Improving Multi-objective Clustering through Support Vector Machine: Application to Gene Expression Data

Anirban Mukhopadhyay, Ujjwal Maulik, Senior Member, IEEE
and Sanghamitra Bandyopadhyay, Senior Member, IEEE

Abstract—Microarray technology facilitates the monitoring of the expression profile of a large number of genes across different experimental conditions simultaneously. This article proposes a novel approach that combines a recently proposed multiobjective fuzzy clustering scheme with support vector machine (SVM), to yield improved solutions. The multiobjective technique is first used to produce a set of non-dominated solutions. The non-dominated set is then used to find some high-confidence points using a fuzzy voting technique. The SVM classifier is trained by this high-confidence points. Finally the remaining points are classified using the trained classifier. Results demonstrating the effectiveness of the proposed technique are provided for three real life gene expression data sets. Moreover statistical significance test has been conducted to establish the significant superiority of the proposed technique.

Index Terms—Fuzzy clustering, Support Vector Machine, multiobjective optimization, Pareto-optimality, cluster validity measures, microarray gene expression data.

I. INTRODUCTION

With the progress in the field of microarray technology it is now possible to study of the expression levels of large number of genes across different experimental conditions simultaneously. Microarray technology has its application in the areas of medical diagnosis, bio-medicine, gene expression profiling, etc [1], [2], [3], [4]. Usually, the gene expression values during some biological experiment are measured along time series. A microarray gene expression data consisting of $g$ genes and $h$ time points are typically organized in a 2D matrix $E = [e_{ij}]$ of size $g \times h$. Each element $e_{ij}$ gives the expression level of the $i$th gene at the $j$th time point. Clustering [5], an important microarray analysis tool, is used to identify the sets of genes with similar expression profiles. Clustering [5], [6] methods partition a set of $n$ objects into $K$ groups based on some similarity/dissimilarity metric where the value of $K$ may or may not be known a priori. Unlike hard clustering, a fuzzy clustering algorithm produces a $K \times n$ membership matrix $U(X) = [u_{kj}]$, $k = 1, \ldots, K$ and $j = 1, \ldots, n$, where $u_{kj}$ denotes the probability of assigning pattern $x_j$ to cluster $C_k$. For probabilistic non-degenerate clustering $0 < u_{kj} < 1$ and $\sum_{k=1}^{K} u_{kj} = 1, 1 \leq j \leq n$ [7].

Genetic algorithms [8] have been effectively used to develop efficient clustering techniques [9], [10], [11]. Recently, a real-coded centroid based multiobjective fuzzy genetic clustering technique has been proposed in [12] that optimizes Xie-Beni (XB) index [13] and the Fuzzy C-means (FCM) [7] measure ($J_m$) simultaneously. In multiobjective optimization (MOO) [12], [14], [15], [16], [17], [18], search is performed over a number of, often conflicting, objective functions. In contrast to single objective optimization which yields a single best solution, in MOO the final solution set contains a number of Pareto-optimal solutions, none of which can be further improved on any one objective without degrading it in another. As the final solution set consists of a number of non-dominated solutions, it is important to obtain a final solution from the set of near Pareto-optimal solutions. This article describes a novel method to obtain the final solution from the set of near Pareto-optimal solutions. The approach is to combine the multiobjective clustering technique with support vector machine (SVM) [19] based classifier to obtain the final solution from the set of Pareto-optimal solutions. The procedure involves in utilizing the points which are given a high membership degree to a particular class by a majority of the non-dominated solutions. These points are taken as the training points to train the SVM classifier. The rest of the points are then classified by the trained SVM classifier to yield the class labels for these points.

The performance of the Multiobjective GA (MOGA) clustering followed by SVM classification (MOGA-SVM) has been demonstrated on three real life gene expression data sets, viz., Yeast Cell Cycle, Arabidopsis Thaliana and Human Fibroblasts Serum data. The superiority of the proposed technique, as compared to original MOGA clustering [12], the well known FCM algorithm[7], single objective GA [9], hierarchical average linkage clustering and Self Organizing Map (SOM) clustering [24] is demonstrated both quantitatively and visually. Furthermore, the superiority of the MOGA-SVM clustering technique has been proved to be statistically significant through statistical significance tests.

The rest of the article is organized as follows: The next section describes the terms related to multiobjective optimiza-
tion. Section III describes the multiobjective fuzzy clustering technique. In Section IV, a brief overview of the SVM classifier is given. Section V presents the proposed MOGA-SVM method. In Section VI, experimental results are demonstrated. Section VII describes the statistical significance test. Finally Section VIII concludes the article.

II. MULTIOBJECTIVE OPTIMIZATION

In many real world situations there may be several objectives that must be optimized simultaneously in order to solve a certain problem. This is in contrast to the problems tackled by conventional GAs which involve optimization of just a single criterion. The main difficulty in considering multiobjective optimization is that there is no accepted definition of optimum in this case, and therefore it is difficult to compare one solution with another one. In general, these problems admit multiple solutions, each of which is considered acceptable and equivalent when the relative importance of the objectives is unknown. The best solution is subjective and depends on the need of the designer or decision maker.

Traditional search and optimization methods such as gradient descent search, and other non-conventional ones such as simulated annealing are difficult to extend as it is to multiobjective case, since their basic design precludes the consideration of multiple solutions. On the contrary, population based methods like evolutionary algorithms are well suited for handling such situations. The multiobjective optimization can formally be stated as [15]: Find the vector \( \mathbf{x}^* = [x_1^*, x_2^*, \ldots, x_n^*]^T \) of decision variables which will satisfy a number of equality and inequality constraints and optimizes the vector function

\[
\mathbf{f}(\mathbf{x}) = [f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_k(\mathbf{x})]^T.
\]

The constraints define the feasible region \( \mathcal{F} \) which contains all the admissible solutions. Any solution outside this region is inadmissible since it violates one or more constraints. The vector \( \mathbf{x}^* \) denotes an optimal solution in \( \mathcal{F} \). The concept of Pareto optimality is useful in the domain of multiobjective optimization. A formal definition of Pareto optimality from the viewpoint of the minimization problem may be given as follows: A decision vector \( \mathbf{x}^* \) is called Pareto optimal if and only if there is no \( \mathbf{x} \) that dominates \( \mathbf{x}^* \), i.e., there is no \( \mathbf{x} \) such that \( \forall i \in \{1, 2, \ldots, k\}, f_i(\mathbf{x}) \leq f_i(\mathbf{x}^*), \exists i \in \{1, 2, \ldots, k\}, f_i(\mathbf{x}) < f_i(\mathbf{x}^*) \). In words, \( \mathbf{x}^* \) is Pareto optimal if there exists no feasible vector \( \mathbf{x} \) which causes a reduction on some criterion without a simultaneous increase in at least one other. In general, Pareto optimum usually admits a set of solutions called non-dominated solutions.

There are a number of multiobjective optimization techniques available. Among them, the GA based techniques such as NSGA-II [14], SPEA [20] and SPEA2 [21] are very popular. The multiobjective fuzzy clustering scheme [12] considered here uses NSGA-II as an underlying multiobjective framework for developing the proposed fuzzy clustering algorithm.

III. MULTIOBJECTIVE FUZZY CLUSTERING

This section briefly describes the NSGA-II based multiobjective fuzzy clustering scheme as proposed in [12]. The algorithm uses real valued chromosomes that denote the coordinates of the cluster centers and each has length \( K \times d \), where \( K \) is the number of clusters and \( d \) is dimension of the data. Each chromosome in the initial population consists of the co-ordinates of \( K \) random points from the data set.

Two cluster validity indices, Xie-Beni (\( XB \)) [13] and fuzzy C-means (FCM) measure (\( J_m \)) [7] are simultaneously optimized. For computing the objective functions, first the centers \( V = \{v_1, v_2, \ldots, v_K\} \) encoded in a given chromosome are extracted. The fuzzy membership values \( u_{ik}, i = 1, 2, \ldots, K, k = 1, 2, \ldots, n \) are computed using the following equation [7]:

\[
u_{ik} = \frac{1}{\sum_{j=1}^{K} \left(\frac{D(v_i, x_k)}{D(v_j, x_k)}\right)^{2/m}}, \quad 1 \leq i \leq K; \quad 1 \leq k \leq n,
\]

where \( D(v_i, x_k) \) denotes the distance between \( i \)th cluster center and \( k \)th data point and \( m \in \{1, \infty\} \) is the fuzzy exponent. In this article, the Euclidean distance measure is used. Subsequently each cluster center \( v_i, i = 1, 2, \ldots, K, \) is updated using the following equation [7]:

\[
v_i = \frac{\sum_{k=1}^{n} (u_{ik})^m x_k}{\sum_{k=1}^{n} (u_{ik})^m}, \quad 1 \leq i \leq K.
\]

The membership values are then recomputed using Eq. (2). The \( XB \) index is defined as a function of the ratio of the total variation \( \sigma(\sigma(U, V; X) = \sum_{i=1}^{n} \sum_{k=1}^{K} u_{ik}^2 D^2(v_i, x_k)) \) to the minimum separation \( sep \) (\( sep(V) = \min_{i \neq j} \{D^2(v_i, v_j)\} \)) of the clusters. Here \( \sigma \) and \( sep \) can be written as: The \( XB \) index is then written as [13]:

\[
XB(U, V; X) = \frac{\sigma(U, V; X)}{n \times sep(V)}.
\]

Note that when the partitioning is compact and the clusters are well separated, value of \( \sigma \) should be low while \( sep \) should be high, thereby yielding lower values of the \( XB \) index. The objective is therefore to minimize it.

The other objective is the \( J_m \) validity measure that is optimized by the FCM algorithm. This computes the global fuzzy variance of the clusters and this is expressed by the following equation [7]:

\[
J_m = \sum_{j=1}^{n} \sum_{k=1}^{c} u_{kj}^m D^2(v_k, x_j).
\]

\( J_m \) is to be minimized to get compact clusters.

\( XB \) and \( J_m \) indices are used as they are contradictory in nature. \( XB \) index is responsible for both compactness and separation for the clusters, whereas \( J_m \) only represents the global compactness of the clusters.

Crowded binary tournament selection followed by conventional crossover and mutation operators is used here. NSGA-II uses the elitist model where non-dominated solutions of the parent and child populations are propagated to the next generation in order to keep track of the best solutions obtained.
so far. The algorithm has been run for a fixed number of generations. It produces a set of non-dominated solutions at the last generation.

IV. SUPPORT VECTOR MACHINE

Support vector machine (SVM) classifiers are inspired by statistical learning theory and they perform structural risk minimization on a nested set structure of separating hyperplanes [19], [22]. Viewing the input data as two sets of vectors in a d-dimensional space, an SVM constructs a separating hyperplane in that space, one which maximizes the margin between the two classes of points. To compute the margin, two parallel hyperplanes are constructed on each side of the separating one, which are “pushed up against” the two classes of points. Intuitively, a good separation is achieved by the hyperplane that has the largest distance to the neighboring data points of both classes. The larger the margin or distance between these parallel hyperplanes indicates better generalization error of the classifier. Fundamentally the SVM classifier is designed for two-class problems. It can be extended to handle multi-class problems by designing a number of one-against-all or one-against-one two-class SVMs.

Suppose a data set consists of \( n \) feature vectors \( < x_i, y_i > \), where \( y_i \in \{ +1, -1 \} \), denotes the class label for the data point \( x_i \). The problem of finding the weight vector \( w \) can be formulated as minimizing the following function:

\[
L(w) = \frac{1}{2}||w||^2,
\]

subject to

\[
y_i[w, \phi(x_i) + b] \geq 1, i = 1, \ldots, n.
\]

Here, \( b \) is the bias and the function \( \phi(x) \) maps the input vector to the feature vector. The dual formulation is given by maximizing the following:

\[
Q(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} y_i y_j \alpha_i \alpha_j K(x_i, x_j),
\]

subject to

\[
\sum_{i=1}^{n} y_i \alpha_i = 0 \quad \text{and} \quad 0 \leq \alpha_i \leq C, i = 1, \ldots, n.
\]

Only a small fraction of the \( \alpha_i \) coefficients are nonzero. The corresponding pairs of \( x_i \) entries are known as support vectors and they fully define the decision function. Geometrically, the support vectors are the points lying near the separating hyperplane, \( K(x_i, x_j) = \phi(x_i), \phi(x_j) \) is called the kernel function. The kernel function may be linear or non-linear, like polynomial, sigmoidal, radial basis functions (RBF), etc. RBF kernels are of the following form:

\[
K(x_i, x_j) = e^{-w ||x_i - x_j||^2},
\]

where \( x_i \) denotes the \( i \)th data point and \( w \) is the weight. In this article, above mentioned RBF kernel is used. Also, the extended version of the two-class SVM that deals with multi-class classification problem by designing a number of one-against-all two-class SVMs, is used here. For example, a \( K \)-class problem is handled with \( K \) two-class SVMs.

V. PROPOSED MOGA-SVM CLUSTERING

This section describes the proposed scheme for combining the multiobjective fuzzy clustering algorithm (MOGA) described in section III with the SVM classifier. The combined approach is named as MOGA-SVM. This approach gives equal importance to each non-dominated solution. A fuzzy majority voting technique is applied. The motivation is that the points that are assigned to a cluster with high membership degree by most of the non-dominated solutions can be considered as they are clustered properly and thus can be used to train the classifier. The remaining low-confidence points can thereafter be classified using the trained classifier. The steps are described below.

Step 1: Apply MOGA clustering on the given data set to obtain a set \( S = \{ s_1, s_2, \ldots, s_N \} \), \( N \leq P \), \( P \) is the population size of non-dominated solution strings consisting of cluster centers.

Step 2: Using Eq. (2), compute the fuzzy membership matrix \( U^{(i)} \) for each of the non-dominated solutions \( s_i \), \( 1 \leq i \leq N \).

Step 3: Reorganize membership matrices to make them consistent with each other, i.e., cluster \( j \) in the first solution should be equivalent to cluster \( j \) in all other solutions. For example, the solution string \( \{ (p, q, r), (a, b, c) \} \) is equivalent to \( \{ (a, b, c), (p, q, r) \} \).

Step 4: Mark the points whose maximum membership degree (to some cluster \( j \)) is greater than \( \alpha \), for at least \( \beta N \) solutions, as training points. The class labels of the points is class \( j \).

Step 5: Train the SVM classifier by the selected training points.

Step 6: Predict the class labels for the remaining points (test points) using the trained SVM classifier.

Step 7: Combine the label vectors corresponding to training and testing points to obtain the label vector \( \lambda \) for the complete data set.

VI. EXPERIMENTAL RESULTS

The performance of the proposed MOGA clustering has been compared with that of MOGA clustering [16], fuzzy C-means (FCM) [7], hierarchical average linkage method [23], single objective genetic clustering scheme which minimizes XB validity measure (SGA) [9] and Self Organizing Map (SOM) clustering [24]. The results of the conducted experiments on three gene expression data sets, viz., Yeast Cell Cycle data, Arabidopsis Thaliana data and Human Fibroblasts Serum data have been provided.
A. Data Sets and Preprocessing

1) Yeast Cell Cycle: The yeast cell cycle dataset was extracted from a dataset that shows the fluctuation of expression levels of approximately 6000 genes over two cell cycles (17 time points). Out of these 6000 genes, 384 genes have been selected to be cell-cycle regulated [25]. This data set is publicly available at http://faculty.washington.edu/kayee/cluster.

2) Arabidopsis Thaliana: This data set consists of expression levels of 138 genes of Arabidopsis Thaliana. The data contains expression levels of the genes over 8 time points viz., 15 min, 30 min, 60 min, 90 min, 3 hours, 6 hours, 9 hours, and 24 hours [26]. It is available at http://homes.esat.kuleuven.be/thijs/Work/Clustering.html.

3) Human Fibroblasts Serum: This dataset contains the expression levels of 8613 human genes. The data set is obtained as follows: First, human fibroblasts were deprived of serum for 48 hours and then stimulated by addition of serum. After the stimulation, expression levels of the genes were computed over twelve time points and an additional data point was obtained from a separate unsynchronized sample. Hence the data set has 13 dimensions. A subset of 517 genes whose expression levels changed substantially across the time points have been chosen [27]. The data is then log2-transformed. This data set can be downloaded from http://www.sciencemag.org/feature/data/984559.shl.

B. Performance Metrics

For evaluating the performance of the clustering algorithms silhouette index [28] is used. Moreover, two cluster visualization tools namely Eisen plot and cluster profile plot have been utilized.

1) Silhouette Index: Silhouette index [28] is a cluster validity index that is used to judge the quality of any clustering solution $C$. Suppose $a$ represents the average distance of a point from the other points of the cluster to which the point is assigned, and $b$ represents the minimum of the average distances of the point from the points of the other clusters. Now the silhouette width $s$ of the point is defined as:

$$s = \frac{b - a}{\max\{a, b\}}.$$  

(11)

silhouette index $s(C)$ is the average silhouette width of all the data points (genes) and it reflects the compactness and separation of clusters. The value of silhouette index varies from -1 to 1 and higher value indicates better clustering result.

2) Eisen Plot: In Eisen plot [3], (see Fig. 1(a) for example) the expression value of a gene at a specific time point is represented by coloring the corresponding cell of the data matrix with a color similar to the original color of its spot on the microarray. The shades of red color represent higher expression level, the shades of green color represent low expression level and the colors towards black represent absence of differential expression values. In our representation, the genes are ordered before plotting so that the genes that belong to the same cluster are placed one after another. The cluster boundaries are identified by white colored blank rows.

C. Input Parameters

The different parameters of MOGA and single objective GA are taken as follows: number of generations = 100, population size = 50, crossover probability = 0.8 and mutation probability = 0.01. The fuzzy exponent $m$ is taken to be 2. The values of both the parameters $\alpha$ and $\beta$ are taken as 0.5. The fuzzy C-means algorithm has been run for 200 iterations unless it converges before that.

D. Results

Table I reports the average $s(C)$ index values for all the algorithms over 20 consecutive runs for the real life data sets. The number of clusters considered for the Yeast, Arabidopsis and Serum data sets are 5, 4 and 6, respectively as per the available literature [16], [25], [29], [30]. The values reported in the tables indicate that for all the real life data sets, MOGA-SVM provides the best silhouette index scores. It is also evident that the results get improved with the application of SVM classification on MOGA.

To demonstrate visually the result of MOGA-SVM clustering, Figs. 1-3 show the Eisen plot and cluster profile plots corresponding to results (in terms of silhouette index) provided by MOGA-SVM on Yeast, Arabidopsis, Serum and Rat CNS data sets, respectively (Note that due to space limitation, the visual results for the other algorithms are not shown). For example, the 5 clusters of the Yeast data are very prominent as shown in the Eisen plot (Fig. 1(a)). It is evident from the figure that the expression profiles of the genes of a cluster is similar to each other and they produce similar color patterns. The cluster profile plots (Fig. 1(b)) also demonstrate how the expression profiles for the different groups of genes differ from each other, while the profiles within a group are reasonably similar. Similar results are obtained for the other three real data sets also.

VII. Statistical Significance Test

As discussed in the previous section, the results indicate significant improvement in clustering performance using the
proposed clustering approach compared to the other algorithms. A statistical significance test has been carried out next to establish that the superior results obtained by MOGA-SVM are statistically significant. In this article, a statistical significance test called t-test [31] has been carried out at the 5% significance level, to compare the average s(C) index scores produced by different algorithms, six groups, corresponding to the six algorithms (MOGA-SVM, MOGA, FCM, SGA, Average Linkage and SOM), have been created for each of the data sets considered here. Each group consists of s(C) index scores produced by 20 consecutive runs of the corresponding algorithm. Two groups are compared at a time, one corresponding to the MOGA-SVM algorithm and the other corresponding to some other algorithm considered in this article.

Tables II-IV report the results of the t-test for the simulated and real life gene expression data sets. The null hypothesis (The means of two groups are equal, \( H_0 : \mu_1 = \mu_2 \)) are shown in the tables. The alternative hypothesis is that the mean of the first group is larger than the mean of the second group (\( H_1 : \mu_1 > \mu_2 \)). For each test, the degree of freedom is \( M + N - 2 \), where \( M \) and \( N \) are the sizes of two groups considered. Here \( M = N = 20 \). Hence the degree of freedom is 38. Also the values of t-statistic and the probability (P-value) of accepting the null hypothesis are shown in the tables. It is clear from the tables that the P-values are much less than 0.05 (5% significance level) which are strong evidences for rejecting the null hypothesis. This proves that the better average s(C) index values produced by the MOGA-SVM clustering is statistically significant and has not come by chance.

### Table II

<table>
<thead>
<tr>
<th>Test #</th>
<th>Null hypothesis ( H_0 : \mu_1 = \mu_2 )</th>
<th>t-test statistic</th>
<th>P-value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>( \mu_{\text{MOGA-SVM}} = \mu_{\text{MOGA}} )</td>
<td>2.5409</td>
<td>3.19E-06</td>
</tr>
<tr>
<td>2</td>
<td>( \mu_{\text{MOGA-SVM}} = \mu_{\text{FCM}} )</td>
<td>21.0522</td>
<td>5.97E-35</td>
</tr>
<tr>
<td>3</td>
<td>( \mu_{\text{MOGA-SVM}} = \mu_{\text{SGA}} )</td>
<td>16.5938</td>
<td>3.95E-19</td>
</tr>
<tr>
<td>4</td>
<td>( \mu_{\text{MOGA-SVM}} = \mu_{\text{Avglink}} )</td>
<td>6.7547</td>
<td>2.93E-08</td>
</tr>
<tr>
<td>5</td>
<td>( \mu_{\text{MOGA-SVM}} = \mu_{\text{SOM}} )</td>
<td>21.7966</td>
<td>2.27E-23</td>
</tr>
</tbody>
</table>

### Table III

<table>
<thead>
<tr>
<th>Test #</th>
<th>Null hypothesis ( H_0 : \mu_1 = \mu_2 )</th>
<th>t-test statistic</th>
<th>P-value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>( \mu_{\text{MOGA-SVM}} = \mu_{\text{MOGA}} )</td>
<td>15.3371</td>
<td>2.56E-18</td>
</tr>
<tr>
<td>2</td>
<td>( \mu_{\text{MOGA-SVM}} = \mu_{\text{FCM}} )</td>
<td>27.6227</td>
<td>4.62E-27</td>
</tr>
<tr>
<td>3</td>
<td>( \mu_{\text{MOGA-SVM}} = \mu_{\text{SGA}} )</td>
<td>24.2561</td>
<td>4.95E-25</td>
</tr>
<tr>
<td>4</td>
<td>( \mu_{\text{MOGA-SVM}} = \mu_{\text{Avglink}} )</td>
<td>620.6756</td>
<td>4.94E-78</td>
</tr>
<tr>
<td>5</td>
<td>( \mu_{\text{MOGA-SVM}} = \mu_{\text{SOM}} )</td>
<td>45.4102</td>
<td>5.07E-35</td>
</tr>
</tbody>
</table>
TABLE IV

<table>
<thead>
<tr>
<th>Test</th>
<th>Null hypothesis ($H_0 : \mu_1 = \mu_2$)</th>
<th>t-test statistic</th>
<th>P-value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$\mu$MOGA–SVM = $\mu$MOGA</td>
<td>41.0766</td>
<td>2.67E-33</td>
</tr>
<tr>
<td>2</td>
<td>$\mu$MOGA–SVM = $\mu$FCM</td>
<td>92.3495</td>
<td>1.41E-46</td>
</tr>
<tr>
<td>3</td>
<td>$\mu$MOGA–SVM = $\mu$SOM</td>
<td>27.1157</td>
<td>1.07E-26</td>
</tr>
<tr>
<td>4</td>
<td>$\mu$MOGA–SVM = $\mu$Avglink</td>
<td>66.7114</td>
<td>2.72E-41</td>
</tr>
<tr>
<td>5</td>
<td>$\mu$MOGA–SVM = $\mu$SOM</td>
<td>22.6545</td>
<td>5.91E-24</td>
</tr>
</tbody>
</table>

VIII. CONCLUSIONS

This article proposes a novel method for obtaining a final solution from the set of non-dominated solutions produced by an NSGA-II based real-coded multiobjective fuzzy clustering scheme, that optimizes Xie-Beni ($XB$) index and the $J_m$ simultaneously. In this regard, support vector machine based classifiers have been utilized. Results on three real life gene expression data sets have been demonstrated and statistical superiority has been established through statistical significance test. As a scope of further research, performance of other popular classifiers combined with different MOGA technique, such as AMOSA [17] has to be tested. Also detailed study on the effects of the parameters $\alpha$ and $\beta$ has to be made. The authors are currently working in these directions.

REFERENCES


