International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:9, No:2, 2015

A Multi-Objective Evolutionary Algorithm of Neural Network for Medical Diseases Problems

Sultan Noman Qasem

***Abstract*—**This paper presents an evolutionary algorithm for solving multi-objective optimization problems-based artificial neural network (ANN). The multi-objective evolutionary algorithm used in this study is genetic algorithm while ANN used is radial basis function network (RBFN). The proposed algorithm named memetic elitist Pareto non-dominated sorting genetic algorithm-based RBFN (MEPGAN). The proposed algorithm is implemented on medical diseases problems. The experimental results indicate that the proposed algorithm is viable, and provides an effective means to design multi-objective RBFNs with good generalization capability and compact network structure. This study shows that MEPGAN generates RBFNs coming with an appropriate balance between accuracy and simplicity, comparing to the other algorithms found in literature.

***Keywords*—**Radial basis function network, Hybrid learning, Multi-objective optimization, Genetic algorithm.

I. Introduction

ARTIFICIAL Neural Networks (ANNs) are commonly used for pattern classification problems. ANNs represent an information-processing paradigm that is inspired by the way biological nervous systems process information. ANNs [1] have been an object of interest in statistics and computer science. Radial basis function Networks (RBFNs) are typical ANNs, and they were introduced into neural network literature by [2] as a means to observe local responses in biological neurons. RBFNs have a number of advantages over other types of ANNs, and these include better approximation capabilities, simpler network structures and faster learning algorithms. Fundamentally, there are many important aspects that influence the quality of an RBFN such as its structure and generalization capability. However, the construction of a quality RBFN to reduce generalization error can be a time-consuming process as the modeler must select both a suitable set of inputs - the inputs are given in the problem and a suitable RBFN structure.

Because some of the traditional algorithms such as back-propagation (BP) still suffer from slow convergence and long training time [3]-[5], there is a clear need to develop sophisticated solutions to improve learning characteristics. In addition, BP and its variants are based on gradient-descent convergence algorithms and can easily become stuck at a local minimum [3]. The key problem with BP and other traditional learning algorithms is the choice of a correct architecture (i.e., number of hidden nodes). Hence, evolutionary algorithms (EAs) are used to train ANNs (that use a single error function) to solve the above problems. An EA still has a number of parameters to tune, similar to tuning BP algorithm parameters. The advantage of using EAs instead of BP is not in reducing the number of parameters to tune. A major advantage of using an EA is its ability to escape from local minimum, its robustness and its ability to adapt itself to a changing environment. Selecting the structure of ANNs is a difficult issue. The major disadvantage of using EAs in ANN applications is high computational cost. Therefore, hybrid algorithms are used to speed up the convergence by augmenting EAs with a local search feature such as BP (also known as a memetic approach). The literature on use of EAs in ANNs does not emphasize the trade-off between the structure and the generalization ability of an EA network. A network with more hidden nodes may learn a training set more quickly, but it may not generalize well on a testing set. This trade-off is a well-known problem in the multi-objective optimization problem (MOP) where a trade-off exists between the structure of the network and generalization error. Multi-objective techniques offer the potential advantage of helping a learning algorithm to escape a local minimum, therefore improving the accuracy of the learning model [4], [6].

Current work provides training in RBFNs based on multi-objective evolutionary algorithms (MOEAs). Kokshenev and Braga [8] proposed a deterministic global solution to a multi-objective problem of supervised learning applied to an RBFN using nonlinear programming. A multi-objective optimization algorithm [7] has been applied to the problem of inductive supervised learning based on smoothness of apparent complexity measures for RBFNs. However, the computational complexity of their algorithm is high in comparison with other state-of-the-art machine learning algorithms. A multi-objective genetic algorithm based design procedure for the RBFN has been proposed in [9]. In addition, a hierarchical rank density genetic algorithm (HRDGA) has been developed to evolve both the neural network's topology and its parameters simultaneously. An RBFN ensemble [10] has been constructed from a Pareto-optimal set obtained by multi-objective evolutionary computation. A Pareto-optimal set of RBFNs was based on three criteria: model complexity, representation ability and model smoothness. An EA, RBF-Gene, was applied to optimize RBFNs [11]. Unlike other works, this algorithm can evolve both the structure and the numerical parameters of the network. In fact, it can evolve a number of neurons and their weights.

Sultan Noman Qasem was with the Department of Computer Science, Faculty of Applied Sciences, Taiz University, Taiz, Yemen. He is now with the Department of Computer Science, College of Computer and Information Sciences, Al Imam Mohammad Ibn Saud Islamic University (IMSIU), Riyadh, 11432 Saudi Arabia (e-mail: sultann.noman@ccis.imamu.edu.sa).

González et al. [12] presented RBFN optimization from training examples as a multi-objective problem and proposed an EA to solve it. This algorithm incorporates mutation operators to guide the search towards good solutions. An algorithm of obtaining a Pareto-optimal RBFN set based on MOEAs has been proposed in [13]. On the other hand, Ferreira et al. [14] proposed a multi-objective algorithm for use with RBFN 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. Qasem and Shamsuddin [15] proposed time variant particle swarm optimization (TVMOPSO) to simultaneously evolve accuracy and connections (centers, widths and weights) of an RBFN, whose structure was fixed, for medical diseases diagnosis.

The main contribution of this paper is to resolve the hybrid learning problem (unsupervised and supervised learning) of an RBFN using memetic elitist Pareto non-dominated sorting genetic algorithm (MEPGAN) and to obtain simple and accurate RBFNs. This mechanism evolves toward the Pareto-optimal front defined by several objective functions involving model accuracy and complexity. The algorithm provides network simplicity with faster convergence to Pareto-optimal solutions. The proposed algorithm is applied to solve medical classification problems. The proposed algorithm has been empirically analyzed, justifying the strategy for medical diseases problems. In addition, the proposed algorithm is compared to Pareto-based multi-objective algorithms described in the literature. The comparison with other algorithms is implemented to examine the efficiency and effectiveness of our proposed algorithm.

The structure of this paper is organized as follows: Section II introduces the fundamentals of the proposed algorithm. In Section III, the proposed MEPGAN is described in detail. Section IV reports the experiments on the medical diseases problems. The significance of the results is presented in Section V. Finally, Section VI presents the summary and concluding remarks and future work.

II. Background Materials

A. RBFN

An RBFN is a feed-forward neural network with three layers: an input layer with n nodes, a hidden layer with p neurons or RBFs, and an output layer with one or several nodes. Each input node corresponds to a feature of the input pattern. The activation function of the hidden layer neurons is the RBF, which is a Gaussian function in this study.

An RBFN is a feed-forward network with a single layer of hidden units. The RBF output from a hidden unit shows the maximum value at its center point and decreases its output value as the input leaves the center. An RBFN uses a three layer feed-forward, fully connected network, and applies the RBF only to neurons in the hidden layer. The output layer is composed of linear units, and the connections of only the output layer are weighted, while the connections from the input to the hidden layer are not weighted. In the input layer, the number of neurons is the same as the number of input dimensions. The input layer neurons transmit data to the hidden layer, where the RBF is calculated based on the signals from the input layer. These values are transmitted to the output layer, which calculates the linear sum of the hidden neuron values. Let $\Phi_j(x)$ be the j[th] RBF. $\Phi_j(x)$ defined as:

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

Here, $x = (x_1, x_2. \ . \ . \ x_d)^T$ is the input vector, and $c_j = (c_{1j}, c_{2j},\ldots,c_{dj})^T$ and σ_j^2 are the center vector and the width parameter, respectively. The output of RBFN y is the linear sum of RBFs,

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

where y is the output of the RBFN, p is the number of hidden layer neurons, and w_j is the weight from the j[th] neuron to the output layer.

B. Genetic Algorithm

Genetic algorithm (GA) is a stochastic search algorithm for solving optimal solutions within large and complicated search spaces. It's a popular type of EA that has been successfully used for optimization problems. The technique is based on ideas from Darwin's theory of natural selection and "survival of the fittest". GA is general purpose optimization algorithms developed by [16]. They are based on principles of natural evolution. In this algorithm, a population of individuals (chromosomes) undergoes a sequence of transformation by means of genetic operators to form a new population. Two operators are mutation and crossover. Mutation creates new individuals by a small change in a single individual and crossover creates new individuals by combining parts of two individuals.

GA can be seen as a unique kind of search strategy, whereby in GA, there is a set of candidate solutions to a problem, and makes it evolve by iteratively applying a set of stochastic operators. It is a procedure that is loosely based on the principle "survival of the fittest". A series of solutions will be generated and applied to a large amount of data. The solutions generated will be evaluated based on a "fitness" formula. The "fitter" child will survive while the "poorer" child will be eliminated. Stochastic operators define the processes in biological evolution, which consist of selection, crossover and mutation.

C. Pareto Optimization

The "optimality" of a solution has to be redefined in multi-objective optimization (MOO), giving rise to the concept of Pareto-optimality because of the multi-criteria nature of the problems. In contrast to the single-objective optimization (SOO) problem, MOO problems are characterized by trade-offs, leading to many Pareto-optimal solutions.

Many real world problems involve multiple measures of performance or objectives, which should be optimized simultaneously. MOO functions by seeking to optimize the component of a vector-valued objective function. Unlike SOO, the solution to a MOO problem is a family of solutions known as the Pareto-optimal set. A general problem of *M* objectives can be mathematically stated as:

Minimize/Maximize $f_i(x)$ i=1,…, *M*

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

where f_i, g_j, and h_k are the i^{th} objective function, j^{th} equality, and k^{th} inequality constraints, respectively, x is a decision vector that represents a solution, and *M*, *N*, and *K* are the number of objectives, equality, and inequality constraints, respectively.

The MOO problem is reduced to finding x such that $f_i(x)$ is optimized. In general, the objectives of the optimization problem are often conflicting. A suitable solution to problems that involve conflicting objectives should offer acceptable – though possibly sub-optimal in the single objective sense – performance in all objective dimensions, where acceptability is problem dependent, and ultimately a subjective concept. An important concept in MOO is that of dominance, where a solution x_i is said to dominate another solution x_j if the two following conditions are true: the solution x_i is not worse than x_j in all objectives and the solution x_i is strictly better than x_j in at least one objective. The set of all such non-dominated solutions is called the Pareto-optimal set or the non-dominated set.

III. Methodology

A. RBFN Encoding

Encoding is the first step when the EA is applied to a particular problem. According to the characteristics of the RBFN, the real-encoded genotype representation can make the search of the solution space more precise and efficient. Thus, the individual used in this study is a record that contains one matrix ω and one vector ρ. The matrix ω is of dimension $(I+1)\times H+(H+1)\times O$. *I*, *H*, *O*, and $w_{ij} \in \omega$ are the numbers of input, hidden, and output neurons, and the connections (centers, widths, and weights) connecting unit *i* with unit *j*, respectively. The vector ρ is of dimension *H*, where $\rho_h \in \rho$ is a binary value used to indicate whether the hidden node *h* exists in the network or not; that is, it works as a switch to turn a hidden node on or off. It also allows connections of disabled nodes to still be trained even though they are not used during network performance valuation. The sum, $\sum_{h=1}^{H} \rho_h$ represents the actual number of hidden nodes in a network, where *H* is the maximum number of hidden nodes.

B. Pareto Learning Problem

The ANN's design is cast as an MOO problem where a number of objectives such as training accuracy and degree of complexity are specified. The conflicting objectives of maximizing network capacity and minimizing network complexity are manifested in the trade-offs between training and test accuracy. The multi-objective learning problem for a single hidden-layer RBFN can be formulated as two objective functions that are used to evaluate the network's performance. The two objective functions for minimization problems are as follows:

1. Accuracy based on Mean-Squared Error (MSE) on the training set.

$$f_1 = \frac{1}{N}\sum_{j=1}^{N}\left(t_j - o_j\right)^2, \quad (4)$$

where o_j and t_j are the network output and the desired output, and *N* is the number of samples used for training purposes.

2. Complexity is computed based on the number of hidden nodes in the hidden layer of RBFNs. The equation is given as

$$f_2 = \sum_{h=1}^{H} \rho_h, \quad (5)$$

where $\rho_h \in \rho$ is a binary value used to indicate whether the hidden node *h* exists in the network or not and *H* is the maximum number of hidden nodes present in the RBFN. The vector ρ is of dimension *H*.

C. MEPGAN

The proposed algorithm is a multi-objective optimization approach for RBFN training called MEPGAN. A MOEA with a local search algorithm is presented in this section. The MOEA proposed is based on the NSGA-II algorithm [17] and the local search algorithm is BP. The Memetic Elitist Pareto non-dominated sorting genetic algorithm based RBFN (MEPGAN) is implemented. The network architecture and accuracy are evolved simultaneously with each individual being a fully specified RBFN. In this study, MEPGAN has been proposed to determine the best centers, widths and weights of RBFN and the corresponding structure of the network.

MEPGAN begins by collecting, normalizing and reading the dataset. The number of hidden nodes and maximum number of iterations are then set. Next, the individual length is computed. In addition, the parameters of RBFN are determined by the traditional algorithms. A population of RBFNs is then generated and initialized. Every individual is evaluated every iteration based on objective functions. The proposed algorithm stops and outputs a set of non-dominated RBFNs after the maximum iterations is reached. The MEPGAN is presented as below:

1. Generate a random population *P(t)* at t = 0 of size *N*, where each individual presents a RBFN and where *t* is the number of the actual generation.
2. Evaluate the individuals *P(t)*on basis of two objectives (accuracy and complexity/minimum of hidden nodes).
3. Use the fast-non-dominated-sort for obtaining a list *F* with the fronts of the population *P(t)*.
4. Assign to each individual a rank value equal to his dominance level and a crowding distance value.
5. While stopping criterion is not met do

a. Use binary tournament for selecting *N* individuals of *P(t)*, according to their rank and crowding distance value.
b. Perform single-point crossover and bitwise mutation for binary encoding and the simulated binary crossover (SBX) operator and polynomial mutation [18] for real encoding for individuals of *P(t)* selected to generate a new offspring population *Q(t)* of size *N*.
c. Apply local search (BP) to each of the individuals and evaluate the individuals of population *Q(t)* on basis of accuracy and complexity.
d. f = 1
e. R(t) = P(t) ∪ Q(t).
f. F = fast-non-dominated-sort (*R(t)*).
g. While size of population P(t+1) is < N to do
   i. Calculate s crowding-distance for front F^f
   ii. P(t+1) = P(t+1) ∪ F^f
   iii. f = f + 1
h. End while
i. Sort population *P(t+1)* according to their rank and crowding value and select the first *N* individuals. The new population *P(t+1)* of size *N*is now completed.
j. t = t + 1

6. End while

IV. Experimental Studies

To evaluate the performance of the proposed algorithm on various medical diseases problems, several experimental results involving machine learning datasets are presented. They are real-world problems that differ with respect to the number of available patterns, attributes, and classes. These problems have been the subject of many studies in ANNs and machine learning. The results of the proposed algorithm for each medical diseases problem are compared to other studies previously reported in the literature and the results are analyzed based on the convergence to the Pareto-optimal set and their overall classification performance.

A. Data Sets

A total of three real-world datasets concerning medical diseases problems, three medical diseases datasets obtained from the University of California at Irvine (UCI) machine learning benchmark repository [19] were considered. The datasets are used to validate the proposed algorithm. Breast cancer, diabetes, and heart datasets represent medical diseases classification problems. Table I summarizes the pertinent details of the datasets such as the number of features and the number of patterns.

TABLE I
Description of Datasets

Dataset	#features	#classes	#patterns
Wisconsin breast cancer	9	2	699
Pima diabetes	8	2	768
Heart	13	2	303

B. Experimental Setup

The proposed algorithm is evaluated by random 10-fold cross-validation for all datasets. In 10-fold cross-validation, the dataset is first divided into ten subsets of equal size. One subset is used as the testing dataset, and the other nine subsets are used as the training datasets. This train and test process is repeated so that all subsets are used as a testing dataset. The training set is used to train the network in order to get the Pareto-optimal solutions while the testing set is used to test the generalization performance of the Pareto RBFN and is not seen by any individual Pareto RBFN during the training process. The performance of the proposed algorithm is evaluated by investigating statistics from ten evaluations. In the experiment, we analyze the evolutionary process of the proposed MEPGAN and evaluate its performance on all medical diseases problems.

For each dataset, the number of input and output nodes is problem-dependent but the maximum number of hidden nodes is 10 [4], [20]. The maximum number of iterations is set to 1000 for all datasets. There are some parameters of the proposed algorithm that must be specified by the user. These parameters were set equivalently for all medical diseases datasets: the learning rate for BP to 0.01 and the number of iterations for BP is set to 5 [4].

TABLE II
Parameters Settings of the Algorithms

RBFN Initialization	
Maximum number of hidden nodes	10
Initial weights	[-0.5,0.5]
MEPGAN	
Population size	100
Maximum number of iterations	1000
Probability of crossover (P_c)	0.9
Probability of mutation (P_m)	1/N
Distribution indices for crossover (η_c)	20
Distribution indices for mutation (η_m)	20
BP Algorithm	
Learning rate	0.01
Number of iterations	5

Other parameter settings of the proposed MEPGAN are presented in Table II. The "N" in this table refers to the dimension of the individual.

C. Results and Discussion

Table III presents the results of the proposed MEPGAN in terms of error and RBFN structure on breast cancer, diabetes, and heart datasets (the Mean, SD, Max and Min indicate the mean value, standard deviation, maximum value and minimum value, respectively). The results were obtained by random 10-fold cross-validation for all datasets. The aim of

this algorithm is to generate Pareto-optimal solutions to improve generalization on unseen data. The results demonstrate that MEPGAN has the capability to evolve compact RBFN that generalize well on unseen data.

TABLE III
RESULTS OF MEPGAN ON TRAINING AND TESTING SETS

Data set		MEPGAN		
		Training error	Testing error	RBFN size
Breast cancer	Mean	0.0263	0.0272	6.6
	SD	0.0052	0.0086	3.6
	Min	0.0212	0.0117	3.0
	Max	0.0398	0.0388	10.0
Diabetes	Mean	0.1751	0.1760	5.4
	SD	0.0054	0.0124	3.3
	Min	0.1670	0.1610	2.0
	Max	0.1829	0.2003	10.0
Heart	Mean	0.1583	0.1673	6.2
	SD	0.0370	0.0338	3.3
	Min	0.1226	0.1094	3.0
	Max	0.2230	0.2226	10.0
Average	Mean	0.1199	0.1235	6.1
	SD	0.0815	0.0835	0.6

The common performance measures used in classification of datasets are accuracy, sensitivity, and specificity. Classification accuracy may not always be the most significant performance criterion in a number of cases. In a classification problem, for example, sensitivity and specificity may outweigh accuracy, where emphasis is on generalization. Therefore, in order to evaluate the classification capabilities of the proposed algorithm, the performance of the proposed MEPGAN based on average sensitivity, specificity, and classification accuracy was tested, with the results shown in Table IV.

Table IV shows the statistical results for the sensitivity, specificity, and classification accuracy of the proposed MEPGAN on the training set and testing set for all medical diseases problems. From Tables III and IV, we can easily verify that in all datasets, on average, the MEPGAN algorithm provides promising results in both training and testing sets.

TABLE IV
RESULTS OF MEPGAN OF SENSITIVITY (SEN), SPECIFICITY (SPE) AND ACCURACY (ACC) ON TRAINING AND TESTING SETS

Dataset		MEPGAN					
		Training set			Testing set		
		SEN	SPE	ACC	SEN	SPE	ACC
Breast Cancer	Mean	97.02	97.15	97.10	96.25	97.08	96.78
	SD	0.32	0.31	0.23	4.14	1.50	1.49
	Min	96.74	96.75	96.91	87.50	95.45	92.75
	Max	97.67	97.74	97.56	100.0	100.0	98.53
Diabetes	Mean	45.46	87.98	73.21	45.20	87.11	72.78
	SD	11.58	4.80	1.55	9.06	8.31	4.56
	Min	28.63	80.44	70.33	33.33	76.00	64.94
	Max	62.66	94.67	75.25	62.96	98.00	79.22
Heart	Mean	66.86	89.38	78.98	66.76	89.38	79.07
	SD	29.08	7.11	10.43	30.62	8.36	11.72
	Min	0.00	78.47	53.73	0.00	81.25	55.17
	Max	84.68	100.0	85.07	92.86	100.0	96.67
Average	Mean	69.78	85.48	83.10	69.40	91.19	88.90
	SD	25.90	5.59	12.47	25.63	5.23	13.96

D. Comparisons with Other Works

For sound justifications, the performance of the proposed algorithm is compared against other algorithms in the literature using these datasets. The summary of the results is shown in Table V and Fig. 1. Table V compares the results of the proposed algorithm with other multi-objective evolutionary ANN algorithms found in the literature. The results that are presented here are not geared towards automatic parameter tuning, i.e., similar parameter and experimental settings are used for all datasets. It can be observed from the experiments that the proposed algorithm MEPGAN at least competitive for all datasets. Breast cancer results are outperformed by MPANN [4] while diabetes results are outperformed by MPENSGA2E [5]. On the other hand, MPANN performs competitive for diabetes dataset; while MPENSGA2E performs competitive for breast cancer datasets. However, HMOEN_L2 [21] performs competitive for breast cancer datasets.

MPANN [4] yielded the average Pareto optimal front of the ANN with the lowest training error for the breast cancer and diabetes datasets. While the HMOEN_L2 [21] and HMOEN_HN [21] results are based on the ANN with the best training accuracy on the dataset for each run. However, the proposed algorithm has given the average Pareto optimal front of RBFN with the smallest error on the training set for all datasets [3], [4]. In addition, Table V shows also the structure of the algorithms (number of hidden nodes or number of rules). It can be observed in terms of RBFN structure, the results of proposed algorithm are competitive or comparable to other algorithms.

V. CONCLUSION AND FUTURE WORK

This paper has proposed hybrid multi-objective learning algorithm called MEPGAN, to achieve a compact RBFN model with both good prediction accuracy and simple structure simultaneously. In this paper, we have proposed the use of the accuracy and the complexity of an RBFN as objectives that should be optimized when selecting the structure of the RBFN for medical diseases problems. MEPGAN is used to improve generalization and classification accuracy for the RBFN. The RBFN and its parameters are encoded to individual agents, and a Pareto-optimal set of RBFN is obtained.

The proposed algorithm, MEPGAN was experimented with for solving medical diseases problems. The experimental results illustrate that the MEPGAN algorithm was able to obtain an RBFN model with better classification accuracy and simpler structure on medical diseases problems compared to other training algorithms.

For the future work, the performance of the proposed algorithm will be improved through adding more objectives. The proposed algorithm will be implemented for other medical problems.

TABLE V
COMPARISON OF RESULTS OF THE PROPOSED AND SOME ALGORITHMS IN LITERATURES FOR ALL MEDICAL DISEASES PROBLEMS

Algorithm/Reference	Breast cancer		Diabetes		Heart	
	Accuracy	Structure	Accuracy	Structure	Accuracy	Structure
MEPGAN	96.78	6.6	72.78	5.4	79.07	6.2
C4.5 [23]	94.71	---	73.13	---	76.61	---
MPANN [4]	98.10	4.1	74.90	6.6	---	---
HMOEN_L2 [21]	96.26	4.7	78.48	7.5	79.69	9.9
HMOEN_HN [21]	96.82	4.8	75.36	8.1	81.06	9.7
MSCC [22]	97.60	2.0	76.50	25.0	---	---
MPENSGA2E [5]	95.87	---	78.99	---	---	---
MPENSGA2S [5]	95.60	---	76.96	---	---	---
RBFN-TVMOPSO [15]	96.53	2.0	78.02	2.0	---	---

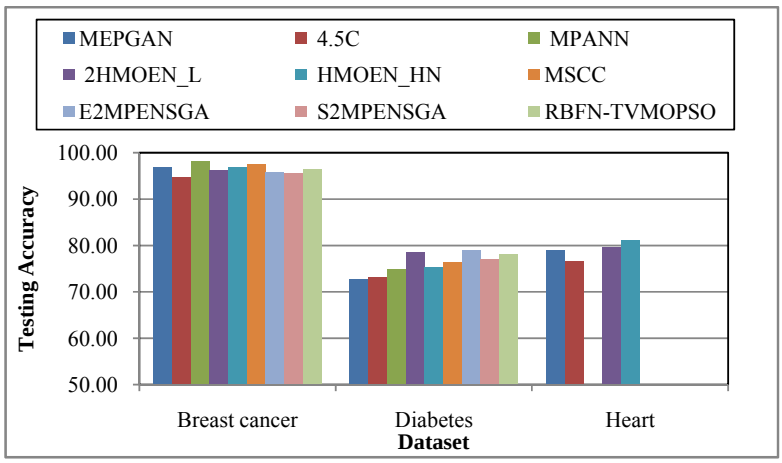

Fig. 1 Performance comparisons of proposed and existing algorithms for all datasets

REFERENCES

[1] C.M. Bishop, *Neural networks for pattern recognition*. Oxford, U.K.: Oxford Univ., Press, 1995.
[2] D. Broomhead, D. Lowe, "Multivariable functional interpolation and adaptive networks," *Complex Systems*, vol. 2, pp. 321–355, 1988.
[3] H.A. Abbass, "An evolutionary artificial neural networks approach for breast cancer diagnosis", *Artificial Intelligence in Medicine*, vol. 25, no. 3,pp. 265–281, 2002.
[4] H.A. Abbass, "Speed up backpropagation using multi-objective evolutionary algorithms," *Neural Computation*, vol. 15, no. 11,pp. 2705–2726, 2003.
[5] J.C. Caballero, F.J. Martínez, C. Hervás, P.A. Gutiérrez, "Sensitivity versus accuracy in multiclass problems using memetic Pareto evolutionary neural networks,"*IEEE Transactions on Neural Networks*, vol. 21, no. 5, pp. 750–770, 2010.
[6] R.A. Teixeira, A.P. Braga, R.R. Saldanha, R.H.C. Takahashi, "Improving generalization of MLPs with multi-objective optimization," *Neurocomputing*, vol. 35, pp. 189–194, 2000.
[7] I. Kokshenev, A.P. Braga, "A multi-objective approach to RBF network learning," *Neurocomputing*, vol. 71, no. 7–9 , pp. 1203–1209, 2008.
[8] I. Kokshenev, A. P. Braga, "An efficient multi-objective learning algorithm for RBF neural network,"*Neurocomputing*, vol. 73, no. 16–18), pp. 2799–2808, 2010.
[9] N. Kondo, T. Hatanaka, K. Uosaki, "Pattern classification by evolutionary RBF networks ensemble based on multi-objective optimization," *International Joint Conference on Neural Networks*, pp. 2919–2925, 2006.
[10] G.G. Yen, *Multi-objective evolutionary algorithm for radial basis function neural network design*. Springer-Verlag, vol. 16, 2006.
[11] V. Lefort, C. Knibbe, G. Beslon, J. Favrel, *Simultaneous optimization of weights and structure of an RBF neural network*. Springer-Verlag, LNCS, vol. 3871, pp. 49–60, 2006.
[12] J. González, I. Rojas, H. Pomares, J. Ortega, *RBF neural networks, multi-objective optimization and time series forecasting*. Springer-Verlag, LNCS, vol. 2084, pp. 498–505, 2001.
[13] N. Kondo, T. Hatanaka, K. Uosaki, "Non linear dynamic system identification based on multi-objectively selected RBF networks," *Proceedings of the IEEE Symposium on Computational Intelligence in Multi-criteria Decision Making*, pp. 112–127, 2007.
[14] P.M. Ferreira, A.E. Ruano, C.M. Fonseca, "Evolutionary multi-objective design of radial basis function networks for greenhouse environmental control," *Proceedings of the 16th IFAC World Congress*, 2005.
[15] S.N. Qasem, S.M. Shamsuddin, "Radial basis function network based on time variant multi-objective particle swarm optimization for medical diseases diagnosis," *Applied Soft Computing*, vol. 11, no. 1, pp. 1427–1438, 2011.
[16] J.H. Holland, *Adaptation in Natural and Artificial Systems*. University of Michigan Press, 1975.
[17] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, "A fast and elitist multi objective genetic algorithm: NSGA-II,"*IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182-197, 2002.
[18] K. Deb, R.B. Agrawal, "Simulated binary crossover for continuous search space," *Complex System*, vol. 9, pp. 115–148, 1995.
[19] A. Asuncion, D. Newman, UCI machine learning repository, URL (http://www.ics.uci.edu/~mlearn/MLRepository.html), 2007.
[20] Y. Jin, B. Sendhoff, "Pareto-based multi-objective machine learning: an overview and case studies," *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews*, vol. 38, no. 3, pp. 397–415, 2008.
[21] C. Goh, E. Teoh, K.C. Tan, "Hybrid multi-objective evolutionary design for artificial neural networks," *IEEE Transactions on Neural Networks*, vol. 19, no. 9, pp. 1531-1548, 2008.
[22] W. Cai, S. Chen, D. Zhang, "A multi-objective simultaneous learning framework for clustering and classification," *IEEE Transactions on Neural Networks*, vol. 21, no. 2, pp. 185–200, 2010.

[23] M.-L. Antonie, O.R. Zaiane, R.C. Holte, "Learning to use a learned model: a two stage approach to classification," *Proceedings of 6th International Conference of Data Mining*, pp. 33–42, 2006.