Proceeding of the International Conference on Advanced Science, Engineering and Information Technology 2011

Hotel Equatorial Bangi-Putrajaya, Malaysia, 14 - 15 January 2011

ISBN 978-983-42366-4-9

An Archived Multi Objective Simulated Annealing Method to Discover Biclusters in Microarray Data

Mohsen Lashkargir[#], Mohammad Sadegh Tabatabaeifar[#], Sadegh Taghizadeh[#]

[#] *Department of Computer Engineering, Islamic Azad University, Mehriz branch*

Mehriz, Yazd, Iran

E-mail: mlashkargir@gmail.com

Abstract—**With the advent of microarray technology it has been possible to measure thousands of expression values of genes in a single experiment. Analysis of large scale geonomics data, notably gene expression, has initially focused on clustering methods. Recently, biclustering techniques were proposed for revealing submatrices showing unique patterns. Biclustering or simultaneous clustering of both genes and conditions is challenging particularly for the analysis of high-dimensional gene expression data in information retrieval, knowledge discovery, and data mining. In biclustering of microarray data, several objectives have to be optimized simultaneously and often these objectives are in conflict with each other. A multi objective model is very suitable for solving this problem. Our method proposes a algorithm which is based on multi objective Simulated Annealing for discovering biclusters in gene expression data. Experimental result in bench mark data base present a significant improvement in overlap among biclusters and coverage of elements in gene expression and quality of biclusters.**

Keywords— ***biclustering, multi objective optimization, Simulated Annealing ; gene expression data.***

I. INTRODUCTION

The microarray technique allows measurement of mRNA levels simultaneously for thousands of genes. It is now possible to monitor the expression of thousands of genes in parallel over many experimental conditions (e.g., different patients, tissue types, and growth environments), all within a single experiment. Microarray data constructs a data matrix in which rows represent genes and columns show condition. Each entry in the matrix is shown the expression level of specific gene (g_i) under particular condition (c_i). Thorough analysis of gene expression data the genes are found that represent similar behavior among a subset of condition. Thorough analysis of gene expression data the genes are found that represent similar behavior among a subset of condition.

In [2] was used clustering for analyses of gene expression data but genes didn't show similar behavior in all conditions, while genes show similar behavior in subset of conditions. However the genes are not necessarily related in all conditions, in other words, there are genes that can be relevant in subset of condition. In fact, both of rows and columns (genes and conditions) are clustered and they refer to biclustering (simultaneously clustering of both rows and columns).

The biclustering problem is even more difficult than clustering, as we tried to find clusters using two dimensions, instance of one. The first biclustering useful algorithm was proposed by Cheng and Church [1] in 2000. They introduced the residue of an element in the bicluster and the *mean squared residue* of submatrix for quality measurement of biclusters. This introduced method is a good measurement tool for biclustering and we use this measurement. Yang improved Cheng and Church approach to find K possibly overlapping biclusters simultaneously [3].It is also robust against missing values which are handled by taking into account the bicluster volume (number of non-missing elements) when computing the score.

The biclustering problem is proven to be NP hard [1]. This high complexity motivated the researcher to use stochastic approach to find biclusters. Federico and Aguilar proposed a Biclustering algorithm with Evolutionary computation [4].

In biclustering of gene expression data, the goal is to find bicluster of maximum size with mean squared residue lower than a given δ, which are relatively high row variance. In [4], the fitness function is made by the sum weighted of this objectives function. Since in biclustering problem some objectives exist, that are in conflict with each other, using multi object methods is very suitable to solve that. In [5],we tackled this problem based on multi objective particle swarm.

In this work we address a biclustering problem based on multi objective simulated annealing optimization.

This paper is organized as follows: in section 2, the definitions related to biclustering are presented. An introduction to SA and multi objective SA is given in section 3. The description of the algorithm is illustrated in section 4. Experimental results and comparative analysis are discussed in section 5. The last section is the conclusion.

II. BICLUSTERING

A bicluster is defined on a gene expression matrix. Let $G=\{g_1, \dots, g_N\}$ be a set of genes and $C=\{c_1, \dots, c_M\}$ be a set of conditions .The gene expression matrix is a matrix of real numbers , with possible null values , where each entry e_{ij} corresponds to the logarithm of the relative abundance of the mRNA of gene g_i under a specific condition c_j[4].A bicluster in gene expression data corresponds to the submatrix that genes in that show similar behavior under a subset of conditions. A bicluster is showed by subset of genes and subset of conditions. The similar behavior between genes is measured by *mean squared residue* that was introduced by Cheng and Church.

Definition 1 : Let (I,J) be a bicluster ($I \subseteq G$, $J \subseteq C$) then the mean squared residue (r_{IJ}) of a bicluster (I,J) is calculated as below :

$$r_{IJ} = \frac{1}{|I||J|}\sum_{i\epsilon I, j\epsilon J}\left(e_{ij} - e_{iJ} - e_{Ij} + e_{IJ}\right)^2 \quad (1)$$

Where

$$e_{iJ} = \frac{\sum_{j\epsilon J} e_{ij}}{|J|} \quad (2)$$

$$e_{Ij} = \frac{\sum_{i\epsilon I} e_{ij}}{|I|} \quad (3)$$

$$e_{IJ} = \frac{\sum_{i\epsilon I, j\epsilon J} e_{ij}}{|I||J|} \quad (4)$$

The lower the mean squared residue, the stronger the coherence exhibited by the bicluster and the quality of the bicluster. If a bicluster has a mean squared residue lower than a given value δ , then we call the bicluster a δ–bicluster. In addition to the mean squared residue, the row variance is used to be relatively large to reject trivial bicluster.

Definition 2: Let (I,J) be a biclusters. The row variance of (I,J) is defined as

$$var_{IJ} = \frac{1}{|I||J|}\sum_{i\epsilon I, j\epsilon J}\left(e_{ij} - e_{iJ}\right)^2 \quad (5)$$

Biclusters characterized by high values of row variance contains genes that present large chances in their expression values under different conditions.

III. SIMULATED ANNEALING

Simulated annealing (SA) is one of the most flexible techniques available for solving hard combinatorial problems. The main advantage of SA is that it can be applied to large problems regardless of the conditions of differentiability, continuity, and convexity that are normally required in conventional optimization methods.

SA is a compact and robust technique, which provides excellent solutions to single and multiple objective optimization problems with a substantial reduction in computation time[7].

The method is inspired by the thermodynamic process of cooling (annealing) of molten metals to attain the lowest free energy state Kirkpatrick et al. (1983). When molten metal is cooled slowly enough it tends to solidify in a structure of minimum energy. This annealing process is mimicked by a search strategy. The key principle of the method is to allow occasional worsening moves so that these can eventually help locate the neighborhood to the true (global) minimum.

A. SA for single objective optimization

In a single objective optimization SA start white with a randomly generated initial solution (x) as current solution. Then at each stage new solution (y) is generated using suitable algorithms from current solution. As with a greedy search, SA accepts all changes that lead to improvements in the fitness of a solution. However, it differs in its ability to allow the probabilistic acceptance of changes which lead to worse solutions. The following probability is calculated in performing the acceptance test for minimize f:

$$p_t(Accept\ y) = \begin{cases} 1 & f(y) \le f(x) \\ e^{\frac{f(x)-f(y)}{t}} & f(y) > f(x) \end{cases} \quad (6)$$

In (6) t is the control parameter that corresponds with the temperature in physical annealing. Initially, when T is large, larger deterioration in the cost function is allowed; as the temperature decreases, the simulated annealing algorithm becomes greedier, and only smaller deteriorations are accepted; and finally when T tends to zero, no deteriorations are accepted.

B. SA for multi objective optimization

As mentioned earlier, biclustering is a multi objective problem. In problems with more than one conflicting objective, there exist no single optimum solution rather there exists a set of solutions which are all optimal involving trade-offs between conflicting objective (pareto optimal set).The concept of archiving the Pareto-optimal solutions coupled with return to base strategy has been used by Suppapitnarm et al. (2000) for solving multi objective problems with simulated annealing.

Definition 3: if there are M objective functions, a solution x is said to *dominate* another solution y if the solution x is no worse than y in all the M objective functions and the solution is strictly better than y in at least one of the M objective functions. Otherwise the two solutions are *non-dominating* to each other. This concept is shown in Fig1.

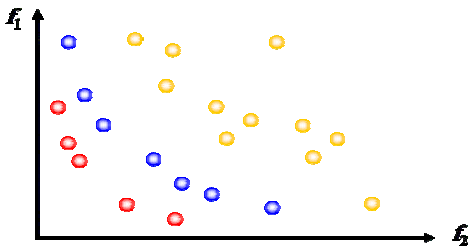

Fig. 1 Example of dominate and non-dominate cocepts (f1 and f2 must be minimaze).Red points dominate blue points and yellow points.Red points are non-diminated each other.

Definition 3: if there are M objective functions, a solution x is said to dominate another solution y if the solution x is no worse than y in all the M objective functions and the solution x is strictly better than y in at least one of the M objective functions. Otherwise the two solutions are non-dominating to each other. This concept is shown in Fig1.

Definition 4: If Z is subset of feasible solutions, a solution x∈Z is said to *non-dominate* with respect to Z if there does not exist another solution y∈Z that y dominates z (Red point in Fig1).

Definition 5: If F is a set of feasible solutions, a solution x∈F is said to be *pareto-optimal*, if x is *non-dominate* with respect to F (Red point in Fig1 if we suppose all feasible solutions are shown in Fig1).

In multi objective optimization problem, we determine the pareto optimal set from the set of feasible solutions. In this problem, we must consider the diversity among solutions in pareto set.

In this article we use Archived multi objective simulated annealing (AMOSA) provided by S.bandyopadhyay et al [8] with modification. In AMOSA nondominated solutions are stored in Archive. The algorithm begin with the initialize number of solutions. The nundominated of them add to Archive. AMOSA uses the concept of amount of domination in computing the acceptance probability of a new solution. Given two solutions a and b, the amount of domination is defined as

$$\Delta dom_{a,b} = \prod_{i=1, f_i(a)\neq f_i(b)}^{M} \frac{|f_i(a)-f_i(b)|}{R_i} \quad (7)$$

Where M= number of objectives and R_i is the range of the ith objective. Note that in several cases, R_i may not be known a priori. In these situations, the solutions present in the Archive along with the new and the current solutions are used for computing it[8].

One of the points, called current-pt, is randomly selected from Archive as the initial solution at temperature temp=Tmax. The current-pt is perturbed to generate a new solution called new-pt. The domination status of new-pt is checked with respect to the current-pt and solutions in Archive. Based on the domination status between current-pt and newpt three different cases may arise. These are enumerated below.

Case 1: current-pt dominates the new-pt and k (k ≥0) points from the Archive dominate the new-pt.

In this case, the new-pt is selected as the current-pt with

$$probability = \frac{1}{1+exp(\Delta dom_{avg} * temp)} \quad (8)$$

Where

$$\Delta dom_{avg} = \left(\left(\sum_{i=1}^{k} \Delta dom_{i,new-pt}\right) + \Delta dom_{current-pt,new-p}\right)$$

Note that Δdom_{avg} denotes the average amount of domination of the new-pt by (k + 1) points, namely, the current-pt and k points of the Archive. Also, as k increases, Δdom_{avg} will increase since here the dominating points that are farther away from the new-pt are contributing to its value.

Case 2: current-pt and new-pt are non-dominating with respect to each other. Based on the domination status of new-pt and members of Archive, the following three situations may arise.

1) new-pt is dominated by k points in the Archive where k≥ 1.The new-pt is selected as the current-pt with

$$probability = \frac{1}{(1+exp(\Delta dom_{avg} * temp))} \quad (9)$$

Where

$$\Delta dom_{avg} = \sum_{i=1}^{k} (\Delta dom_{i,new-pt})/k$$

2) *new-pt* is non-dominating with respect to the other points in the *Archive* as well. In this case the *new-pt* is on the same front as the *Archive*. So the *new-pt* is selected as the *current-pt* and added to the *Archive*.

3) *new-pt* dominates *k* (k ≥ 1) points of the *Archive*.. Again, the *new-pt* is selected as the *current-pt*, and added to the *Archive*. All the *k* dominated points are removed from the *Archive*. Note that here too the *current-pt* may or may not be on the archival front.

Case 3: *new-pt* dominates *current-pt*

Now, based on the domination status of *new-pt* and members of *Archive*, the following three situations may arise.

1) *new-pt* dominates the *current-pt* but *k* (k ≥ 1) points in the *Archive* dominate this *new-pt*. Here, the minimum of the difference of domination amounts between the *new-pt* and the *k* points, denoted by Δdom_{min} , of the *Archive* is computed. The point from the *Archive* which corresponds to the minimum difference is selected as the *current-pt* with

$$probability = \frac{1}{1+\exp(-\Delta dom_{min})} \quad (10)$$

Otherwise the *new-pt* is selected as the *current-pt*.

2) *new-pt* is non-dominating with respect to the points in the *Archive* except the *current-pt* if it belongs to the *Archive*. Thus *new-pt*, which is now accepted as the *current-pt*, can be considered as a new nondominated solution that must be stored in *Archive*. Hence *new-pt* is added to the *Archive*. If the *current-pt* is in the *Archive*, then it is removed.

3) *new-pt* also dominates *k* (k ≥ 1), other points, in the *Archive*. Hence, the *new-pt* is selected as the *current-pt* and added to the *Archive*, while all the dominated points of the *Archive* are removed.[8]

IV. Our Multi Objective Simulated Annealing Biclustering Method

```
Initialize the Archive
Current -pt = random(Archive) /* randomly chosen solution from Archive */
While (temp > Tmin)
    For (i=0 ; i < itr ; i++)
        New-pt=Generate_solution(current -pt)
            if (current–pt dominates new-pt) /* case 1 */
                    Set new-pt as current-pt with probability (8)
             if (current-pt and new-pt are non-dominating to each other) /* case 2 */
                    if (new-pt is dominated by k (k≥1)points in the Archive) /* case 2(a) */
                            Set new-pt as current-pt with probability (9)
                      if (new-pt is non-dominating w.r.t all the points in the Archive) /* case 2(b) */
                            set new-pt as current-pt and add new-pt to the Archive.
                            If ( |Archive| > SL)
                                Diversity () // using crowding distance removing |Archive|-HM item from archive
                      if (new-pt dominates k, (k≥1) points of the Archive) /* case 2(c) */
                            set new-pt as current-pt and add it to the Archive.
                             Remove all the k dominated points from the Archive.
                if (new-pt dominates current-pt) /* case 3 */
                        check the domination status of new-pt and points in the Archive.
                        if (new-pt is dominated by k (k ≥ 1) points in the Archive) /* case 3(a) */
                            Set point of the archive which corresponds to Δdom_min as current-pt with probability (11)
                            Else set new-pt as current-pt.
                    if (new-pt is non-dominating with respect to the points in the Archive)/*case3(b)*/
                            set new-pt as current-pt and add it to the Archive
                            if current-pt is in the Archive, remove it from the Archive
                            else if Archive-size > SL
                                comput crowding distance remove SL –ML
                        if (new-pt dominates k other points in the Archive) /* case 3(c) */
                            set new-pt as current-pt and add it to the Archive
                            remove all the k dominated points from the Archive
        End for
 Remove overlap from Archive until Archive size = HL
End while
If Archive-size > SL
    Remove overlap from Archive until Archive size = HL
```

Fig.2 A general scheme of our algorithm

Our goal is to find biclusters G(I, J) (I is subset of genes, J is subset of conditions) of maximum size, with mean squared residue lower than a given δ , with a relatively high row variance, and with a low level of overlapping among biclusters.

The size of bicluster is defined as |I|*|J| if we use this definition as an objective since the number of rows is higher than the number of columns , columns have less effect in objective. So we separate rows and columns and consider two objective functions one for rows and one for columns.

Problem is formulated as below:

Find G(I,J)

That minimize

$$f_1(I,J) = \frac{|G|}{|I|} \tag{11}$$

$$f_2(I,J) = \frac{|C|}{|J|} \tag{12}$$

$$f_3(I,J) = \frac{r_{IJ}}{\delta} \tag{13}$$

$$f_4(I,J) = \frac{1}{var(I,J)} \tag{14}$$

In the AMOSA [8] clustering is used for maintain diversity in Archive but in our problem we can't use clustering for solution stored in Archive because in practical Data Base solution are huge and clustering is not useful in our problem except diversity in Archive we have another challenge and it is overlapping. overlapping among biclusters stored in Archive must be minimum. in [8] tow limits are kept in Archive size but in this article for maintain diversity and decrease overlapping among solution in Archive three limits used in Archive size: a hard limit denoted by HL and medium limit denoted by ML and soft limit denoted by SL. For maintain diversity crowding distance that is provided by Deb[10] is used.

After add new solution to Archive if the size of Archive is greater than SL crowding distance between element in Archive are computed according to[10] and then (SL - ML) element in Archive are selected to remove according to diversity. We use roulette wheel to do this selection.

A. Archive Initialization

We use cheng and church algorithm to create a set of solution at the first. Then nondominated solution from that set are inserted into archive.

B. Generate new solution

The generation method provided by [11] is used the current solution, is then iteratively perturbed by deletion and addition of rows or columns in the input matrix. In this method the authors take into account the ratio of rows to columns in the current solution and adjusts the probability of a row or column flip accordingly. So, for example, if there are 100 columns and 10 rows in a current solution, the probability of choosing a row to flip is 1/10.

V. EXPERIMENTAL RESULT

The proposed biclustering algorithm is implemented in matlab and applied to mine biclusters from two well know data set. The first data set is the *yeast saccharomy cerevisiae cell cycle* expression [1] .The expression matrix contained in this data set consists of 2884 genes and 17 experimental conditions. All entries are integers lying in the range of 0-600. The second data set, the *human B-cells* expression data, is a collection of 4,026 genes and 96 conditions the values of δ for the two data sets are taken from [1]. For the yeast data δ=300 and for the human B-cells expression data δ=1200.

A. result on yeast data set

Our method is applied to mining fifty biclusters from yeast data set simultaneously this biclusters cover 92% of the genes , 100% of the condition and 82.3% cells of the expression matrix while the MOPSOB[7] and AMOPSO[5] method cover 73.1% and 91.3 %of genes and 52.4% and 79% cells of the expression data respectively. In table 1 information about five out of fifty biclusters are summarized.

In order to show the performance of our method, we compare it with other multi objective biclustering method. In [6][9][5] four multi objective biclustering are proposed, we summarize their result, and our result in table 2.

TABLE I. YEAST BICLUSTERS

Bicluster	Genes	Conditions	Residue	Row variance
1	433	17	239.65	587.23
16	1127	14	247.87	774.32
24	1003	12	238.45	863.87
39	793	15	223.04	723.65
47	1367	9	248.73	890.12

TABLE II. COMPARATIVE WITH OTHER METHOD FOR YEAST DATASE

Method	Avg size	Avg residue	Avg genes	Avg condition	Max size
NAGA 2	10301.17	234.87	1095.43	9.29	14828
SEEA 2 B	8547.21	287.56	785.42	8.92	10503
MOPSOB	10510	218.54	1102.84	9.31	15613
AMOPSOB	11065	238.32	12981.5	11.66	14027
Our method	11245	227.31	12894.7	11.84	15778

B. result on human data set

Our method is applied to mining one hundred biclusters from human data set too. This biclusters cover 58% of the genes , 100% of the condition and 45.2% cells of the expression matrix while the MOPOB[7] and AMOPSO[5] method cover 46.7% and 53.6% of genes and 35.7% and 41.6% cells of the expression data respectively. In table 3 information about five out of one hundred biclusters are summarized and a comparative study is expressed in table 4.

TABLE III. HUMAN BICLUSTERS

Bicluster	Genes	Conditions	Residue	Row variance
1	903	44	997.68	3678.55
32	105	93	1007.41	7628.44
53	1072	35	957.58	2572.78
77	620	35	841.25	6421.36
89	503	28	784.53	7745.24

TABLE IV. COMPARATIVE WITH OTHER METHOD

Method	Avg size	Avg residue	Avg genes	Avg condition	Max size
NAGA 2	33463.70	987.56	915.81	36.54	37560
SEEA 2 B	29874.8	1128.1	784.68	35.48	29654
MOPSOB	34012.24	927.47	902.41	40.12	37666
AMOPSOB	33987.51	941.32	1009.75	42.11	37908
Our method	34125	928.36	987.65	43.57	39732

VI. CONCLUSIONS

In this paper, we introduced an algorithm based on adaptive multi objective simulated annealing for finding biclusters on expression data. In biclustering problem several objective have to be optimized simultaneously. We must find maximum biclusters with lower mean score residue and high row variance. These three objectives are in conflict with each other. We use crowding distance for maintain diversity. In addition we consider a low level of overlap among biclusters by using archive with variable size. A comparative assessment of results is provided on bench mark gene expression data set to demonstrate the effectiveness of the proposed method. Experimental results show that proposed method is able to find interesting biclusters on expression data and comparative analysis show better performance in result.

REFERENCES

[1] Y. Cheng and G.M. Church,"Biclustering of Expression Data," Proc. Eighth Int'l Conf. Intelligent Systems for Molecular Biology,pp. 93-103, 2000.

[2] A. Ben-Dor, R. Shamir, and Z. Yakhini, "Clustering Gene Expression Patterns," J. Computational Biology, vol. 6, nos. 3-4, pp. 281-297, 1999.

[3] 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.

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

[5] M.Lashkargir, S.Monadjemi, A.Baraani, "A New Biclustering Method for Gene Expersion Data Based on Adaptive Multi Objective Particle Swarm Optimization," iccee, vol. 1, pp.559-563, 2009 Second International Conference on Computer and Electrical Engineering, 2009

[6] S. Mitra and H. Banka, "Multi-objective evolutionary biclustering of gene expression data," *Pattern Recognition*,vol. 39, no. 12, 2006, pp. 2464-2477.

[7] B.Suman1 and P.Kumar, " A survey of simulated annealing as a tool for single and multiobjective optimization," Journal of the Operational Research Society (2006) 57, pp. 1143–1160.2006

[8] S.Bandyopadhyay, S.Saha, U.Maulik and K.Deb, "A Simulated Annealing Based Multi-objective Optimization Algorithm: AMOSA," Evolutionary Computation, IEEE Transactions on., Vol. 12,pp. 269 – 283.,2008

[9] J.Liu, Z.Li, F.Liu and Y. Chen "Multi-Objective Particle Swarm Optimization Biclustering of Microarray Data, " IEEE International Conference on Bioinformatics and Biomedicine, pp.363-366,2008

[10] K. Deb, S. Agarwal, A. Pratap, and T. Meyarivan, "A fast and elitist multi-objective genetic algorithm : NSGA-II," IEEE Transactions on Evolutionary Computation, vol. 6, pp. 182–197, 2002.

[11] K.Bryan, P. Cunningham, and N.Bolshakova, , "Application of Simulated Annealing to the Biclustering of Gene Expression Data," IEEE Transactions on Information Technology in Biomedicine, VOL. 10, NO. 3,pp.519-525.2006.