

Improving Generalization of Radial Basis Function Network with Adaptive Multi-Objective Particle Swarm Optimization

Sultan Noman Qasem

Soft Computing Research Group

Faculty of Computer Science and Information System
Universiti Teknologi Malaysia
Skudai, 81300 Johor, Malaysia
sultannoman@yahoo.com

Siti Mariyam Hj. Shamsuddin

Soft Computing Research Group

k-Economy Research Alliance
Universiti Teknologi Malaysia
Skudai, 81300 Johor, Malaysia
mariyam@utm.my

Abstract— In this paper, an adaptive evolutionary multi-objective selection method of RBF Networks structure is discussed. The candidates of RBF Network structures are encoded into particles in Particle Swarm Optimization (PSO). These particles evolve toward Pareto-optimal front defined by several objective functions with model accuracy and complexity. The problem of unsupervised and supervised learning is discussed with Adaptive Multi-Objective PSO (AMOPSO). This study suggests an approach of RBF Network training through simultaneous optimization of architectures and weights with Adaptive PSO-based multi-objective algorithm. Our goal is to determine whether Adaptive Multi-objective PSO can train RBF Networks, and the performance is validated on accuracy and complexity. The experiments are conducted on two benchmark datasets obtained from the machine learning repository. The results show that our proposed method provides an effective means for training RBF Networks that is competitive with PSO-based multi-objective algorithm.

Keywords— Radial basis function network, Adaptive Multi-objective particle swarm optimization, Multi-Objective particle swarm optimization

I. INTRODUCTION

Radial Basis Function (RBF) Networks form a class of Artificial Neural Networks (ANNs), which has certain advantages over other types of ANNs. It has three layers feed forward fully connected network, which uses RBFs as the only nonlinearity in the hidden layer neurons. The output layer has no nonlinearity and the connections of the output layer are only weighted, the connections from the input to the hidden layer are not weighted [1]. RBF Networks have been widely applied in many science and engineering fields. It is a feedback network of three layers, where each hidden unit implements a radial activation function and each output unit implements weighted sum of hidden units' outputs.

The training of RBF Networks typically consists of combination of unsupervised and supervised learning. Initially, a number of hidden layer nodes (centers) must be positioned in the input space. When each input vector is presented to the network a value is calculated at each center using radial basis function, e.g. Gaussian functions. In the case of Gaussian

function, this value represents measure in the quality of the match between the input vector and location of the center in the input space. Each hidden node therefore, can be considered as a local detector in the input data space. The second phase of the model construction process for RBF Network is the determination of value of the weights on the connections between the hidden layer and the output layer. In training these weights, the output value for each input vector are known, as are the activation values for that input vector at each hidden layer node, so a supervised learning method can be used.

Since the number of input and output layer neurons is determined from the dimension of data, the model designer can only set the number of hidden layer neuron. Hence the model structure determination is to determine the number of hidden layer neuron. Although parameter estimation methods such as the backpropagation method are well known, the method of model structure determination has not been established due to model complexity [2]. If a model is complex, the generalization ability is low because of its high variance error [3]. Conversely, if a model is simple, then it can't represent better correlation between input and output because of the high bias error. The learning method with regularization and the model structure selection method using information criteria have been studied [4] to take a trade-off about the model complexity into account.

In this study, the construction of the Pareto RBF Network set obtained from the perspective of the multi-objective optimization is considered. The set of Pareto optimal RBF Networks based on evaluation of approximation ability and structure complexity using evolutionary computations which show good results on multi-objective optimization is obtained. The optimal RBF Network is constructed from the Pareto set since the obtained Pareto-optimal RBF Networks are diverse on their structure and the diversity. The Pareto optimal RBF Network is applied to the pattern classification problems.

The remainder of this paper is organized as follows: Section II describes an outline of related work of Evolutionary Multi-Objective RBF Networks; Section III describes RBF Network and multi-objective optimization; Section IV introduces the concept of multi-objective optimization and adaptive multi-objective particle swarm optimization; Section V presents the

proposed approach to train RBF Networks involving adaptive PSO-based multi-objective algorithm. The implementation, experiments done and results are discussed in Section VI. The paper ends with the summary, conclusion and future work in Section VII.

II. RELATED WORK OF EVOLUTIONARY MULTI-OBJECTIVE RBF NETWORK

Although there are few studies regarding the implementation of multi-objective RBF Network training, but research on training of RBF Network with evolutionary multi-objective is still new. This section presents some existing work of training RBF Network based on Multi-Objective Evolutionary Algorithms (MOEAs)

In [5], multi-objective (MOBJ) optimization algorithm has been applied to the problem of inductive supervised learning depended on smoothness based apparent complexity measure for RBF Networks. However, the computational complexity of the proposed algorithm is high in comparison with other state-of-art machine learning methods. A multi-objective genetic algorithm based design procedure for the RBF Network has been proposed in [6]. A Hierarchical Rank Density Genetic Algorithm (HRDGA) has been developed to evolve both the neural network's topology and parameters simultaneously.

A method in which RBF Network ensemble has been constructed from Pareto-optimal set obtained by multi-objective evolutionary computation has been proposed in [7]. Pareto-optimal set of RBF Networks has been obtained by multi-objective GA based on three criteria, i.e. model complexity, representation ability, and model smoothness, and RBF Network ensemble has been constructed of this Pareto set. A new evolutionary algorithm, the RBF-Gene algorithm, has been applied to optimize RBF Networks [8]. Unlike other works, their algorithm can evolve both from the structure and the numerical parameters of the network: it is able to evolve the number of neurons and their weights.

Study in [9], presented optimizing RBF Network from training examples as a multi-objective problem and an evolutionary algorithm has been proposed to solve it properly. This algorithm incorporates mutation operators to guide the search to good solutions. A method of obtaining Pareto optimal RBF Network set based on multi-objective evolutionary algorithms has been proposed in [10]. RBF Networks are widely used as model structure for nonlinear systems. The determination of its structure that is the number of basic functions, and the tradeoff between model complexity and accuracy exists.

On the other hand [11], proposed a multi-objective genetic algorithm to the identification of RBF Network couple models of humidity and temperature in a greenhouse. Two combinations of performance and complexity criteria were used to steer the selection of model structures, resulting in distinct sets of solutions. Study in [12], presented PSO as a single objective optimization algorithm to improve performance and classification of RBF Network.

Unlike previous studies mentioned earlier, this study shares the problem of unsupervised learning and supervised learning of RBF Network with adaptive multi-objective PSO to evolve

toward Pareto-optimal front defined by several objective functions with model accuracy and complexity.

III. RBF NETWORK AND MULTI-OBJECTIVE OPTIMIZATION

A. RBF Network

An ANN using radial basis function (RBF) as activation function instead of sigmoid functions is RBF network. The RBF is a function which outputs the maximum value at its center point and decreases its output value as the input leaves from its center. Typically the Gaussian function is used for the activation function.

RBF Network is used in function approximation, time series prediction, and control, and the network architecture is constructed with three layers: input layer, hidden layer, and output layer as shown in Fig.1. The input layer is made up of source nodes that connect the network to its environment. The second layer, the only hidden layer of the network, applies a non-linear transformation from the input space to a hidden space. The nodes in the hidden layer are associated with centers that determine the behavior structure of network. The response from the hidden unit is activated through RBF using Gaussian function or other functions. The output layer neuron calculates the linear sum of values of the hidden neuron and outputs it. In this study the Gaussian function is used as basis functions. Let

$\Phi_j(x)$ be the j -th basis function, $\Phi_j(x)$ is represented as follows :

$$\Phi_j(x) = \exp\left(-\frac{(x - c_j)^2}{2\sigma_j^2}\right) \quad (1)$$

Here, $x = (x_1, x_2, \dots, x_d)^T$ is the input vector, $c_j = (c_{1j}, c_{2j}, \dots, c_{dj})^T$ and σ_j^2 are the j -th center vector and the width parameter, respectively. The output of RBF network y which is the linear sum of basis function, is follows:

$$y = \sum_{j=1}^m w_j \Phi_j(x) \quad (2)$$

Here, y is the output of the RBF Network, m is the number of the hidden layer neuron and w_j is the weight from j th neuron to the output layer.

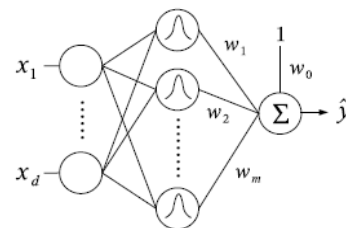


Figure 1. Structure of RBF Network

To construct RBF network, at first the number of the hidden layer neuron m must be set, and then the centers c_j , the widths σ_j and the weights w_j must be estimated. In the typical learning of the RBF network, the network structure will be determined based on prior knowledge or the experiences of experts, and

then the parameters are estimated by methods such as the clustering and the least square method. On the other hand, there are approaches in which the network structure and the parameters are estimated by the evolutionary computation [13][14]. Such a RBF network is called an evolutionary RBF network.

B. Multi-objective Optimization of RBF Network

Generally a mathematical model has dilemma about model complexity[2]. If a model is complex, then it can fit to data well because it has low bias error, but the variance error become high, so the model generalization ability become worse. Moreover, a complex model is not desirable since it is not easy to treat. Conversely, if a model is simple then variance error is small. Such a model can prevent overlearning and the treatment of it will be easy. But it can't represent the relation for input and output enough, since its inflexibility cause the high bias error. The learning method with regularization and the model structure selection method using information criteria have been studied [4] to take a trade-off about the model complexity into account. The trade off about model complexity is expressed by adding a penalty term about model complexity to the error term and using it as evaluation criterion of model construction or model selection. However, there is an approach to solve the above problem by considering the model construction as the multi-objective optimization problem about the model representation ability and the model complexity [15][16][17]. The evolutionary RBF network needs to evolve its structure with consideration of model complexity. By utilizing characteristics of evolutionary algorithms that they are methods of multi-point search, the method obtaining trade-off models has been studied [16][18].

IV. MULTI-OBJECTIVE OPTIMIZATION AND PARTICLE SWARM OPTIMIZATION

A. Multi-objective Optimization Problem

Many real-world problems involve simultaneous optimization of several objective functions. Generally, these functions are non-commensurable and often conflicting objectives. Multi-objective optimization with such conflicting objective functions gives rise to a set of optimal solutions, instead of one optimal solution. The reason for the optimality of many solutions is that no one can be considered to be better than any other with respect to all objective functions. These optimal solutions are known as Pareto-optimal solutions.

A general multi-objective optimization problem consists of a number of objectives to be optimized simultaneously and is associated with a number of equality and inequality constraints. It can be formulated as follows:

$$\begin{aligned} & \text{Minimize/Maximize } f_i(x) \quad i=1, \dots, M \\ & \text{Subject to: } \begin{cases} g_j(x) = 0 & j = 1, \dots, N \\ h_k(x) \leq 0 & k = 1, \dots, K \end{cases} \end{aligned} \quad (3)$$

Where f_i is the i th objective function, x is a decision vector that represents a solution, and M is the number of objectives.

For a multi-objective optimization problem, any two solutions x_1 and x_2 can have one of two possibilities- one dominates the other or none dominates the other. In a minimization problem, without loss of generality, a solution x_1 dominates x_2 iff the following two conditions are satisfied:

$$\begin{aligned} & \forall i \in \{1, 2, \dots, M\} : f_i(x_1) \leq f_i(x_2), \\ & \exists j \in \{1, 2, \dots, M\} : f_j(x_1) < f_j(x_2) \end{aligned} \quad (4)$$

The solutions which are not dominated by any other solutions are called the Pareto-optimal solution or non-dominated solution. Generally many Pareto-optimal solutions exist. The set of Pareto-optimal solutions is called Pareto optimal front. A non-dominated set is required to be near to the true Pareto front and to be distributed uniformly.

B. Adaptive Multi-objective Particle Swarm Optimization Algorithm(AMPSO)

AMOPSO [19] is made adaptive in nature by allowing its vital parameters (viz., inertia weight and acceleration coefficients) to change with iterations. This adaptiveness helps the algorithm to explore the search space more efficiently. A new diversity parameter has been used to ensure sufficient diversity amongst the solutions of the non dominated fronts, while retaining at the same time the convergence to the Pareto-optimal front. The basic structure of AMOPSO is given in algorithm AMOPSO.

The AMOPSO extends the algorithm of the single-objective PSO to handle multi-objective optimization problems. It incorporates the mechanism of crowding distance computation into the algorithm of PSO specifically on global best selection and in the deletion method of an external archive of non-dominated solutions. The crowding distance mechanism together with a mutation operator maintains the diversity of non-dominated solutions in the external archive. AMOPSO also has a constraint handling mechanism for solving constrained optimization problems.

Algorithm TV-MOPSO: $O_f = AMOPSO(N_s, N_a, C, d)$

/ N_s : size of the swarm, N_a : size of the archive, C: maximum number of iterations, d: the dimensions of the search space, of: the final output */*

1) $t = 0$, randomly initialize S_0 ,

/ S_t : swarm at iteration t */*

1. initialize $X_{i,j}$

/ $X_{i,j}$: the jth coordinate of the ith particle */*

2. initialize $V_{i,j}$

/ $V_{i,j}$: the velocity of ith particle in jth dimension */*

3. $Pb_{i,j} = X_{i,j}$

/ $Pb_{i,j}$: the jth coordinate of the personal best of the ith particle */*

4. $A_0 = \text{non_dominated}(S_0)$, $I_0 = |A_0|$ */* returns the non dominated solutions from the swarm */*

/ A_t : archive at iteration t */*

- 2) for $t = 1$ to $t = C$,
 5. for $i = 1$ to $i = N_s$ /* update the swarm S_t */
 - . /* updating the velocity of each particle */
 - $G_b = \text{get_gbest}()$ /* returns the global best */
 - $P_b_i = \text{get_pbest}()$ /* returns the personal best */
 - $\text{adjust_parameters}(w_t, c_{1t}, c_{2t})$
 - /* adjusts the parameters in each iteration. w_t : the inertia coefficient, c_{1t} : the local acceleration coefficient and c_{2t} : the global acceleration coefficient */
 - $V_{ij} = w_t V_{ij} + c_{1t} r_1 (P_{b_{ij}} - X_{ij}) + c_{2t} r_2 (G_{b_{ij}} - X_{ij})$
 - /* updating coordinates */
 - $X_{ij} = X_{ij} + V_{ij}$
 6. /* updating the archive */
 - . $A_t = \text{non_dominated}(S_t \cup A_t)$
 - .if ($I_t > N_a$) $\text{truncate_archive}()$
 - /* I_t : size of the archive */
 7. $\text{mutate}(S_t)$ /* mutating the swarm */
- 3) $O_f = A_t$ and stop. /* returns the Pareto optimal front */

This algorithm terminates after executing a specified number of iterations. After termination, the archive contains the final non-dominated front. This is mentioned in Step 3 of AMOPSO

V. HYBRID LEARNING OF RBF NETWORK USING AMOPSO

The proposed algorithm called RBFN-AMOPSO is a multi-objective optimization approach to RBF Network training with AMOPSO as the multi-objective optimizer. The algorithm will simultaneously determine the set of connection weights and its corresponding architecture by treating this problem as a multi-objective minimization problem. In this study, a particle represents a one-hidden layer RBF Network and the swarm consists of a population of one-hidden layer networks. We set the number of hidden neurons depended on the problem to be solved.

A. Parameters and Structure Representation

The RBF Network is represented as a vector with dimension D contains the connections. The dimension of a particle is:

$$D = (I \times H) + (H \times O) + H + O \quad (5)$$

Where I , H and O are refer to the number of input, hidden and output neurons respectively. The centers of RBF are initialized from k-means clustering algorithm and the connection weights of RBF Network are initialized with random values. The number of input and output neurons is problem-specific and there is no exact way of knowing the best number of hidden neurons. We set the number of hidden neurons (RBFs) depended on the number of clusters (classes) of the problem to be solved.

B. RBFN-AMOPSO

RBFN-AMOPSO starts by reading the dataset. This is followed by setting the desired number of hidden neurons and the maximum number of generation for AMOPSO. The next step is determining the dimension of the particles and initializing the population with fully-connected feed-forward RBF Network particles. In each generation, every particle is evaluated based on the two objective functions and after the maximum generation is reached the algorithm outputs a set of non-dominated Pareto RBF Networks.

C. Objective Functions

Two objective functions are used to evaluate the RBF Network particle's performance. The two objective functions are:

1. Mean-squared error (MSE) on the training set.

$$f_1 = \frac{1}{N} \sum_{j=1}^N (t_{kj} - o_{kj})^2 \quad (6)$$

2. Complexity based on the sum of the squared weights which is based on the concept of regularization and represents the smoothness of the model.

$$f_2 = \frac{1}{2} \sum_{j=1}^M w_j^2 \quad (7)$$

VI. EXPERIMENTS STUDIES

To evaluate the performance of RBFN-AMOPSO, several experiments were conducted on two data sets listed in Table I. All data sets come from the machine learning benchmark repository [20]. These problems have been the subjects of many studies in ANNs and machine learning. Experimental details, results and comparisons with other work are presented in the following sub-sections.

A. Experimental Setup

In this study, all data sets are partitioned into three: a training set, a validation set and a testing set. The validation set is used to select the best one from the pareto optimal solutions, while the testing set is used to test the generalization performance of pareto RBF Network. It is known that the experimental results may vary significantly for different partitions of the same data set.

TABLE I. DESCRIPTION OF DATA SETS

Data set	Attributes	Classes	Instances
Breast Cancer	9	2	699
Diabetes	8	2	768

In the experiment, initially, we analyze the evolutionary process of RBFN-AMOPSO and evaluate the performance of breast cancer and diabetes data sets. To do so, we partitioned the two data sets randomly as follows:

1. For the breast cancer data set, 50% of data (i.e., 349 examples) were used for the training set, 25% of data

(i.e., 175 examples) for the validation set, and the rest (i.e., 175 examples) for the testing set.

- For the diabetes data set, 50% of data (i.e., 384 examples) were used for the training set, 25% of data (i.e., 192 examples) for the validation set, and the rest (i.e., 192 examples) for the testing set.

For each data set, the experiments were implemented to minimize the influence of random effects. Each experiment uses a different randomly generated initial population. In addition, the number of input and output nodes is problem-dependent but the number of hidden nodes is the number of classes (clusters) of data. The number of iterations are the same for all gorithms.

There are some parameters in AMOPSO-RBFN which need to be specified by the user. Therefore, these parameters were set to the same for all these problems: the population size 100, the number of generations (1,000), the initial position and velocity range of particles $[-0.5, +0.5]$, the inertia weights w_{max} and w_{min} (0.7 and 0.4), the initial acceleration coefficients c_1 and c_2 (2.5 and 0.5). c_1 has been allowed to decrease from its initial value of 2.5 to 0.5 while c_2 has been increased from 0.5 to 2.5.

B. Experimental Results

This section presents results of study on AMOPSO-based RBF Network. The experiments are conducted by using two datasets. The results for each dataset are compared to MOPSO Algorithm and analysed based on the convergence to Pareto optimal set, error and classification performance. One advantage of evolutionary multi-objective optimization approach to RBF Network generation is that a number of RBF Networks with a spectrum of complexity can be obtained in one single run. With these RBF Networks that trade off between training error and model complexity, we hope that we can identify those networks that are most likely to have good generalization capability, i.e., perform well both on training data and unseen test data.

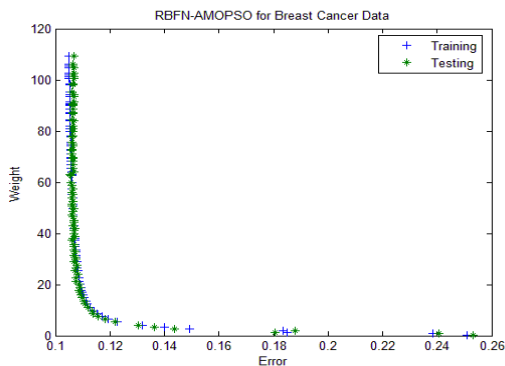


Figure 2. Pareto front obtained using RBFN-AMOPSO for Breast Cancer

Table II presents the results of RBFN-AMOPSO on the breast cancer and diabetes data sets (the training error, validation error and testing error respectively). The result of this algorithm is pareto optimal solutions to improve the generalization on unseen data . We report the results in terms of error and correct classification on the two data sets. Fig. 2 and Fig. 3 demonstrate that AMOPSO has the capability to evolve compact RBF Networks which generalize well on unseen data.

TABLE II. RESULTS OF RBFN-AMOPSO

Data set	Training Error	Validation Error	Testing Error
Breast Cancer	0.105957	0.090161	0.105389
Diabetes	0.177349	0.167655	0.153665

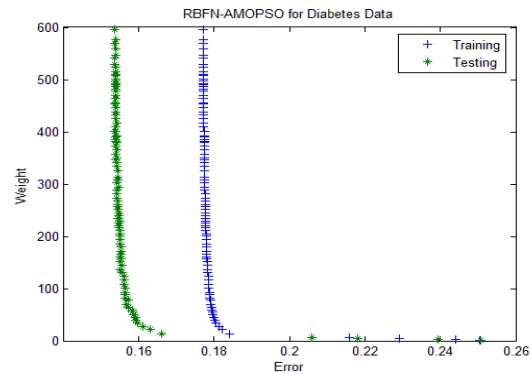


Figure 3. Pareto front obtained using RBFN-AMOPSO for Diabetes

C. Comparison with Other Work

In order to evaluate the performance of RBFN-AMOPSO under generalization, the comparison was carried out by using MOPSO [21] based RBF Network on all data sets. Table III shows the results of RBFN-MOPSO on training, validation and testing of breast cancer and diabetes data sets. Fig. 4 and Fig. 5 illustrate that pareto front obtained for breastcancer and diabetes data sets. From comparison, Fig.2-5 show that RBFN-MOPSO is better than or similar to RBFN-AMOPSO in convergence to pareto front but RBFN-AMOPSO maintains diversity of the pareto front solutions for breast cancer and diabetes dat sets.

TABLE III. RESULTS OF RBFN-MOPSO

Data set	Training Error	Validation Error	Testing Error
Breast Cancer	0.107342	0.088288	0.105992
Diabetes	0.169091	0.166373	0.146295

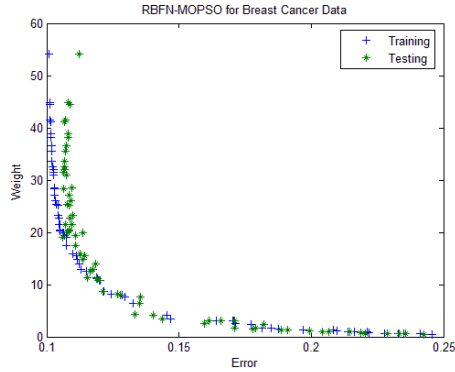


Figure 4. Pareto front obtained using RBFN-MOPSO for Breast Cancer

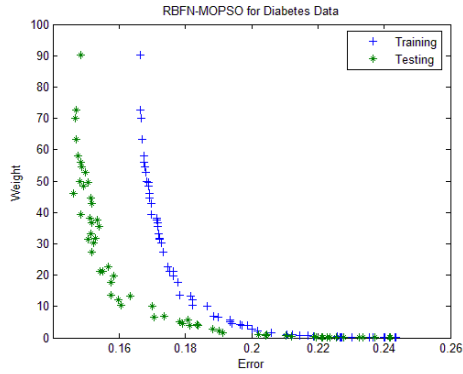


Figure 5. Pareto front obtained using RBFN-MOPSO for Diabetes

D. Classification Results

Table IV presents the accuracy results of RBF Network based on AMOPSO and MOPSO for breast cancer and diabetes dat sets. The accuracy in the table refers to the percentage of correct classification on training and testing data sts.

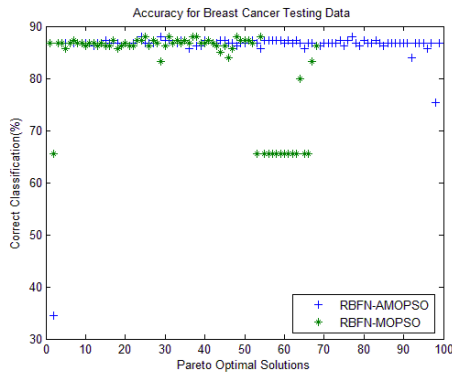


Figure 6. Correct Classification of parto front for Breast Cancer

RESULTS OF RBFN-AMOPSO AND RBFN-MOPSO

Data set	RBFN-AMOPSO		RBFN-MOPSO	
	Training Accuracy	Testing Accuracy	Training Accuracy	Testing Accuracy
Breast Cancer	87.43	86.78	86.86	87.36
Diabetes	73.7	79.69	72.14	76.56

Fig. 6 and 7 show that correct classification of pareto fronts which are produced with RBFN-AMOPSO and RBFN-MOPSO on breast cancer and diabetes testing data sets. RBF Network based on AMOPSO has high percentages of accuracy rather than RBFN-MOPSO because AMOPSO maintains diversity of pareto fronts solutions in RBF Network .

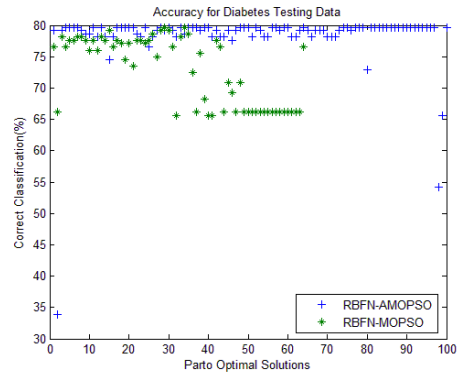


Figure 7. Correct Classification of parto front for Diabetes

VII. CONCLUSIONS AND FUTURE WORK

This paper introduces adaptive multi-objective PSO approach to RBF Network design called Adaptive Multi-objective PSO-RBF Network Optimizer to concurrently optimize the architectures and connections of network. The optimal RBF Network is constructed from Pareto optimal front set obtained by AMOPSO. RBF Network structure and its parameters are encoded to the particles, and Pareto-optimal set of RBF Networks is obtained by AMOPSO based on two criteria, i.e. model complexity, representation ability. The benchmark of pattern classification indicates that our proposed method can get better or comparable results similar to multi-objective PSO.

Further improvement of the proposed algorithm will be the automatic parameters tuning and structure tuning of RBF Network. The proposed method will optimize the network performance and its structure at the same time, in terms of hidden nodes (RBF) and active connections and we will do the cross validation and ROC analysis for data sets in future work.

ACKNOWLEDGMENT

This work is supported by Ministry of Higher Education (MOHE) under Fundamental Research Grant Scheme (Vote No.78183). Authors would like to thank Research Management Centre (RMC) Universiti Teknologi Malaysia, for the research

activities and *Soft Computing Research Group (SCRG)* for the support and motivation in making this study a success.

REFERENCES

- [1] J. A. Leonard and M. A. Kramer, "Radial basis function networks for classifying process faults", *Control Systems Magazine*, 1991, Vol. 11 Iss.3, p. 31-38
- [2] O. Nelles, "Nonlinear System Identification", Springer, 2001.
- [3] L. Yu, K. K. Lai, S. Wang and W. Huang, "A Bias Variance-Complexity Trade-Off Framework for Complex System Modeling", Springer-Verlag, 2006, pp. 518 – 527
- [4] T. Hatanaka, K. Uosaki and T. Kuroda, "Structure Selection of RBF Neural Network Using Information Criteria", *Proceedings of Fifth International Conference on Knowledge-Based Intelligent Information Engineering Systems and Allied Technologies*, 2001.
- [5] I. Kokshenev and A. P. Braga, "A multi-objective approach to RBF network learning", *Neurocomputing*, Vol.71,p. 1203–1209, 2008.
- [6] G. G. Yen, "Multi-Objective Evolutionary Algorithm for Radial Basis Function Neural Network Design", Springer-Verlag, Vol. 16, pp. 221–239, 2006.
- [7] N. Kondo, T. Hatanaka and K. Uosaki, "Pattern Classification by Evolutionary RBF Networks Ensemble Based on Multi-objective Optimization", *International Joint Conference on Neural Networks*, pp. 2919-2925, 2006.
- [8] V. Lefort, C. Knibbe, G. Beslon and J. Favrel, "Simultaneous Optimization of Weights and Structure of an RBF Neural Network", Springer-Verlag, LNCS 3871, pp. 49–60, 2006.
- [9] J. Gonzalez, I. P. H. Rojas and j. Ortega, "RBF Neural Networks, Multiobjective Optimization and Time Series Forecasting", Springer-Verlag, LNCS 2084, p.498-505, 2001.
- [10] N. Kondo, T. Hatanaka and K. Uosaki, "Nonlinear Dynamic System Identification Based on Multiobjectively Selected RBF Networks", *Proceedings of the 2007 IEEE Symposium on Computational Intelligence in Multicriteria Decision Making*, pp. 112-127.
- [11] P. M. Ruano Ferreira, A.E. Fonseca and C.M., "Evolutionary Multi-Objective Design of Radial Basis Function Networks for Greenhouse Environmental Control", *Proc. of the 16 the IFAC World Congress*, 2005.
- [12] S. Noman, S. M. Shamsuddin and A. E. Hassanien, "Hybrid Learning Enhancement of RBF Network with Particle Swarm Optimization", Springer-Verlag, Vol. 201, pp. 381-397, 2009.
- [13] Y. Bai and L. Zhang, "Genetic Algorithm Based Self-Growing Training for RBF Neural Networks", *Proceedings of International Joint Conference on Neural Networks 2002*, pp. 840-845, 2002.
- [14] Z. Wenbo, H. De-Shuang and Y. Ge, "The Structure Optimization of Radial Basis Probabilistic Neural Networks Based on Genetic Algorithms", *Proceedings of International Joint Conference on Neural Networks 2002*, pp. 1086-1091, 2002.
- [15] H. A. Abbass, "A Memetic Pareto Evolutionary Approach to Artificial Neural Networks", *Australian Joint Conference on Artificial Intelligence*, pp.1-12, 2001.
- [16] G. G. Yen and H. Lu, "Hierarchical Rank Density Genetic Algorithm for Radial-Basis Function Neural Network Design", *International Journal of Computational Intelligence and Applications*, vol.3, No.3, pp.213-232, 2003.
- [17] Y. Jin, T. Okabe and B. Sendhoff, "Neural Network Regularization and Ensembling Using Multi-objective Evolutionary Algorithms", *Proceedings of Congress on Evolutionary Computation*, pp.1-8, 2004.
- [18] N. Kondo, T. Hatanaka and K. Uosaki, "Pareto RBF Networks Based on Multiobjective Evolutionary Computation", *SICE Annual Conference in Sapporo*, pp.2177-2182, 2004.
- [19] P. K. Tripathi, S. Bandyopadhyay and S. K. Pal, "Multi-Objective Particle Swarm Optimization with time variant inertia and acceleration coefficients", *International Journal of Information Sciences*, vol.177, No.3, pp. 5033–5049, 2007.
- [20] A. Asuncion, D. Newman, *Machine learning repository*. URL(http://www.ics.uci.edu/_mlearn/MLRepository.html), 2007.
- [21] S. N. Qasem and S. M. Shamsuddin, "Radial Basis Function Network Based on Multi-Objective Particle Swarm Optimization", *Proceeding of the 6th International Symposium on Mechatronics and its Applications*, 2009, in press.