Pareto Genetic Design of GMDH-type Neural Networks for Nonlinear Systems

N. Nariman-zadeh and A. Jamali
Department of Mechanical Engineering, University of Guilan,
PO Box 3756, Rasht, Iran

nnzadeh@guilan.ac.ir

Abstract. *In this paper, Genetic Algorithms (GAs) are deployed for multi-objective Pareto optimal design of Group Method of Data Handling (GMDH)-type neural networks that have been used for modelling of a nonlinear system. In this way, GAs with a specific encoding scheme is firstly presented to evolutionary design of the generalized GMDH-type neural networks in which the connectivity configurations in such networks are not limited to adjacent layers. Multi-objective GAs with a new diversity preserving mechanism are secondly used for Pareto optimization of such GMDH-type neural networks. The important conflicting objectives of GMDH-type neural networks that are considered in this work are, namely, Training Error (TE), Prediction Error (PE) and Number of Neurons (N) of such neural networks. It is shown that the obtained non-dominated Pareto points are inclusive of those which can be found using Akaike's Information Criterion (AIC) for both training and prediction errors .Moreover, an important trade-off can be discovered by such Pareto optimum approach to the design of GMDH-type neural networks which helps a designer to select a network compromisingly.*

Keywords

Multi-objective optimization, Genetic algorithms, GMDH, Pareto.

1 Introduction

Group Method of Data Handling (GMDH) algorithm is a self-organizing approach by which gradually complicated models are generated based on the evaluation of their performances on a set of multi-input-single-output data pairs (i=1, 2, …, M). The GMDH was first developed by Ivakhnenko [1] as a multivariate analysis method for complex systems modelling and identification. In this way, GMDH was used to circumvent the difficulty of knowing a priori knowledge of mathematical model of the process being considered. Therefore, GMDH can be used to model complex systems without having specific knowledge of the systems. The main idea of GMDH is to build an analytical function in a feedforward network based on a quadratic node transfer function [2] whose coefficients are obtained using regression technique [3]. In recent years, however, the use of such self-organizing networks leads to successful application of the GMDH-type algorithm in a broad range of areas in engineering, science, and economics [3-5].

The inherent complexity in the design of feedforward neural networks in terms of understanding the most appropriate topology and coefficients has a great impact on their performance. There have been extensive efforts in recent years to deploy population-based stochastic search algorithms such as evolutionary methods to design artificial neural networks since such evolutionary algorithms are particularly useful for dealing with complex problems having large search spaces with many local optima [3]. A very comprehensive review of using evolutionary algorithms in the design of artificial neural networks can be found in [6]. Recently, genetic algorithms have been used in a feedforward GMDH-type neural network for each neuron searching its optimal set of connection with the preceding layer [5]. In this reference, authors have proposed a hybrid genetic algorithm for a simplified GMDH-type neural network in which the connection of neurons are restricted to adjacent layers. All these methods devised previously have been based on single objective optimization process in which either training error or prediction error selected to be minimized with no control of other objectives. In order to consider the complexity of such networks, a Minimum Description Length (MDL) approach has been used in [iba] to involve a tradeoff between fitness of training error and the number of parameters in the network (complexity). Recently, Akaike's Information Criterion (AIC) for prediction error of the network has been used in [7] to tradeoff such objective functions in the design of a revised GMDH-type neural network. However, in order to obtain more robust models of such complex process, it is required to consider all the non-commensurable conflicting objectives, namely, training error (TE), prediction error (PE) and number of neurons (N) (representing the complexity of the models) be minimized simultaneously in the sense of multi-objective Pareto optimization process. The obtained Pareto front shown in different plane of those objectives would visualize the existed tradeoffs which, therefore, help the designer to compromise and choose the appropriate network.

In Multi-objective optimization problems (MOPs), there are several objective or cost functions (a vector of objectives) to be optimized (minimized or maximized) simultaneously. These objectives often conflict with each other so that improving one of them will deteriorate another. Therefore, there is no single optimal solution as the best with respect to all the objective functions. Instead, there is a set of optimal solutions, known as Pareto optimal solutions or Pareto front [8-9] for multi-objective optimization problems. A very good and comprehensive survey of these methods has been presented in [9]. In addition to its popularity and effectiveness, NSGA-II [8] has been modified in [10] to enhance its diversity preserving mechanism which will be used in this work.

In this paper, EAs with a new encoding scheme are used to evolutionary design the generalized structure GMDH-type (GS-GMDH) neural networks in which the connectivity configuration in such networks is not limited to adjacent layers for modelling and prediction of a nonlinear system. In this way, multi-objective EAs (non–dominated sorting genetic algorithm, NSGA-II) with a new diversity preserving mechanism are applied for Pareto optimization of such GS-GMDH-type neural networks. The important conflicting objectives of the GS-GMDH neural networks that are considered in this work are, namely, training error (TE), prediction error (PE) and number of neurons (N). The total numbers of randomly generated data are 100 from which 50 are randomly used for evaluations of TE whilst the remaining 50 data are used for evaluation of PE. All these 3 conflicting objectives are considered in a 3-objective optimization process which consequently leads to a complete Pareto set of solutions of GMDH-type neural networks models. It is shown that AIC formulation either for training error or for prediction error will coincide to only two non-dominated optimum points obtained from such Pareto approach of GMDH-type neural networks. The results of this work demonstrate that some useful and informative design concepts regarding the optimal choices of models can be unveiled by the combination of GMDH-type neural networks and multi-objective EAs.

2 Modeling Using GMDH Neural Networks

By means of GMDH algorithm a model can be represented as set of neurons in which different pairs of them in each layer are connected through a quadratic polynomial and thus produce new neurons in the next layer. Such representation can be used in modelling to map inputs to outputs. The formal definition of the identification problem is to find a function $\hat{f}$ so that can be approximately used instead of actual one, f in order to predict output $\hat{y}$ for a given input vector $X = (x_1, x_2, x_3, ..., x_n)$ as close as possible to its actual output y. Therefore, given *M* observation of multi-input-single-output data pairs so that

$$y_i = f(x_{i1}, x_{i2}, x_{i3}, ..., x_{in}) \quad (i=1, 2 \ldots M), \tag{1}$$

it is now possible to train a GMDH-type neural network to predict the output values $\hat{y}_i$ for any given input vector $X = (x_{i1}, x_{i2}, x_{i3}, ..., x_{in})$, that is

$$\hat{y}_i = \hat{f}(x_{i1}, x_{i2}, x_{i3}, ..., x_{in}) \quad (i=1, 2 \ldots M) \quad . \tag{2}$$

The problem is now to determine a GMDH-type neural network so that the square of difference between the actual output and the predicted one is minimized, that is

$$\sum_{i=1}^{M} [\hat{f}(x_{i1}, x_{i2}, x_{i3}, ..., x_{in}) - y_i]^2 \to \min \quad . \tag{3}$$

General connection between inputs and output variables can be expressed by a complicated discrete form of the Volterra functional series in the form of

$$y_o = a_0 + \sum_{i=1}^{n} a_i x_i + \sum_{i=1}^{n}\sum_{j=1}^{n} a_{ij} x_i x_j + \sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{k=1}^{n} a_{ijk} x_i x_j x_k + \ldots \quad , \tag{4}$$

where is known as the Kolmogorov-Gabor polynomial [2]. This full form of mathematical description can be represented by a system of partial quadratic polynomials consisting of only two variables (neurons) in the form of

$$\hat{y} = G(x_i, x_j) = a_0 + a_1 x_i + a_2 x_j + a_3 x_i x_j + a_4 x_i^2 + a_5 x_j^2 \quad . \tag{5}$$

In this way, such partial quadratic description is recursively used in a network of connected neurons to build the general mathematical relation of inputs and output variables given in equation (4). The coefficients a_i in equation (5) are calculated using regression techniques [1-3] so that the difference between actual output, y, and the calculated one, $\hat{y}$ for each pair of (x_i, x_j) as input variables is minimized. Indeed, it can be seen that a tree of polynomials is constructed using the quadratic form given in equation (5) whose coefficients are obtained in a least-squares sense. In this way, the coefficients of each quadratic function G are obtained to optimally fit the output in the whole set of input-output data pair, that is

$$E = \frac{\sum_{i=1}^{M} (y_i - y_o)^2}{M} \rightarrow \min \quad . \tag{6}$$

In the basic form of the GMDH algorithm, all the possibilities of two independent variables out of total n input variables are taken in order to construct the regression polynomial in the form of equation (5) that best fits the dependent observations (y_i, $i=1, 2, \ldots, M$) in a least-squares sense. Consequently, $\binom{n}{2} = \frac{n(n-1)}{2}$ neurons will be built up in the first hidden layer of the feed forward network from the observations $\{(y_i, x_{ip}, x_{iq});\ (i=1, 2, \ldots, M)\}$ for different $p,q \in \{1,2,\ldots,n\}$. In other words, it is now possible to construct M data triples $\{(y_i, x_{ip}, x_{iq});\ (i=1, 2\ldots M)\}$ from observation using such $p,q \in \{1,2,\ldots,n\}$ in the form

$$\left[\begin{array}{cc|c} x_{1p} & x_{1q} & y_1 \\ x_{2p} & x_{2q} & y_2 \\ \hline x_{Mp} & x_{Mq} & y_M \end{array}\right].$$

Using the quadratic sub-expression in the form of equation (5) for each row of M data triples, the following matrix equation can be readily obtained as

$$A\mathbf{a} = Y \,, \tag{7}$$

where $\mathbf{a}$ is the vector of unknown coefficients of the quadratic polynomial in equation (5)

$$\mathbf{a} = \{a_0, a_1, \ldots, a_5\} \,, \tag{8}$$

and

$$Y = \{y_1, y_2, y_3, \ldots, y_M\}^T \,, \tag{9}$$

is the vector of output's value from observation. It can be readily seen that

$$A = \left[\begin{array}{cccccc} 1 & x_{1p} & x_{1q} & x_{1p}x_{1q} & x_{1p}^2 & x_{1q}^2 \\ 1 & x_{2p} & x_{2q} & x_{2p}x_{2q} & x_{2p}^2 & x_{2q}^2 \\ \hline 1 & x_{Mp} & x_{Mq} & x_{Mp}x_{Mq} & x_{Mp}^2 & x_{Mq}^2 \end{array}\right] \quad . \tag{10}$$

The least-squares technique from multiple-regression analysis leads to the solution of the normal equations in the form of

$$\mathbf{a} = (A^T A)^{-1} A^T Y \,, \tag{11}$$

which determines the vector of the best coefficients of the quadratic equation (5) for the whole set of M data triples. It should be noted that this procedure is repeated for each neuron of the next hidden layer according to the connectivity topology of the network. However, such a solution directly from normal equations is rather susceptible to round off errors and, more importantly, to the singularity of these equations.

2.1. Application of SVD to the design of GMDH-Type neural networks

The SVD of a matrix, $A \in \Re^{M \times 6}$ is a factorisation of the matrix into the product of three matrices, column-orthogonal matrix $U \in \Re^{M \times 6}$, diagonal matrix $W \in \Re^{6 \times 6}$ with non-negative elements (singular values), and orthogonal matrix $V \in \Re^{6 \times 6}$ such that

$$A = U\, W\, V^T \quad . \tag{12}$$

The problem of optimal selection of vector of the coefficients in equations (7) (11) is firstly reduced to finding the modified inversion of diagonal matrix W [11] in which the reciprocals of zero or near zero singulars (according to a threshold) are set to zero. Then, such optimal $\mathbf{a}$ is calculated using the following relation

$$\mathbf{a} = V\,[diag(1/w_j)]\,U^T\; Y \quad . \tag{13}$$

Such parametric identification problem is part of the general problem of modelling when structure identification is considered together with the parametric identification problem simultaneously. In this work, an encoding scheme developed by authors [10] is used in an evolutionary approach for simultaneous determination of structure and parametric identification of GMDH neural networks.

2.2 Application of GA in the topology design of GMDH-Type neural networks

GAs as stochastic methods are commonly used in the training of neural networks in terms of associated weights or coefficients and have successfully performed better than traditional gradient-based techniques [5]. The literature shows that a wide range of evolutionary design approaches either for architectures or for connection weights separately, in addition to efforts for them simultaneously [6]. In the most GMDH-type neural network, neurons in each layer are only connected to neurons in its adjacent layer as it was the case in Methods I and II previously reported in [5]. Taking this advantage, it is possible to present a simple encoding scheme for the genotype of each individual in the population [12]. The encoding scheme in generalized GMDH (GS-GMDH) neural networks must demonstrate the ability of representing different length and size of such neural networks and is now presented in summary.

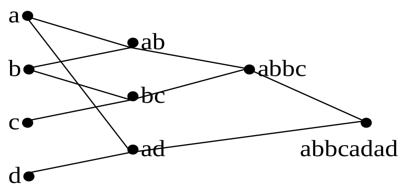

Fig. 1. A GS-GMDH network structure of a chromosome

In figure 1, neuron <u>*ad*</u> in the first hidden layer is connected to the output layer by directly going through the second hidden layer. Therefore, it is now very easy to notice that the name of output neuron (network's output) includes <u>*ad*</u> twice as <u>*abbcadad*</u>. In other words, a virtual neuron named <u>*adad*</u> has been constructed in the second hidden layer and used with <u>*abbc*</u> in the same layer to make the output neuron <u>*abbcadad*</u> as shown in the figure 1. It should be noted that such repetition occurs whenever a neuron passes some adjacent hidden layers and connects to another neuron in the next 2nd, or 3rd, or 4th, or … following hidden layer. In this encoding scheme, the number of repetition of that neuron depends on the number of passed hidden layers, ñ, and is calculated as $2^{\tilde{n}}$. It is easy to realize that a chromosome such as <u>*abab bcbc*</u>, unlike chromosome <u>*abab acbc*</u> for example, is not a valid one in GS-GMDH networks and has to be simply re-written as <u>*abbc*</u>.

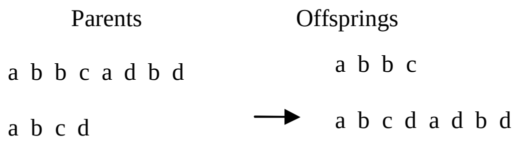

Fig. 2. Crossover operation for two individuals in GS-GMDH networks

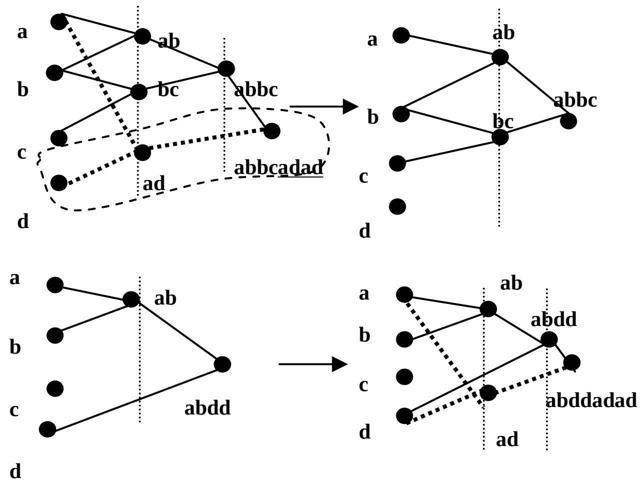

Fig. 3. Crossover operation on two GS-GMDH networks

2.3 Genetic Operators for GS-GMDH Network Reproduction

The genetic operators of crossover and mutation can now be implemented to produce two offsprings from two parents. The natural roulette wheel selection method is used for choosing two parents producing two offsprings. The crossover operator for two selected individuals is simply accomplished by exchanging the tails of two chromosomes from a randomly chosen point as shown in figure 2. It should be noted, however, such a point could only be chosen randomly from the set $\{2^1, 2^2, ..., 2^{n_l+1}\}$ where n_l is the number of hidden layers of the chromosome with the smaller length. It is very evident from figures 2 and 3 that the crossover operation can certainly exchange the building blocks information of such GS-GMDH neural networks. In addition, such crossover operation can also produce different length of chromosomes which in turn leads to different size of GS-GMDH network structures. Similarly, the mutation operation can contribute effectively to the diversity of the population. This operation is simply accomplished by changing one or more symbolic digits as genes in a chromosome to another possible symbol, for example, *abbcadad* to *abbccdad*. It should be noted that such evolutionary operations are acceptable only if a valid chromosome is produced. Otherwise, these operations are simply repeated until a valid chromosome is constructed.

3 Multi-objective optimization

Multi-objective optimization which is also called multicriteria optimization or vector optimization has been defined as finding a vector of decision variables satisfying constraints to give optimal values to all objective functions [8-10]. In general, it can be mathematically defined as:

find the vector $X^* = \left[x_1^*, x_2^*, ..., x_n^*\right]^{\mathrm{T}}$ to optimize

$$F(X) = \left[f_1(X), f_2(X), ..., f_k(X)\right]^{\mathrm{T}}, \tag{14}$$

subject to m inequality constraints

$$g_i(X) \le 0 \quad , \quad i = 1 \text{ to } m, \tag{15}$$

and p equality constraints

$$h_j(X) = 0 \quad , \quad j = 1 \text{ to } p, \tag{16}$$

where $X^* \in \Re^n$ is the vector of decision or design variables, and $F(X) \in \Re^k$ is the vector of objective functions. Without loss of generality, it is assumed that all objective functions are to be minimized. Such multi-objective minimization based on the Pareto approach can be conducted using some definitions:

Definition of Pareto dominance

A vector $U = [u_1, u_2, ..., u_k] \in \Re^k$ dominates to vector $V = [v_1, v_2, ..., v_k] \in \Re^k$ (denoted by $U \prec V$) if and only if $\forall i \in \{1,2,...,k\}$, $u_i \le v_i \wedge \exists j \in \{1,2,...,k\} : u_j < v_j$. It means that there is at least one u_j which is smaller than v_j whilst the rest u's are either smaller or equal to corresponding v's.

Definition of Pareto optimality

A point $X^* \in \Omega$ (Ω is a feasible region in $\Re^n$ satisfying equations (15) and (16)) is said to be Pareto optimal (minimal) with respect to all $X \in \Omega$ if and only if $F(X^*) \prec F(X)$. Alternatively, it can be readily restated as $\forall i \in \{1,2,...,k\}$, $\forall X \in \Omega - \{X^*\}$ $f_i(X^*) \le f_i(X) \wedge \exists j \in \{1,2,...,k\}$: $f_j(X^*) < f_j(X)$. It means that the solution X^* is said to be Pareto optimal (minimal) if no other solution can be found to dominate X^* using the definition of Pareto dominance.

Definition of Pareto front

For a given MOP, the Pareto front $\mathcal{PT}^*$ is a set of vectors of objective functions which are obtained using the vectors of decision variables in the Pareto set $\mathcal{P}^*$, that is, $\mathcal{PT}^* = \{F(X) = (f_1(X), f_2(X), ..., f_k(X)) : X \in \mathcal{P}^*\}$. Therefore, the Pareto front $\mathcal{PT}^*$ is a set of the vectors of objective functions mapped from $\mathcal{P}^*$.

Definition of Pareto Set

For a given MOP, a Pareto set $\mathcal{P}^*$ is a set in the decision variable space consisting of all the Pareto optimal vectors, $\mathcal{P}^* = \{X \in \Omega \mid \nexists X' \in \Omega : F(X') \prec F(X)\}$. In other words, there is no other X' in Ω that dominates any $X \in \mathcal{P}^*$ in terms of their objectiove functions.

4 Modelling of a Nonlinear System using Multi-objective GMDH neural networks

The input–output data used in such modelling involve 100 data pairs randomely generated from a nonlinear system with three inputs x_1, x_2, x_3, and a single output y given by

$$y = (1 + x_1^{0.5} + x_2^{-1.} + x_3^{-1.5}), \qquad 1 \le x_1, x_2, x_3 \le 5 \quad . \tag{17}$$

There are 50 pattern numbers which have been randomly selected from those data pairs to train such GMDH–type neural networks. However, a testing set which consists of 50 unforeseen input-output data samples during the training process, is merely used for testing to show the prediction ability of such evolved GMDH-type neural network models during the training process.

The GMDH-type neural networks are now used for such input-output data to find the polynomial model of y in such nonlinear system process with respect to their input parameters. In order to design GMDH-type neural network described in previous section from a multi-objective optimum point of view, a population of 60 individuals with a crossover probability of 0.95 and mutation probability of 0.1 has been used in 250 generation that no further improvement has been achieved for such population size. A multi-objective optimization of GMDH-type neural networks including all three objectives can offer more choices for a designer. Figure 4 depicts the non-dominated points of 3-objective optimization process in the plane of (TE-PE). It should be noted that there is a single set of non-dominated points as a result of 3-objective Pareto optimization of TE, PE and N that are shown in that plane. Therefore, there are some points in the plane that may dominate others in the case of 3-objective optimization. However, these points are all non-dominated when considering all three objectives simultaneously. By careful investigation of the results of 3-objective optimization in that plane, the Pareto front of the corresponding 2-objective optimization (TE-PE) can now be observed. In this figure, points A and B stand for the best (TE) and the best (PE), respectively. The corresponding values of errors, number of neurons, and the structure of these extreme optimum design points are given in table 1.

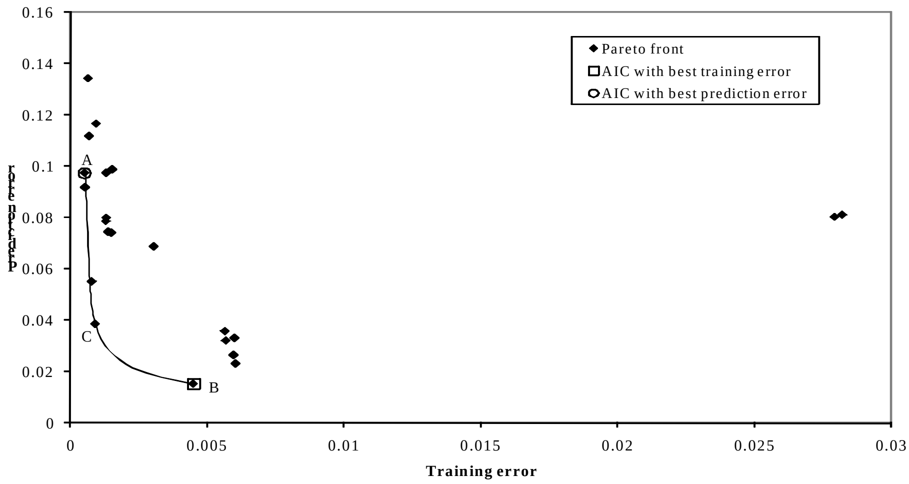

Fig. 4. Prediction error variation with training error in 3-objective optimization.

Clearly, there is an important optimal design fact between these two objective functions which has been discovered by the Pareto optimum design of GMDH-type neural networks. Such important design fact could not have been found without the multi-objective Pareto optimization of those GMDH-type neural networks. From figure 4 points C is the point which demonstrates such important optimal design fact. Point C in the Pareto front of optimum design of TE and PE, exhibits small increase in the value of TE in comparison with that of point A whilst its PE shows significant improvement (about 150 times better prediction error). Therefore, point C could be a trade-off optimum choice when considering the minimum values of both PE and TE simultaneously. The structure and network configuration corresponding to point C is shown in figure 5.

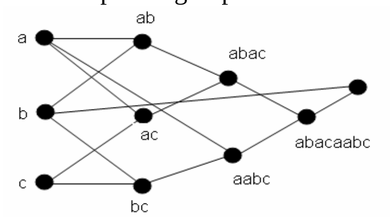

Fig. 6. The network's structure of point C in which a, b, c and d stand for x_1, x_2, x_3 respectively.

In order to compare these results, AIC [13][7] has been used both for training and testing data in two different single objective optimization processes. AIC is defined by

$$AIC = n \log_e (E) + 2 (N+1) + C \qquad (18)$$

where E, the mean square of error, is computed using equation 6 , N is the number of neurons, n is number of training/testing error, and C is a constant.

Tab.1. Objective functions and structure of networks of different points shown on figure 4.

	Network's chromosome	**No. of Neurons**	**Training error**	**Prediction error**
Point A	bbbbbbbbaabcabab	5	0.000545619	0.097273886
Point B	bbbbabacbbabaaaa	7	0.004518445	0.015062286
Point C	bbbbbbbbaabcabac	7	0.000938418	0.038422696

Therefore, two optimum points have been found using AIC and are shown in figure 4. Clearly, these two points coincide with the points A and B correspondingly. It is then evident that the Pareto optimum design of GMDH-type neural networks presented in this paper are inclusive of those obtained by AIC and also presents more effective way of choosing trade-off optimum models with respect to conflicting objective functions. It should be noted that point C could be achieved by a proper weighting coefficients (which is not know *a priori*) of prediction and training errors using AIC in only convex programming problems.

5 Conclusion

Genetic algorithms have been successfully used for multi-objective Pareto based optimization of generalized GMDH-type neural networks used for modelling and prediction of a nonlinear system. Such multi-objective optimization led to the discovering of useful optimal design principles in the space of objective functions. In this work, the important conflicting objective functions of GMDH-type neural networks have been selected as Training Error (TE), Prediction Error (PE) and Number of Neurons (N) of such neural networks. In addition to discovering the trade-off optimum points, it has been shown that the Pareto front obtained by the approach of this paper involves those that can be found by Akaike's Information Criterion which thus exhibits the effectiveness of the Pareto optimum design of GMDH-type neural networks presented in this paper.

References

[1] Ivakhnenko, A. G.,: Polynomial Theory of Complex Systems. *IEEE Trans. Syst. Man & Cybern*, SMC-1, 364-378, 1971.

[2] Farlow, S. J.,: Self-organizing Method in Modeling: GMDH type algorithm. Marcel Dekker Inc., 1984.

[3] Iba, H., deGaris, H. and Sato, T.,: A numerical Approach to Genetic Programming for System Identification. *Evolutionary Computation* 3(4):417-452, 1996.

[4] Nariman-Zadeh, N., Darvizeh, A., Felezi, M. E. and Gharababaei, H.,: Polynomial modelling of explosive compaction process of metallic powders using GMDH-type neural networks and singular value decomposition. *Modelling and Simulation in Materials Science and Engineering*, Vol. 10, no. 6, pp. 727-744(18), 2002.

[5] Nariman-Zadeh, N., Darvizeh, A. and Ahmad-Zadeh, G. R.,: Hybrid Genetic Design of GMDH-Type Neural Networks Using Singular Value Decomposition for Modelling and Prediction of the Explosive Cutting Process. *Proceedings of the I MECH E Part B Journal of Engineering Manufacture*, Volume: 217, Page: 779 – 790, 2003.

[6] Yao, X.,: Evolving Artificial Neural Networks. *Proceedings of IEEE* 87(9):1423-1447, 1999.

[7] Kondo, T. and Ueno, J.,: Revised GMDH-type Neural Network Algorithm with a Feedback Loop Identifying Sigmoid Function Neural Network. *Int. J. of Innovative Computing , Information and Control*, Vol. 2 No.5, 2006.

[8] Deb, K., Agrawal, S., Pratap, A. and Meyarivan, T.,: A fast and elitist multi-objective genetic algorithm: NSGA-II. *IEEE Trans. On Evolutionary Computation* 6(2):182-197, 2002.

[9] Coello Coello, C. A.,: A comprehensive survey of evolutionary based multiobjective optimization techniques. *Knowledge and Information Systems: An Int. Journal*, (3), pp 269-308, 1999.

[10] Nariman-zadeh, N., Atashkari, K., Jamali, A., Pilechi, A. and Yao, X.,: Inverse Modelling of Multi-objective Thermodynamically Optimized Turbo Engines using GMDH-type Neural Networks and Evolutionary Algorithms. *Engineering Optimization*, Vol. 37, No. 5, 437-462, 2005.

[11] Press, W. H., Teukolsky, S. A., Vetterling, W. T. and Flannery B. P.,: Numerical Recipes in FORTRAN: The Art of Scientific Computing, 2nd Edition, *Cambridge University Press*, 1992.

[12] Nariman-zadeh, N., Darvizeh, A., Jamali, A. and Moeini,: A. Evolutionary Design of Generalized Polynomial Neural Networks for Modelling and Prediction of Explosive Forming Process. *Journal of Material Processing and Technology*, Vol 164-165, pp 1561-1571, Elsevier, 2005.

[13] Akaike, H.,: A new look at the statistical model identification, *IEEE Trans. Automatic Control*, vol.AC-19, no.6, pp.716-723, 1974.