and FLOC to avoid random interference suffered by CC[6].As for average number of genes and average number of conditions, the proposed ROB discovers biclusters with higher number of genes and lower

number of condition averagely than that of CC and FLOC. This is probably because ROB puts the random gene list in front of random condition list at the beginning of each iteration and has a bias towards finding coherent genes as much as possible. As far as the mean squared residue is concerned, ROB is able to find biclusters of relatively lower average mean squared residue than that of CC and FLOC. FLOC outperforms CC with respect to average mean squared residue as well as average volume of biclusters found.

TABLE II. PERFORMANCE COMPARISON OF ROB, CC AND FLOC

	Avg.mean squared residue	*Avg.volume*	*Avg.gene num.*	*Avg.cond.num*
ROB	159	1759.45	226	7.1
CC	204.293	1576.98	167	12
FLOC	187.543	1825.78	195	12.8

V. CONCLUSION

In this paper, we proposed a suite of biclustering algorithms in framework of generalized rough sets. In this framework, a two-way rough *k*-means clustering method and a rough overlapping biclustering algorithm are developed and applied on gene expression matrix. Our method is able to find a set of biclusters of maximum size, with stronger coherence, and particularly with a reasonable degree of overlapping simultaneously. By associating each bicluster with a lower and an upper approximation, our approach dynamically adjusts the memberships of genes and conditions and temporarily blocks certain adjustment which has a potential to violate the propose criterion. The experimental results confirmed that our ROB approach is capable of finding highly coherent biclusters with a reasonable degree of overlapping effectively.

ACKNOWLEDGMENT

The authors appreciate the availability of microarray data published in the web site [27].

REFERENCES

[1] M.B.Eisen, P.T.Spellman, P.O.Brown, D.Botstein, "Clustering analysis and display of genome-wide expression patterns," In Proc. Natl. Acad. Sci. U S A, 95 (25): 14863-14868,1998.

[2] R.Herwig, A.J.Poustka, C.Muller, C.Bull, H.Lehrach, J. O'Brien, "Large-scale clustering of cDNA-fingerprinting data," Genome Res., 9(11):1093-1105,1999.

[3] P. Tamayo, D.Slonim, J.Mesirov, Q.Zhu, S.Kitareewan, E.Dmitrovsky, E.S.Lander, T.R.Golub, "Interpreting patterns of gene expression with self-organizing maps: Methods and application to hematopoietic differentiation," In Proc. Natl. Acad. Sci. U S A, 96(6):2907-2912,1999.

[4] J.A.Hartigan, "Direct clustering of a data matrix," J. Am.Statistical Assoc.(JASA), 67(337): 123-129,1972.

[5] C. Cano, L. Adarve, J. López, A. Blanco, "Possibilistic approach for biclustering microarray data," Computers in Biology and Medicine, 2007, in press.

[6] Y. Cheng and G.M.Church, " Biclustering of expression data," In Proceedings of the 8th International Conference on Intelligent Systems for Molecular Biology (ISMB00), pp.93-103,2000.

[7] Dhillon S, "Co-clustering documents and words using bipartite spectral graph partitioning," In Proc. of the 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining(KDD01), pp.267-274, 2001.

[8] Mirkin, B, " Mathematical classification and clustering," Dordrecht: Kluwer, 1996.

[9] J. Yang, H.Wang, W.Wang, P.Yu, "Enhanced biclustering on expression data," In Proc. Third IEEE Conf. Bioinformatics and Bioeng., pp.321-327,2003.

[10] G.Getz,E.Levine and E.Domany, "Coupled two-way clustering analysis of gene microarray data," In Proc. Natl. Acad. Sci. U S A, 97(22):12079-12084,2000.

[11] C.Tang, L.Zhang, A.Zhang, and M.Ranmanathan, "Interrelated two-way clustering: an unsupervised approach for gene expression data analysis," In Proc. Second IEEE Int' Symp. Bioinformatics and Bioeng., pp.41-48, 2001.

[12] L.Lazzeroni and A.Owen, "Plaid models for gene expression data," Technical Report,Standford Univ.,2000.

[13] E.Segal, B.Taskar, A. Gasch, N.Friedman and D.Koller, "Rich probabilistic models for gene expression," Bioinformatics, 17(Suppl 1):S243-S252,2001

[14] A.Tanay, R.Sharan. and R.Shamir, "Discovering statistically significant biclusters in gene expression data," Bioinformatics, 18(Suppl 1):136-144,2002.

[15] Y.Kluger, R.Basri, J.T.Chang, and M.Gerstein, "Spectral biclustering of microarray data: coclustering genes and conditions," Genome Research, 13(4):703-716,2003.

[16] Q.Z.Sheng, Y.Moreau and B.D.Moor, "Biclustering microarray data by Gibbs Sampling," Binformatics, 19(Suppl 2):ii196-ii205,2003.

[17] C.J. Wu, Y.Fu, T.M.Murali, S.Kasif, "Gene expression module discovery using Gibbs Sampling," Genome Informatics,15(1): 239-248, 2004.

[18] K.Bryan, P.Cunningham, N.Bolshakova, "Biclustering of expression data simulated annealing," In Proceedings of the 18th IEEE Symposium on Computer-Based Medical Systems(CBMS'05), pp.383-388,2005.

[19] F.Divina and J.S.Aguilar-Ruiz, "Biclustering of expression data with evolutionary computation," IEEE Transaction on Knowledge and Data Engineering, 18(5): 590-602,2006.

[20] Z.Pawlak, "Rough sets," International Journal of Information and Computer Sciences, 11:145-172,1982.

[21] P.Lingras,C.West, "Interval set clustering of web users with rough k-means," J. Intell. Inform. Syst, 23: 5-16, 2004.

[22] G.Peters, "Some refinements of rough k-means clustering," Pattern Recognition, 39: 1481-1491, 2006.

[23] Y.Y.Yao, X.Li, T.Y.Lin, and Q.Liu, "Representation and classification of rough set models," In Proceedings of Third International Workshop on Rough Sets and Soft Computing, pp.630-637, 1994.

[24] W.Zhu, F.Y.Wang, "Reduction and axiomization of covering generalized rough sets," Information Sciences, 152: 217 230,2003.

[25] S. Madeira, A.Oliveria, "Biclustering algorithm for biological data analysis: a survey," IEEE/ACM Trans. Comput.Biol.Bioinformatics, 1:24-45,2004.

[26] S.Tavazoie, J.D.Hughes, M.J.Campbell, R.J.Cho, and G. M.Church, "Systematic determination of genetic network architecture," Nature Genetics, 22:281-285,1999.

[27] http://arep.med.harvard.edu/biclustering

[28] R.Z.Wang, D.Q.Miao, and G.R.Hu, "Discernibility matrix based algorithm for reduction of attributes," In: Workshops Proceedings of 2006 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology, pp.477-480, 2006.