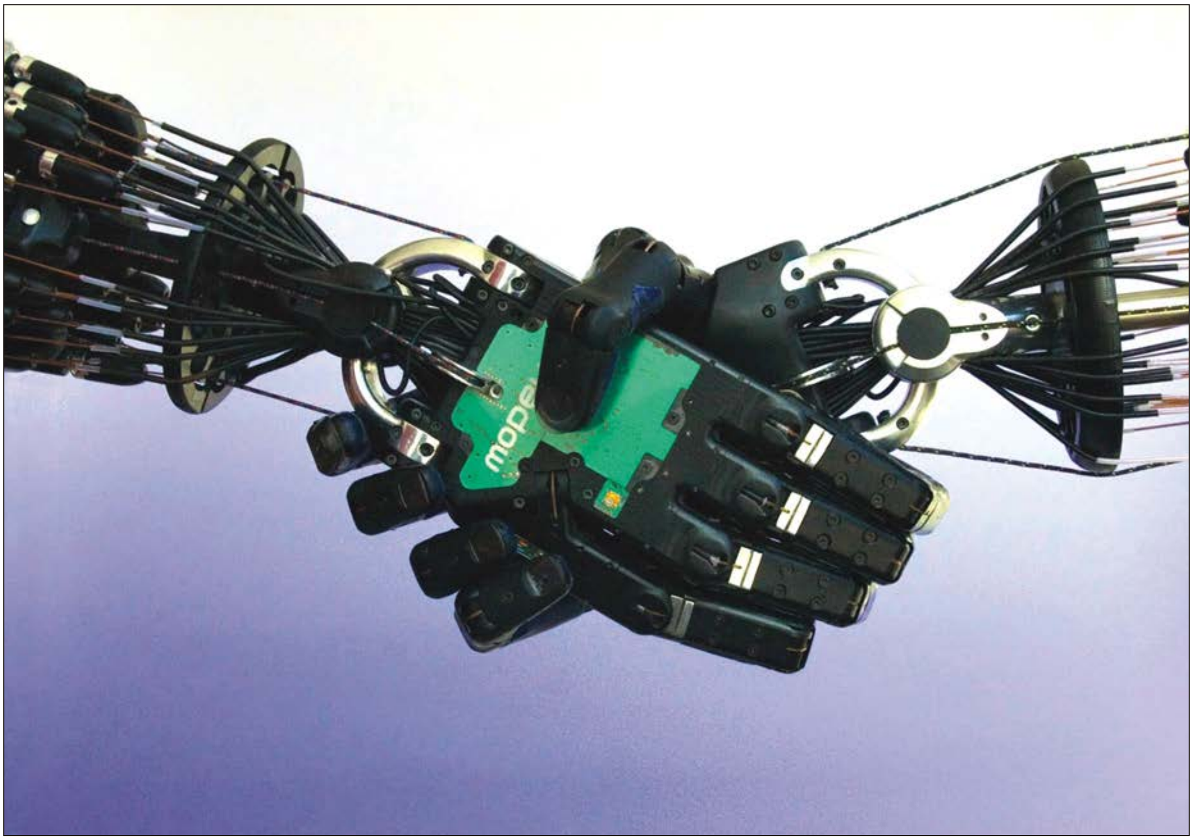

Journal of Artificial Intelligence

ISSN 1994-5450

science alert

ANSInet
an open access publisher
http://ansinet.com

Journal of Artificial Intelligence 3 (1): 1-16, 2010
ISSN 1994-5450

Generalization Improvement of Radial Basis Function Network Based on Multi-Objective Particle Swarm Optimization

S.N. Qasem and S.M. Shamsuddin
Soft Computing Research Group,
Faculty of Computer Science and Information System,
Universiti Teknologi Malaysia, 81310, Johor, Malaysia

Abstract: The problem of unsupervised and supervised learning of RBF networks is discussed with Multi-Objective Particle Swarm Optimization (MOPSO). This study presents an evolutionary multi-objective selection method of RBF networks structure. The candidates of RBF networks structures are encoded into particles in PSO. These particles evolve toward Pareto-optimal front defined by several objective functions with model accuracy and complexity. This study suggests an approach of RBF network training through simultaneous optimization of architectures and connections with PSO-based multi-objective algorithm. Present goal is to determine whether MOPSO 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; the best results are obtained for our proposed method that has obtained 100 and 80.21% classification accuracy from the experiments made on the data taken from breast cancer and diabetes diseases database, respectively. The results also show that our approach provides an effective means to solve multi-objective RBF networks and outperforms multi-objective genetic algorithm.

Key words: Radial basis function network, hybrid learning, multi-objective optimization, multi-objective particle swarm optimization, elitist non-dominated sorting genetic algorithm

INTRODUCTION

Radial Basis Function (RBF) networks form a class of Artificial Neural Networks (ANNs) which has certain advantages over other types of ANNs. Due to their better approximation capabilities, simpler network structures and faster learning algorithms, RBF networks have certain advantages over other types of ANNs and have been widely applied in many science and engineering fields. It has three layers feed forward fully connected network which uses Radial Basis Functions (RBFs) as the only nonlinearity in the hidden layer neurons. The output layer has no nonlinearity, the connections of the output layer are only weighted and the connections from the input to the hidden layer are not weighted (Leonard and Kramer, 1991).

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,

Corresponding Author: Sultan Noman Qasem, Soft Computing Research Group,
Faculty of Computer Science and Information System,
Universiti Teknologi Malaysia, 81310, Johor, Malaysia

parameter estimation methods such as the backpropagation method are well known, the method of model structure determination has not been established. This is caused by the model complexity (Nelles, 2001). If a model is complex, the generalization ability is low because of its high variance error (Yu *et al*., 2006). Conversely, if a model is simple, then it can't represent better correlation between input and output because of high bias error. The learning method with regularization and the model structure selection method using information criteria have been studied by Hatanaka *et al.* (2001) 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 had show good results on multi-objective optimization. The optimal RBF network is constructed from the Pareto set since the obtained Pareto-optimal RBF networks are diverse on its structure and the diversity. The Pareto optimal RBF network is applied to the pattern classification problems.

The objective of this study to design a novel framework based on multi-objective PSO and RBF network for improving generalization and classification accuracy while avoiding over-fitting in data and it compared to multi-objective GA.

RBF NETWORK

Artificial Neural Network (ANN) using Radial Basis Function (RBF) as activation function instead of sigmoid functions is RBF network. The RBF network is constructed of three layers which are the input layer, the hidden layer and the output layer, as shown in Fig. 1. The input layer neuron has a role to transmit data to the hidden layer. The hidden layer neuron calculates value of the basis function by the received data from the input layer and then transmits the value to the output layer. The output layer neuron calculates the linear sum of values of the hidden neuron. In this study, the Gaussian function is used as radial basis functions. Let Φ_j (x) be the j-th basis function, Φ_j (x) is represented as follows:

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

where, $x = (x_1, x_2 \ldots x_d)^T$ is the input vector, $c_j = (c_{1j}, c_{2j}, \ldots, c_{dj})^T$ and σ^2_j 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:

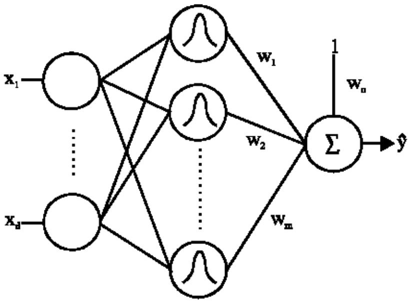

Fig. 1: Structure of RBF Network

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

where, y is the output of the RBF Network, m is the number of the hidden layer neuron and w_j is the weight from jth neuron to the output layer.

In the typical learning of 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 its parameters are estimated by the evolutionary computation (Bai and Zhang, 2002; Wenbo *et al*., 2002). Such a RBF network is called an evolutionary RBF network.

MULTI-OBJECTIVE OPTIMIZATION OF RBF NETWORK

Generally, a mathematical model has dilemma about model complexity (Nelles, 2001). 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 becomes 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 over-learning and the treatment of it will be easy. The learning method with regularization and the model structure selection method using information criteria have been studied by Hatanaka *et al.* (2001). 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 the model construction and selection. However, there is an alternative approach that solve the above problem by considering the model construction as the multi-objective optimization problem about the model representation ability and the model complexity (Abbass, 2001; Yen and Lu, 2003; Jin *et al*., 2004). 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 by Yen and Lu (2003) and Kondo *et al.* (2004).

Neural Networks (NNs) are bio-inspired tools that have been widely used in tasks such as control processes, optimization, pattern recognition, classification, etc. The use of a neural network to solve problem involves selecting the appropriate network model and network structure and then training. These stages often require high computational time and human interaction until the required network is obtained. In case of Radial Basis Function (RBF) networks, the classic training algorithms are limited and so, these tasks may become even more difficult.

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. Here, some existing work of training RBF network based on multi-objective evolutionary algorithms (MOEAs) is presented.

Kokshenev and Braga (2008) applied multi-objective (MOBJ) optimization algorithm 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-the-art machine learning methods. A multi-objective genetic algorithm based design procedure for the RBF network has been proposed by Yen (2006). A hierarchical rank density genetic algorithm (HRDGA) has been developed to evolve both the neural network's topology and its parameters simultaneously.

Kondo *et al*. (2006) proposed method in which RBF network ensemble has been constructed from Pareto-optimal set obtained by multi-objective evolutionary computation. 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. Lefort *et al.* (2006) applied new evolutionary algorithm, the RBF-Gene algorithm to optimize RBF networks. 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.

González *et al*. (2001) presented optimizing RBF network from training examples as a multi-objective problem and an evolutionary algorithm has been proposed to solve it. Their 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 by Kondo *et al*. (2007) to solve nonlinear dynamic system identification problem. RBF networks are widely used as model structure for nonlinear systems. The determination of its structure that is the number of radial basic functions and the tradeoff between model complexity and accuracy exists. On the other hand, Ferreira *et al*. (2005) 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 guide the selection of model structures, resulting in distinct sets of solutions.

Unlike previous studies mentioned earlier, this study shares the problem of unsupervised learning and supervised learning of RBF network with multi-objective PSO which evolve toward Pareto-optimal front defined by several objective functions with model accuracy and complexity to improve generalization while avoiding over-fitting on data.

MULTI-OBJECTIVE OPTIMIZATION PROBLEM

Many real-world problems involve simultaneous optimization of several objective functions. Generally, these functions are 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:

$$f_i(x) \qquad i = 1,\ldots, M$$

$$\text{Subject to}: \begin{cases} g_j(x) = 0 & j = 1,\ldots, N \\ h_k(x) \le 0 & k = 1,\ldots,K \end{cases} \tag{3}$$

where, f_i is the ith 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:

$$\forall i \in \{1,2,\ldots,M\} : f_i(x_1) \le f_i(x_2), \quad \exists j \in \{1,2,\ldots,M\} : f_j(x_1) \prec f_j(x_2) \tag{4}$$

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 distributed uniformly.

PARTICLE SWARM OPTIMIZATION

Particle Swarm Optimization (PSO) is a population based stochastic optimization technique developed by Kennedy and Eberhart (1995), inspired by social behavior of bird flocking or fish schooling, in which each individual is treated as an infinitesimal particle in the n-dimensional space, with the position vector and velocity vector of particle i being represented as $X_i(t) = (X_{i1}(t), X_{i2}(t),\ldots, X_{in}(t))$ and $V_i(t) = (V_{i1}(t), V_{i2}(t),\ldots, V_{in}(t))$. The particles move according to the following equations:

$$V_{id}(t+1) = W \times V_{id}(t) + c_1 r_1 \left(P_{id}(t) - X_{id}(t)\right) + c_2 r_2 \left(P_{gd}(t) - X_{id}(t)\right) \tag{5}$$

$$x_{id}(t+1) = x_{id}(t) + V_{id}(t+1) \tag{6}$$

$$I = 1,2,\ldots, M \; ; \; d = 1,2,\ldots,n$$

where, c_1 and c_2 are the acceleration coefficients, Vector $P_i = (P_{i1}, P_{i2},\ldots, P_{in})$ is the best previous position (the position giving the best fitness value) of particle i known as the personal best position (pbest); Vector $P_g = (P_{g1}, P_{g2},\ldots, P_{gn})$ is the best position among the personal best positions of the particles in the population and is known as the global best position (gbest). The parameters r_1 and r_2 are two random numbers distributed uniformly in (0, 1). Generally, the value of V_{id} is restricted in the interval $[-V_{max}, V_{max}]$. Inertia weight w was first introduced by Shi and Eberhart in order to accelerate the convergence speed of the algorithm (Shi and Eberhart, 1998).

MULTI-OBJECTIVE PARTICLE SWARM OPTIMIZATION ALGORITHM

A particle swarm algorithm for the solution of multi-objective problems was presented by (Coello and Lechuga, 2002). In MOPSO, in contrast PSO, there are many fitness functions. Differently from PSO, in MOPSO there is no global best, but a repository with the non-dominated solutions found.

The MOPSO (Raquel and Naval, 2005) 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 (Fig. 2). The crowding distance mechanism together with a mutation operator maintains the diversity of non-dominated solutions in the external archive. This algorithm also has a constraint handling mechanism for solving constrained optimization problems. The MOPSO algorithm is presented below:

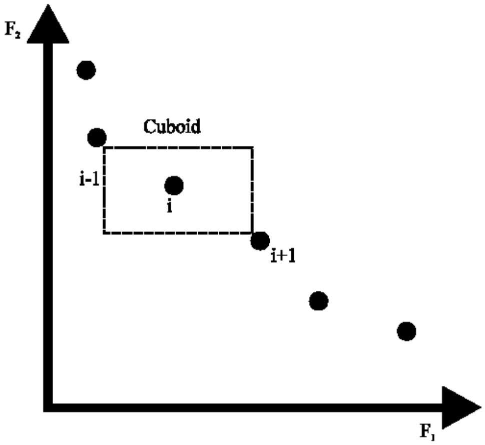

Fig. 2: Crowding distance computation

MOPSO Algorithm

BEGIN

Input: Optimization problem
Output: Non-dominated solutions in archive (A)

```
1.  For i = 1 to M (M is the population size)
    a.  Initialize $P_i$ randomly (P is the population of particles)
    b.  Initialize $V_i = 0$ (V is the speed of each particle)
    c.  Evaluate $P_i$
    d.  Initialize the personal best of each particle
        $PBESTS_i = P_i$
    e.  GBEST = Best particle found in $P_i$
2.  End For
3.  Initialize the iteration counter t = 0
4.  Store the non-dominated vectors found in P into A
    (A is the external archive that stores non-dominated solutions found in P)
5.  Repeat
    a.  Compute the crowding distance values of each non-dominated solution in the
        archive A
    b.  Sort the non-dominated solutions in A in descending crowding distance values
    c.  For i = 1 to M
        i.    Randomly select the global best guide for $P_i$ from a specified top portion
              (e.g., top 10%) of the sorted archive A and store its position to GBEST.
        ii.   Compute the new velocity:

              $V_{i+1} = W \times V_i + R_1C_1(PBEST_i - P_i) + R_2C_2\left(A\left(GBEST\right) - P_i\right)$

              (W is the inertia weight equal, $R_1$ and $R_2$ are random numbers in the range
              [0..1]), $C_1$ and $C_2$ are acceleration coefficients, $PBESTS_i$ is the best position
              that the particle i have reached and A(GBEST) is the global best guide for
              each non-dominated solution)
        iii.  Calculate the new position of $P_i$:
                $P_{i+1} = P_i + V_{i+1}$
        iv.  If $P_i$ goes beyond the boundaries, then it is reintegrated by having the
             decision variable take the value of its corresponding lower or upper boundary
             and its velocity is multiplied by -1 so that it searches in the opposite direction
        v.   If (t < (MAXT * PMUT),
             Then perform mutation on $P_i$
             (MAXT is the maximum number of iterations and PMUT is the probability of
             mutation)
        vi   Evaluate $P_i$
    d.  End For
    e.  Insert all new non-dominated solution in P into A if they are not dominated by any
        of the stored solutions. All dominated solutions in the archive by the new solution
        are removed from the archive. If the archive is full, the solution to be replaced is
        determined by the following steps:
        i.   Compute the crowding distance values of each non-dominated solution in the
             archive A
        ii.  Sort the non-dominated solutions in A in descending crowding distance
             values
        iii. Randomly select a particle from a specified bottom portion (e.g., lower 10%)
             which comprise the most crowded particles in the archive then replace it with
             the new solution
    f.  Update the personal best solution of each particle in P. If the current PBESTS
        dominates the position in memory, the particles position is updated using
        $PBESTS_i = P_i$
    g.  Increment iteration counter t
6.  Until maximum number of iterations is reached

END
```

ELITIST NON-DOMINATED SORTING GENETIC ALGORITHM (NSGA-II)

The capabilities of multi-objective genetic algorithms (MOGAs) to explore and discover Pareto optimal fronts on multi-objective optimization problems have been well recognized. It has been shown that MOGAs outperform traditional deterministic methods due to their capacity to explore and combine various solutions to find the Pareto front in a single run. We will implement a multi-objective optimization technique called the non-dominated sorting genetic algorithm II (NSGA-II) (Deb *et al.*, 2002) to RBF network training. The NSGA-II algorithm may be stated as follows:

1. Create a random parent population of size N
2. Sort the population based on the non-domination
3. Assign each solution a fitness (or rank) equal to its non-domination level (minimization of fitness is assumed)
4. Use the usual binary tournament selection, recombination and mutation operators to create a new offspring population of size N
5. Combine the offspring and parent population to form extended population of size 2N
6. Sort the extended population based on non-domination

7. Fill new population of size N with the individuals from the sorting fronts starting from the best
8. Invoke the crowding comparison operator to ensure diversity if a front can only partially fill the next generation
9. Repeat the steps (2)-(8) until the stopping criterion is met. The stopping criterion may be a specified number of generations

It is clear from the above description that NSGA-II uses (i) a fast non-dominated sorting approach, (ii) an elitist strategy and (iii) no niching parameter (Deb *et al.*, 2002).

HYBRID LEARNING OF RBF NETWORK BASED ON MOPSO

The proposed algorithm called RBFN-MOPSO is a multi-objective optimization approach to RBF network training with MOPSO as the multi-objective optimizer. The algorithm will simultaneously determine the set of connections 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.

Parameters and Structure Representation

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

$$D = (I \times H) + (H \times O) + H + O \tag{7}$$

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.

RBFN-MOPSO

RBFN-MOPSO starts by collecting, normalize and reading the dataset. This is followed by setting the desired number of hidden neurons and the maximum number of generation for MOPSO. The next step is determining the dimension of the particles and initializing the population with fully-connected feed-forward RBF network. 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. Figure 3 shows the description of proposed study.

Objective Functions

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

$$f_1 = \frac{1}{N}\sum_{j=1}^{N}(t_{kj} - o_{kj})^2 \tag{8}$$

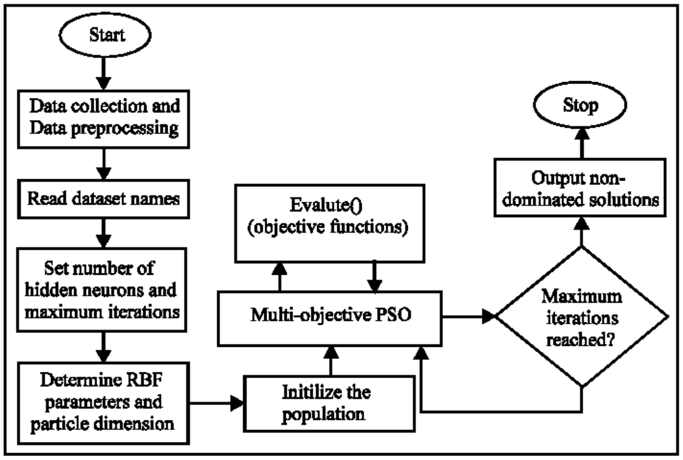
Start
Data collection and Data preprocessing
Read dataset names
Set number of hidden neurons and maximum iterations
Determine RBF parameters and particle dimension
Initilize the population
Multi-objective PSO
Evalute() (objective functions)
Maximum iterations reached?
Output non-dominated solutions
Stop

Fig. 3: The description of the proposed study RBFN-MOPSO

- Accuracy based on Mean-Squared Error (MSE) on the training set where o_{kj} and t_{kj} are the model output and the desired output, respectively and N is the number of data pairs in the training data
- Complexity based on the sum of the squared weights which is based on the concept of regularization and represents the smoothness of the model

$$f_2 = \frac{1}{2}\sum_{j=1}^{M} w_j^2 \tag{9}$$

where w_j, j=1... M is a weight in the network mode and M is the number of weights in total.

EXPERIMENTAL STUDIES

To evaluate the performance of RBFN-MOPSO, several experiments were conducted on two data sets are listed in Table 1. All data sets have been loaded from the machine learning benchmark repository (Asuncion and Newman, 2007). These problems have been the subjects of many studies in ANNs and machine learning.

Table 1: Description of data sets

Data Set	Attributes	Classes	Samples	Remarks
Breast cancer	9	2	699	Determine the patients for whom the tumor is benign or malignant. This data set has 16 missing values
Diabetes	8	2	768	Determine whether a patient shows sign of diabetes according to World Health Organization Criteria

Experimental Setup

In this study, all data sets are partitioned into three sets: 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 2: Parameters settings for RBFN-MOPSO

Parameters	RBFN-MOPSO
Optimization type	Minimization
Population size	100
Archive size	100
Objective functions	2
Constraints	0
Lower limit of variable	-0.5
Upper limit of variable	0.5
Probability of mutation (pM)	0.5

In the experiment, we analyze the evolutionary process of RBFN-MOPSO and evaluate the performance of it on breast cancer and diabetes data sets and compare with RBFN-NSGA-II. To do so, we partitioned the data sets sequentially as follows:

- 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 is the same for all algorithms. There are some parameters in MOPSO which need to be specified by the user. Therefore, these parameters were set to the same for all data sets: the number of generations (1,000), the inertia weights w_{max} and w_{min} (0.7 and 0.4), the initial acceleration coefficients c_1 and c_2 (1.5). The other various parameters settings of RBFN-MOPSO are presented in Table 2.

RESULTS AND DISCUSSION

Here, the results of study on RBF network based on MOPSO are presented. The experiments are conducted using two data sets. The results for each dataset are analyzed based on the convergence to Pareto optimal set and classification results. One advantage of evolutionary multi-objective optimization approach to RBF network generation is that accuracy of RBF networks with complexity can be obtained in one single run.

Table 3 and 4 showed that the best results of RBF network based on MOPSO algorithm on training, validation and testing in terms of convergence to error and classification accuracy for the breast cancer and diabetes data sets 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 for the two data sets. Figure 4a and b demonstrated that MOPSO has the capability to evolve compact RBF networks which generalize well on unseen data while avoiding over-fitting on data.

In order to evaluate the performance of RBFN-MOPSO under generalization and classification, the comparison was carried out by using RBFN-NSGA-II in terms of convergence to error and classification accuracy for all data sets. Table 3 also showed that the results of RBFN-NSGA-II on training, validation and testing in terms of convergence to

Table 3: Results of RBFN-MOPSO and RBFN-NSGA-II for all data sets

	RBFN-MOPSO			RBFN-NSGA-II		
Data set	Training error	Validation error	Testing error	Training error	Validation error	Testing error
Breast cancer	0.0314	0.0179	0.0063	0.0355	0.0231	0.0103
Diabetes	0.1755	0.1649	0.1497	0.1878	0.1787	0.1712

Table 4: Classification results of RBFN-MOPSO and RBFN-NSGA-II for all data sets

	RBFN-MOPSO			RBFN-NSGA-II		
Data set	Training accuracy	Validation accuracy	Testing accuracy	Training accuracy	Validation accuracy	Testing accuracy
Breast cancer	95.91	97.66	100.00	95.32	97.66	99.41
Diabetes	72.92	73.96	80.21	70.83	70.83	74.48

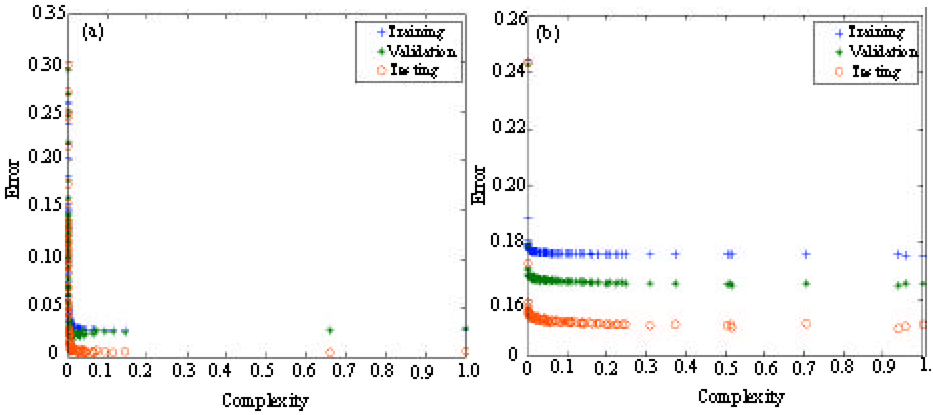

Fig. 4: Pareto front obtained using RBFN-MOPSO for data, (a) Breast cancer and (b) Diabetes

error for all data sets respectively. Figure 5a and b illustrated that Pareto front obtained using RBFN-NSGA-II for breast cancer and diabetes data sets respectively. From Table 3 and Fig. 5, the RBFN-NSGA-II results are weak for all data sets because the test error has the trend to oscillate even when the complexity increases. This event can be considered as that the resulting networks are over-fitted. From the comparison, Fig. 4-6 showed that RBFN-MOPSO is better than RBFN-NSGA-II in convergence to Pareto front solutions and improved generalization while avoiding over-fitting for all data sets.

When, we minimize both error and complexity of the network in multi-objective approach, we are able to achieve a number of Pareto optimal solutions with complexity ranging from simple networks to highly complex ones. From all data sets, we can conclude that by trading off accuracy against complexity, the Pareto based multi-objective optimization algorithm is able to find the simplest structures that solve the problem best. Besides, the simple Pareto optimal networks are able to generalize well on unseen data.

Moreover, from Fig. 4-6, we can see that when the network complexity increases, the testing error decreases. This phenomenon can be observed from the results by all of the selected testing approaches.

However, this phenomenon is only partially maintained for the relationship between the test performance and the network complexity. Test error still decreases as the network complexity increases. After that, the test error has the tendency to fluctuate even when value of weights increases. This occurrence can be considered as that the resulting networks are

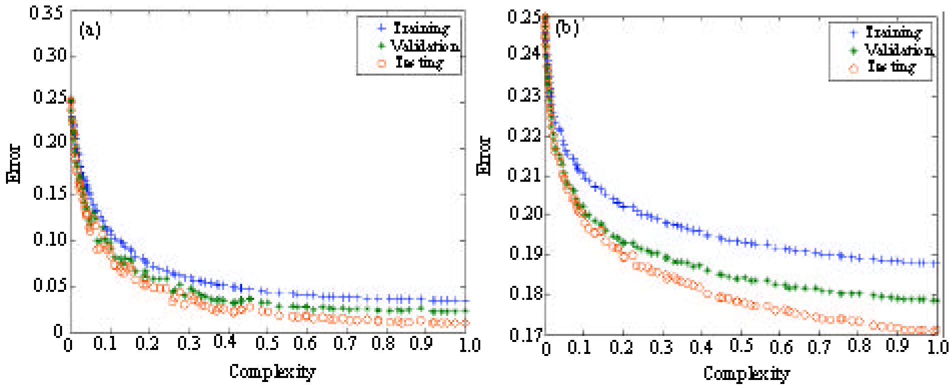

Fig. 5: Pareto front obtained using RBFN-NSGA-II for data, (a) Breast cancer and (b) Diabetes

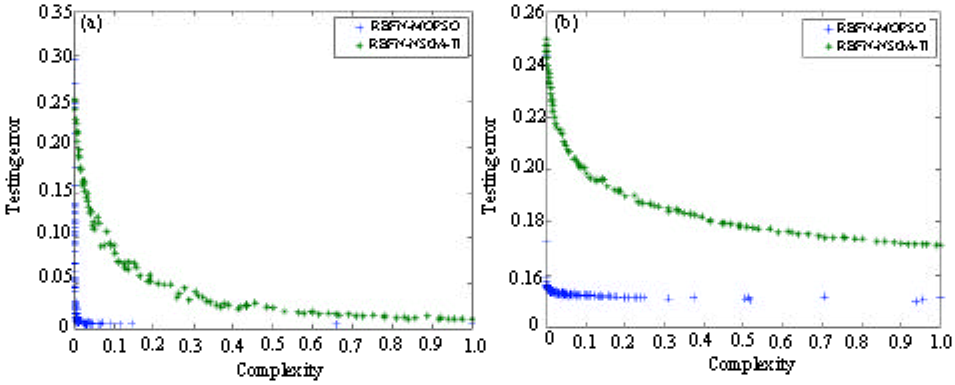

Fig. 6: Accuracy versus complexity of Pareto front of testing data for (a) Breast cancer and (b) Diabetes

over-fitted in RBFN-NSGA-II. Note that RBF network based on NSGA-II solutions are far from the Pareto optimal solutions and therefore it could still be minimized considering both objectives.

From Fig. 6a and b, it is very difficult to find a single optimal network that can offer the best performance for all datasets. Therefore, instead of searching for a single optimal RBF network, an algorithm that can result in a near complete set of optimal networks can be more reasonable and applicable option. This is the essential reason that MOPSO algorithms can be justified for this type of neural network design problems.

The classification accuracy results of RBF network based on MOPSO and NSGA-II for breast cancer and diabetes data sets are shown in Table 4. The accuracy in the table refers to the percentage of correct classification on training, validation and testing data sets, respectively.

Figure 7a and b showed that the correct classification of Pareto front solutions for testing data which are produced with RBFN-MOPSO and RBFN- NSGA-II for all data sets. Table 4 showed that RBF networks based on MOPSO have higher classification accuracy than RBFN- NSGA-II for diabetes data set. RBFN-MOPSO has best classification accuracy

Table 5: Results of computational time (in seconds) of RBFN-MOPSO and RBFN-NSGA-II for all data sets

Data set	RBFN-MOPSO	RBFN-NSGA-II
Breast cancer	97.298	299.388
Diabetes	107.903	183.569

Table 6: Comparison between our proposed approach with the various work in terms of testing classification accuracy and testing error and the best results are highlighted in bold

	Breast cancer		Diabetes	
Method/reference	Accuracy	Error	Accuracy	Error
RBFN-MOPSO	100.00	0.0063	80.21	0.1497
RBFN-NSGA-II	99.41	0.0103	74.48	0.1712
HMOEN_L2 (Goh *et al.*, 2008)	96.30	-	78.50	-
HMOEN_HN (Goh *et al.*, 2008)	96.82	-	75.36	-
MPANN (Abbass, 2003)	98.10	-	74.90	-
MPANN (Abbass, 2001)	-	-	-	0.251
IPSONet (Yu *et al.*, 2007)	97.07	0.0127	77.63	0.230
MOBNET (Garcia-Pedrajas *et al.*, 2002)	-	-	77.85	-

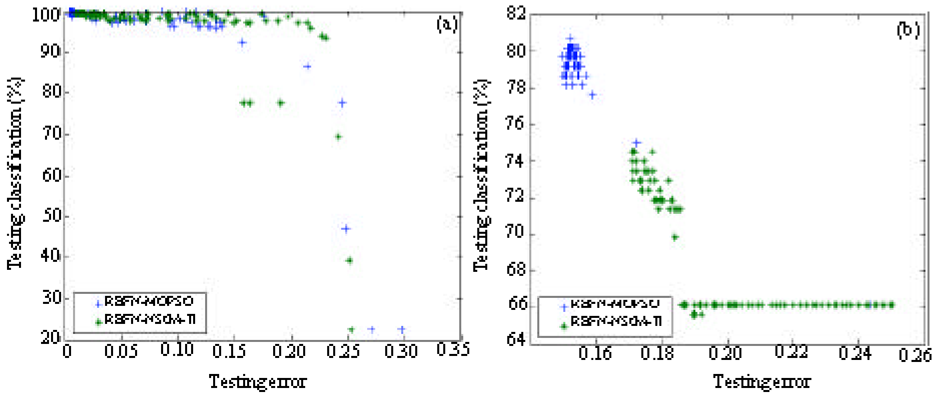

Fig. 7: Correct classification of Pareto front of testing data for (a) Breast cancer and (b) Diabetes

and maintains diversity of Pareto fronts solutions. However, RBFN-NSGA-II provides competitive classification accuracy compared to RBFN-NSGA-II for breast cancer data set (Fig. 7).

From Table 5, the results show that RBFN-MOPSO is significantly faster convergence time than RBFN-NSGA-II for all datasets. This is attributed to the crowding distance used by MOPSO which can be computed faster than elitism used by NSGA-II.

Having validated the effectiveness of multi-objective optimization, the performances of RBFN-MOPSO and RBFN-NSGA-II are compared against other works in the literature using these data sets. The summary of the results is shown in Table 6. We note that comparisons between the results obtained from different approaches have to be made cautiously, as there are numerous ways in which the experimental and simulation setups are done, for example, the training/testing ratio, the pre/postprocessing, etc. The results that are presented here are not fine-tuned in any manner, i.e., the same parameter and experimental settings are used for all the data sets. Nonetheless, it can be observed that proposed approach is better or at least competitive for breast cancer and diabetes data sets. Breast cancer results are proposed by RBFN-NSGA-II showed that better than HMOEN_L2 (Goh *et al.*, 2008), HMOEN_HN

(Goh *et al*., 2008), MPANN (Abbass, 2003) and IPSONet (Yu *et al*., 2007) except RBFN-MOPSO in terms of classification accuracy and generalization error while diabetes results is outperformed by HMOEN_L2 (Goh *et al*., 2008) except RBFN-MOPSO. On the other hand, RBFN-NSGA-II and MPANN (Abbass, 2003) perform poorly for diabetes in term of classification accuracy. In term of generalization error, RBFN-NSGA-II provided smaller error than MPANN (Abbass, 2001) and IPSONet (Yu *et al*., 2007) except RBFN-MOPSO for diabetes data set. We notice here that RBFN-MOPSO optimized network performance while improving its generalization ability. The results can be observed from Table 6 that the proposed approach RBFN-MOPSO is the best results for all data sets.

CONCLUSION AND FUTURE WORK

In this study, a novel approach for multi-objective optimization based on swarm intelligence principles, called MOPSO, is proposed and applied to develop generalization and classification accuracy for multi-objective RBF network. This study introduces multi-objective PSO approach to RBF network design called Multi-objective PSO-RBF network Optimizer to concurrently optimize the architectures and connections of network. The optimal RBF network is constructed from Pareto front set obtained by MOPSO. RBF network structure and its parameters are encoded to particle and Pareto-optimal set of RBF networks is obtained by MOPSO based on two criteria, i.e. model accuracy and complexity. The benchmark of pattern classification indicates that our proposed method provides better results than multi-objective GA in terms of yielding diverse of solutions along the true Pareto optimal fronts. The main advantages of the proposed MOPSO approach are that it is simple algorithm, faster convergence and yet robust in yielding efficient Pareto frontiers. Hence it can be concluded that, for RBF networks, the proposed technique is a viable tool for multi-objective analysis. 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.

ACKNOWLEDGMENTS

This work is supported by Ministry of Higher Education (MOHE) under Fundamental Research Grant Scheme (Vote No. 78183). The 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 incisive comments in making this study a success.

REFERENCES

Abbass, H.A., 2001. A memetic Pareto evolutionary approach to artificial neural networks. Proceedings of the Australian Joint Conference Artificial Intelligence, Dec. 10-14, LNCS, Berlin, Springer-Verlag, Germany, pp: 113-152.

Abbass, H.A., 2003. Speed up backpropagation using multiobjective evolutionary algorithms. Neural Comput., 15: 2705-2726.

Asuncion, A. and D. Newman, 2007. UCI machine learning repository. URL: http://www.ics.uci.edu/~mlearn/MLRepository.html.

Bai, Y. and L. Zhang, 2002. Genetic algorithm based self-growing training for RBF neural networks. Proceedings of International Joint Conference on Neural Networks, May 12-17, Honolulu, HI, USA., pp: 840-845.

Coello, C.A.C. and M.S. Lechuga, 2002. MOPSO: A proposal for multiple objective particle swarm optimization. Proceedings of the 2002 Congress on Evolutionary Computation part of the 2002 IEEE World Congress on Computational Intelligence, May, 12-17, Hawaii, pp: 1051-1056.

Deb, K., A. Pratap, S. Agarwal and T. Meyarivan, 2002. A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Tran. Evol. Comput., 6: 182-197.

Ferreira, P.M., A.E. Ruano and C.M. Fonseca, 2005. Evolutionary multiobjective design of radial basis function networks for greenhouse environmental control. Proceedings of the 16 th Ifac World Congress, Paper 4459.

Garcia-Perdrajas, N., C. Hervas and J. Munoz-Perez, 2002. Multi-objective cooperative coevolution of artificial neural networks. Neural Netw., 15: 1259-1278.

Goh, C., E. Teoh and K.C. Tan, 2008. Hybrid multiobjective evolutionary design for artificial neural networks. IEEE Trans. Neural Netw., 19: 1531-1548.

González, J., I.P.H. Rojas and J. Ortega, 2001. RBF Neural Networks, Multiobjective Optimization and Time Series Forecasting. Springer-Verlag, Berlin, ISBN: 3-540-42235-8, pp: 498-505.

Hatanaka, T., K. Uosaki and T. Kuroda, 2001. Structure selection of RBF neural network using information criteria. Proceedings of 5th International Conference on Knowledge-Based Intelligent Information Engineering Systems and Applied Technologies, Sept. 6-8, Japan, pp: 531-532.

Jin, Y., T. Okabe and B. Sendhoff, 2004. Neural network regularization and ensembling using multi-objective evolutionary algorithms. Proceedings of Congress on Evolutionary Computation, Jun. 19-23, Portland, pp: 1-8.

Kennedy, J. and R.C. Eberhart, 1995. Particle swarm optimization. Proceedings of IEEE International Conference on Neural Networks, Nov. 27-Dec. 1, Piscataway, pp: 1942-1948.

Kokshenev, I. and A.P. Braga, 2008. A multi-objective approach to RBF network learning. Neurocomputing, 71: 1203-1209.

Kondo, N., T. Hatanaka and K. Uosaki, 2004. Pareto RBF networks based on multiobjective evolutionary computation. SICE Annual Conference in Sapporo, Aug. 4-6, Society of Instrument and Control Engineers, Tokyo, Japan, pp: 2177-2182.

Kondo, N., T. Hatanaka and K. Uosaki, 2006. Pattern classification by evolutionary RBF networks ensemble based on multi-objective optimization. International Joint Conference on Neural Networks, Jul. 16-21, Busan, Korea, pp: 2919-2925.

Kondo, N., T. Hatanaka and K. Uosaki, 2007. Nonlinear dynamic system identification based on multiobjectively selected rbf networks. Proceedings of the 2007 IEEE Symposium on Computational Intelligence in Multicriteria Decision Making, Apr. 1-5, Honolulu, HI., pp: 112-127.

Lefort, V., C. Knibbe, G. Beslon and J. Favrel, 2006. Simultaneous Optimization of Weights Springer-Verlag, Berlin, pp: 49-60.

Leonard, J.A. and M.A. Kramer, 1991. Radial basis function networks for classifying process faults. Cont. Syst. Mag., 11: 31-38.

Nelles, O., 2001. Nonlinear System Identification. Springerlink, New York.

Raquel, C.R. and P.C. Naval, 2005. An effective use of crowding distance in multiobjective particle swarm optimization. Proceedings of the Conference on Genetic and Evolutionary Computation, Jun. 25-29, ACM, New York, USA., pp: 257-264.

Shi, Y. and R. Eberhart, 1998. A modified particle swarm optimizer. Proceedings of the IEEE Congress on Evolutionary Computation, May 4-9, Piscataway, New Jersey, pp: 69-73.

Wenbo, Z., H. De-Shuang and Y. Ge, 2002. The structure optimization of radial basis probabilistic neural networks based on genetic algorithms. Proceedings of International Joint Conference on Neural Networks, May, 12-17, Hilton Hawaiian Village Hotcl, Honolulu Hawaii, pp. 1086-1091.

Yen, G.G. and H. Lu, 2003. Hierarchical rank density genetic algorithm for radial-basis function neural network design. Int. J. Comput. Intell. Appl., 3: 213-232.

Yen, G.G., 2006. Multi-Objective Evolutionary Algorithm for Radial Basis Function Neural Springer-Verlag, Berlin, ISBN: 978-3-540-30676-4, pp: 221-239.

Yu, L., K.K. Lai, S. Wang and W. Huang, 2006. A Bias Variance-Complexity Trade-off Framework for Complex System Modeling. Springer-Verlag, Berlin, pp: 518-527.

Yu, J., L. Xi and S. Wang, 2007. An improved particle swarm optimization for evolving feedforward artificial neural networks. Neural Proc. Lett., 16: 217-231.