A Polynomial Model for Concrete Compressive Strength Prediction using GMDH-type Neural Networks and Genetic Algorithm

N. Hamid-zadeh[1], A. Jamali[2], N. Nariman-zadeh[2], H. Akbarzadeh[1]
[1]Department of Civil Engineering
[2]Department of mechanical Engineering
Faculty, The University of Guilan
P.O. Box 3756, Rasht, IRAN

Abstract: Compressive strength of concrete is experimentally determined at different ages such as 7, 28 and 42-day old as an witness specimen for final judgment. In this paper, compressive strength of 42-day is modeled and predicted using GMDH-type neural networks base on some experimental data. The aim of such modelling is to show how compressive strength of 42-day change with the variation of Compressive strength of 7 and 28 days old. In this way, a new encoding scheme is presented to genetically design generalized GMDH-type neural networks in which the connectivity configuration in such networks is not limited to adjacent layers. Such generalization of network's topology provides optimal networks in terms of hidden layers and/or number of neurons so that a simple polynomial expression to model and predict the compressive strength of 42-day old concrete consequently.

Key-Words:- Concrete, Compressive strength, Group Method of Data Handling (GMDH), GAs.

1 Introduction

Concrete is such a construction material that is widely used in the world. The advantages of concrete are low cost, availability of constituents, workability, durability and convenient compressive strength that make it popular near engineers and builders. However, these advantages seriously depend on the correct mix and placing and curing [1,2]. Quality control of concrete is made at three stages, control of properties of the constituents (water, cement and aggregates), tests on fresh concrete and test on hardened concrete(compressive, tensile and bending strength tests).

Compressive strength of concrete specimens is determined at different ages in different ways. Usually this test is made on samples adopted from fresh concrete when they hardened and sometimes it is made on in field as a nondestructive test for a final judgment. In most structural codes, specially Iranian concrete code, ABA, the design of reinforced concrete structures is based on compressive strength of standard cylindrical specimens at 28 days old. At executive projects, compressive strength of specimens is measured at specific times such as 7, 28 days old. If it is necessary, a witness specimen is also tested at 42 or 60 or 90 days old. There are some relationships in previously published studies that can predict the 28 days strength easily from 7 days values [1,3]. This enable us to know very soon about quality of concrete and its probable weakness and decide to continue the construction or manage the destruction program.

Current investigations have showed that existing relationships are not suitable for use in Iran and might not have high accuracy because of the differences in chemical properties of cements and other factors [4]. Also most of these relations correlate compressive strength at one age to another, so we are looking for a relation that can predict compressive strength at 42-day old with high accuracy using values of 7 and 28 days. Therefore, a comprehensive database of concrete compressive strength tests case histories at some projects in Iran since 1998 to 2006 have been investigated and analyzed.

System identification techniques are applied in many fields in order to model and predict the behaviors of unknown and/or very complex systems based on given input-output data [5]. Theoretically, in order to model a system, it is required to understand the explicit mathematical input-output relationship precisely. Such explicit mathematical modelling is, however, very difficult and is not readily tractable in poorly understood systems. Alternatively, soft-computing methods [6], which concern computation in imprecise environment, have gained significant attention. The main components of soft computing, namely, fuzzy-logic, neural network, and genetic algorithm have shown great ability in solving complex non-linear system identification and control problems. Several research efforts have been expended to use evolutionary methods as effective tools for system identification [6-7]. Among these methodologies, Group Method of Data Handling (GMDH) algorithm is 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 firstly developed by Ivakhnenko [8] 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. In other words, 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 [9] whose coefficients are obtained using regression technique. In fact, real GMDH algorithm in which model coefficients are estimated by means of the least squares method has been classified into complete induction and incomplete induction, which represent the combinatorial (COMBI) and multilayered iterative algorithms (MIA), respectively [9]. In recent years, however, the use of such self-organizing network leads to successful application of the GMDH-type algorithm in a broad range area in engineering, science, and economics [9-10].

In this paper, compressive strength of 42-day concrete is modeled and predicted using GMDH-type neural networks based on some experimental data. The aim of such modelling is to show how compressive strength of 42-day change with the variation of compressive strength of 7 days and 28 days old. In this way, genetic algorithms are deployed in a new approach to design the whole architecture of the GMDH-type neural networks, i.e., the number of neurons in each hidden layer and their connectivity configuration, in combination with using regression method to find optimal set of appropriate coefficients of quadratic expressions for modelling and prediction of compressive strength of 42-day old. The connectivity configuration is not limited to the adjacent layers, as have already proposed in [10], so that general structure GMDH-type neural networks, in which neurons in any layers can be connected to each other, are evolved. Furthermore, a new yet simple encoding scheme has been introduced to present the genotype of general structure GMDH-type neural networks so that the length of chromosome is kept minimal and thus leads to simpler evolutionary operations including crossover and mutation. Such encoding scheme must allow for producing and/or switching both general and simplified networks using the genetic operators during evolutionary process.

2 Modelling Using GMDH-Type 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 $X = (x_1, x_2, x_3, \ldots, x_n)$ as close as possible to its actual output y. Therefore, given M observation of multi-input-single-output data pairs so that

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

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

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

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

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

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

$$y = a_0 + \sum_{i=1}^{n} a_i x_i + \sum_{i=1}^{n}\sum_{j=1}^{n} a_{ij} x_i x_j + \sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{k=1}^{n} a_{ijk} x_i x_j x_k + \ldots \qquad (4)$$

which is known as the Kolmogorov-Gabor polynomial [9-10]. 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

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

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 coefficient a_i in equation (5) are calculated using regression techniques [9-10] 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_i are obtained to optimally fit the output in the whole set of input-output data pair, that is

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

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 feedforward network from the observations $\{(y_i, x_{ip}, x_{iq});\ (i=1, 2, \ldots, M)\}$ for different $p, q \in \{1,2,...,n\}$. 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 $p, q \in \{1,2,...,n\}$ 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)

$$\mathbf{a} = \{a_0, a_1, a_2, a_3, a_4, a_5\} \tag{8}$$

and

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

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

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

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.

3 Application of GAs in the Topology Design of GMDH

Evolutionary methods such as genetic algorithms have been widely used in different aspects of design in neural networks because of their unique capabilities of finding a global optimum in highly multi-modal and/or non-differentiable search space [7, 11]. Such stochastic methods are commonly used in the training of neural networks in terms of associated weights or coefficients which have successfully performed better than traditional gradient-based techniques [10]. 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 [11]. In the most GMDH-type neural network, neurons in each layer are only connected to neuron in its adjacent layer as it was the case in Methods I and II previously reported in [10]. Taking this advantage, it was possible to present a simple encoding scheme for the genotype of each individual in the population as already proposed by authors [10].

However, in order to make it more general, it is necessary to remove such restriction. In such representation, neurons in different layers including the input layer can be connected to others far away, not only in the very adjacent layers. In addition, the genotype of these networks must be encoded in such a way to include the restricted networks, namely, conventional structure GMDH-type neural networks [10]. The encoding schemes in generalized GMDH neural networks must demonstrate the ability of representing different length and size of such neural networks. Moreover, the ability of changing building blocks of information using crossover and mutation operators must be preserved in such encoding representation. In the next sections, the encoding scheme of the newly developed generalized GMDH neural networks is discussed.

3.1 The Genome Representation of Generalized GMDH Neural Networks

In the generalized GMDH neural networks, neurons connections can occur between different layers which are not necessarily very adjacent ones, unlike the conventional structure GMDH neural networks in which such connections only occur between adjacent layers. For example, a network structure which depicted in figure (1) shows such connection of neuron ***ad*** directly to the output layer. Consequently, this generalisation of network's structure can evidently extend the performance of generalized GMDH neural networks in modelling of real-world complex processes. Such generalization is accomplished by repeating the name of the neuron which directly passing the next layers. In figure (1), neuron ***ad*** 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 ***ad*** twice as ***abbcadad***. In other words, a virtual neuron named ***adad*** has been constructed in the second hidden layer and used with ***abbc*** in the same layer to make the output neuron ***abbcadad*** 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 ***abab bcbc***, unlike chromosome ***abab acbc*** for example, is not a valid one in GS-GMDH networks and has to be simply re-written as ***abbc*** .

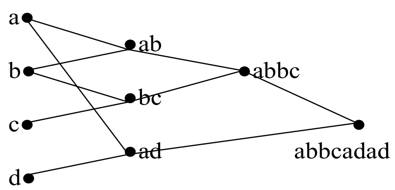

Figure 1: A Generalized GMDH network structure of a chromosome

3.2 Genetic Operators for 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 generalized GMDH neural networks so that the two type of generalized GMDH and conventional GMDH-type neural networks can be converted to each other, as can be seen from figure (3). In addition, such crossover operation can also produce different length of chromosomes which in turn leads to different size of either generalized GMDH or conventional 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 symbols, for example, *abbcadad* to *abbccdad*. It is very evident that mutation operation can also convert a generalized GMDH network to a conventional GMDH network or vice versa. It should be noted that such evolutionary operations are acceptable provided a valid chromosome is produced. Otherwise, these operations are simply repeated until a valid chromosome is constructed

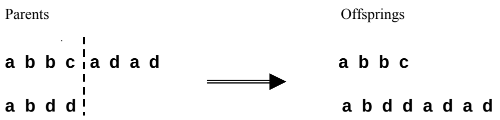

Figure 2: Crossover operation for two individuals in generalized GMDH networks

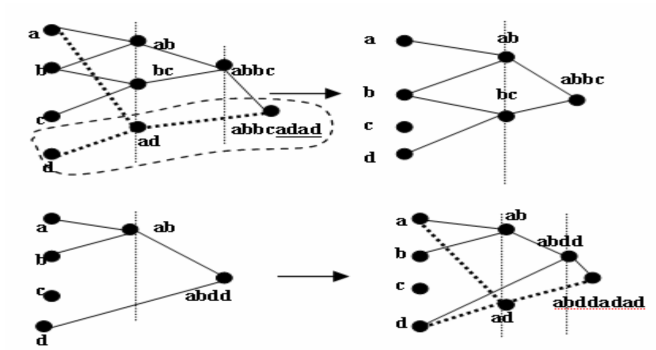

Figure 3: Crossover operation on two generalized GMDH networks

The incorporation of genetic algorithm into the design of such GMDH-type neural networks starts by representing each network as a string of concatenated sub-strings of alphabetical digits. The fitness, (Φ), of each entire string of symbolic digits which represents a GMDH-type neural network to model compressive strength of 42-day old concrete is evaluated in the form

$$\Phi = 1/E \quad , \qquad (12)$$

where *E*, is the mean square of error given by equation (6), is minimized through the evolutionary process by maximizing the fitness Φ. The evolutionary process starts by randomly generating an initial population of symbolic strings each as a candidate solution. Then, using the aforementioned genetic operations of roulette wheel selection, crossover, and mutation, the entire populations of symbolic strings to improve gradually. In this way, GMDH-type neural network models of compressive strength of concrete at 42-day old with progressively increasing fitness, Φ, are produced until no further significant improvement is achievable.

4 Results

There have been a total number of 77 input-output experimental data considering compressive strength of 7 days and compressive strength of 28 days as inputs and compressive strength of 42-day as output. The GMDH-type neural networks are now used for such input-output data to find the polynomial model of compressive strength of 42-day in respect to their effective input parameters. In order to genetically design such GMDH-type neural network described in previous section a population of 50 individuals with a crossover probability of 0.85 and mutation probability of 0.1 has been used in 200 generation which no further improvement has been achieved for such population size. The structure of the evolved 2-hidden layer GMDH-type neural network is shown in figure (4). The very good behaviour of such GMDH-type neural network models are also depicted in figures (5). The corresponding polynomial representation of such model is:

$$Y1 = 25 - 0.27(f7) + 1.13(f28) + 0.00058(f7)^2 - 0.0014(f28)^2 + 0.0024(f7)(f28) \quad (13a)$$

$$Y2 = -7 - 0.57(f7) + 1.38(Y1) - 0.0081(f7)^2 - 0.004(Y1)^2 + 0.011(f7)(Y1) \quad (13b)$$

$$Y3 = -13.3 - 0.21(Y1) + 1.6(f28) + 0.09(Y1)^2 + 0.12(f28)^2 - 0.21(Y1)(f28) \quad (13c)$$

$$f42 = 13.8 + 5.04\,(Y2) - 4.2(Y3) - 0.07\,(Y2)^2 - 0.05(Y3)^2 + 0.12(Y2)(Y3) \quad (13d)$$

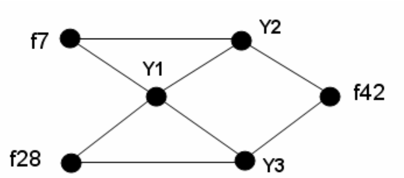

Figure 4: Evolved structure of generalized GMDH neural network

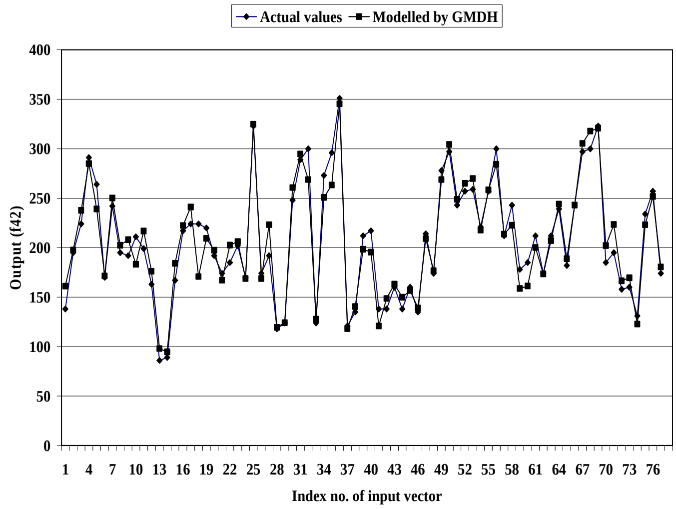

figure 5: Comparison of Actual values and 2-hidden Evolved GMDH Neural Model

However, in order to demonstrate the prediction ability of evolved GMDH-type neural networks, the data has been divided into two different sets, namely, training and testing sets. The training set, which consists of 60 out of 77 inputs-output data pairs, is used for training the neural network models the evolutionary method of this paper. The testing set, which consists of 17 unforeseen inputs-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 structure of the evolved 2-hidden layer GMDH-type neural network is that same as figure (4). The very good behaviour of such GMDH-type neural network models are also depicted in figures (6).

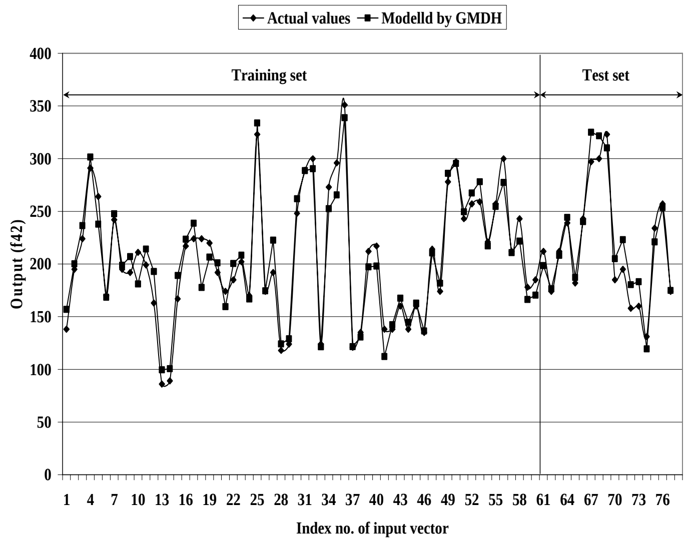

figure 6: Comparison of Actual values and 2-hidden Evolved GMDH Neural (Model1ing & Prediction)

5 Conclusion

Evolutionary methods for designing generalized GMDH-type networks have been proposed and successfully used for the modelling and prediction of compressive strength of 42-day old concrete. In this way, a new encoding scheme has been presented to genetically design generalized GMDH-type neural networks in which the connectivity configuration in such networks is not limited to adjacent layers, unlike the conventional GMDH-type neural networks. Such generalization of network's topology provides optimal networks in terms of hidden layers and/or number of neurons and their connectivity configuration so that a polynomial expression for modelling and prediction of the 42-day old concrete strength can be achieved cosequently

References:

[1] Neville, A. M., "Properties of concrete", 3rd edition, Pitman publishing limited, London, 1981.

[2] Tasdemir, C., et al., "The influence of aggregate type on the mechanical properties of concrete", Proceeding of the 12th European realy mixed concrete congress, Vol. 1, Lisbon-Portugal, June, pp. 564-573, 1998.

[3] Merrit, F.S., and Loftin, M.K. and Ricketts, J.T., "Standard handbook for civil engineers", 4th edition, McGraw-hill, companies, Inc, 1996.

[4] Akbarzadeh, H., and Hamidzadeh, N., and Ranjbar, N.M., "study on the relations between compressive strength of concrete at different ages using 165 case studies", Proceeding of the 7th International Congress on Civil engineering, Tehran, Iran, 2006.

[5] Astrom, K.J. and Eykhoff, P., "System identification, a survey", *Automatica* 7-123-62, (1971).

[6] Sanchez E., Shibata T., and Zadeh L.A., "*Genetic Algorithms and Fuzzy Logic Systems*", World Scientific, (1997).

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

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

[9] Farlow, S.J., ed., " *Self-organizing Method in Modeling: GMDH type algorithm*", Marcel Dekker Inc., (1984).

[10] Nariman-Zadeh, N., Darvizeh, A., Ahmad-Zadeh, 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*, Vol. 217, pp. 779 – 790, (2003).

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