Advances in Engineering Software 45 (2012) 105–114

Contents lists available at SciVerse ScienceDirect

Advances in Engineering Software

journal homepage: www.elsevier.com/locate/advengsoft

A new predictive model for compressive strength of HPC using gene expression programming

Seyyed Mohammad Mousavi [a], Pejman Aminian [b], Amir Hossein Gandomi [c,*], Amir Hossein Alavi [d], Hamed Bolandi [e]

[a] Department of Geography and Urban Planning, Faculty of Humanities and Social Sciences, Science and Research Branch, Islamic Azad University, Tehran, Iran
[b] Department of Civil Engineering, Islamic Azad University, Shahrood Branch, Shahrood, Iran
[c] Young Researchers Club, Central Tehran Branch, Islamic Azad University, Tehran, Iran
[d] Young Researchers Club, Mashhad Branch, Islamic Azad University, Mashhad, Iran
[e] Department of Civil Engineering, Bandar Abbas Branch, Islamic Azad University, Bandar Abbas, Iran

ARTICLE INFO

Article history:
Received 7 May 2010
Received in revised form 4 June 2011
Accepted 31 August 2011
Available online 8 November 2011

Keywords:
High performance concrete
Gene expression programming
Compressive strength
Regression analysis
Sensitivity analysis
Prediction

ABSTRACT

In this study, gene expression programming (GEP) is utilized to derive a new model for the prediction of compressive strength of high performance concrete (HPC) mixes. The model is developed using a comprehensive database obtained from the literature. The validity of the proposed model is verified by applying it to estimate the compressive strength of a portion of test results that are not included in the analysis. Linear and nonlinear least squares regression analyses are performed to benchmark the GEP model. Contributions of the parameters affecting the compressive strength are evaluated through a sensitivity analysis. GEP is found to be an effective method for evaluating the compressive strength of HPC mixes. The prediction performance of the optimal GEP model is better than the regression models.

1. Introduction

High performance concrete (HPC) is a class of concretes with a better performance than conventional types. According to American Concrete Institute (ACI), HPC is a concrete that meets special combinations of performance and uniformity requirements. These characteristics cannot be achieved using conventional constituents and normal mixing, placing and curing practices [1,2]. This type of concrete can provide increased durability. Therefore, service life is effectively increased and maintenance is reduced. In order to determine the HPC mix properties, both the cost and the availability of local materials should be taken into account. Hence, more trail mix batches and testing are required in comparison with conventional concrete [2]. In addition to the basic ingredients in conventional concrete, supplementary cementitious materials and chemical admixture are used to make HPC [2–4]. Compressive strength is an important property of HPC mixes. Developing robust prediction models for this key property leads to saving costs and time and making a successful mixture. Researchers have employed statistical regression techniques for this purpose [3,4]. Empirical modeling using regression can have notable drawbacks. To perform regression analyses, the structure of the model should be pre-defined by a linear or nonlinear equation. Another remarkable constraint in application of regression analysis is the assumption of normality of residuals [2]. On the other hand, there are some concerns regarding the validity of the empirical equations presented in codes and standards for estimating the compressive strength. This is because such equations are established upon tests of concrete without supplementary cementitious [2].

Over the last decade, machine learning has attracted much attention in both academic and empirical fields for tackling civil engineering problems. The machine learning systems are powerful tools for design of computer models. Artificial neural networks (ANNs) are the most widely used machine learning methods. ANNs have widely been used to assess different characteristics of concretes [5–8]. They have been employed to predict the compressive strength and slump flow of HPC mixes [2,4,5,10,11]. Rajasekaran and Amalraj [12] and Rajasekaran et al. [13] developed prediction models for the strength of HPC mixes using sequential learning neural network (SLNN). In this connextion, Rajasekaran and Lavanya [14] employed wavelet neural network (WNN) method to assess the compressive strength of HPC. ANNs are commonly considered as black box systems since they are unable to explain

* Corresponding author.
E-mail addresses: mousavi.seyyed.mohamad@gmail.com (S.M. Mousavi), p.aminian@iau-shahrood.ac.ir (P. Aminian), a.h.gandomi@gmail.com, ag72@zips.uakron.edu (A.H. Gandomi), ah_alavi@hotmail.com (A.H. Alavi).

0965-9978/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.advengsoft.2011.09.014

Nomenclature

σ	compressive strength of high performance concrete	S	superplasticizer content
W	water content	CA	coarse aggregate content
C	cement content	FA	fine aggregate content
B	blast furnace slag content	A	age of specimens
F	fly ash content		

the underlying principles of prediction. In spite of acceptable performance of ANNs, they are considered as black box as they are not usually able to generate practical prediction equations.

Genetic programming (GP) [15] is another branch of the machine learning methods. GP automatically generates computer models based on the rules of natural genetic evolution. A major advantage of GP-based approaches is their ability to generate prediction equations without assuming prior form of the existing relationship. Classical GP and its variants have been recently utilized to derive simplified models for civil engineering problems [16–18]. Mousavi et al. [2] have recently derived prediction model for the compressive strength of HPC mixes using a combined algorithm of GP and orthogonal least squares (OLS), called GP/OLS. They formulated the compressive strength in terms of ratio of water and superplasticizer summation to binder, coarse to fine aggregate content and age of specimens. Gene expression programming (GEP) [19] is a new subset of GP. GEP evolves computer programs of different sizes and shapes encoded in linear chromosomes of fixed length. The GEP approach can reliably be utilized as an efficient alternative to the traditional GP [19,20]. There have been some scientific efforts aiming to apply GEP to civil engineering tasks [21–26].

The main purpose of this paper is to build a new GEP-based model for determining the compressive strength of HPC. GEP can substantially be useful for this purpose by directly extracting the knowledge contained in the experimental data. Several predictor variables are included in the analysis. A reliable database comprising hundreds of test results is utilized to develop the model.

2. Genetic programming

GP was introduced by Koza [15] as an extension of genetic algorithms (GAs). The main difference between GP and GAs is the representation of the solution. The GP solutions are computer programs, while GA creates a string of numbers as the solution [15]. GP works with population of individuals (computer programs) that are created randomly. Each of the programs represents a possible solution to a given problem. The classical GP technique is also referred to as standard tree-based GP [23,25]. A program evolved by classical GP is a hierarchically structured tree comprising functions and terminals. The functions and terminals are chosen at random and constructed together to form a computer model in a tree-like structure with a root point with branches extending from each function and ending in a terminal [23,25]. Fig. 1 shows an example of a simple tree representation of classical GP.

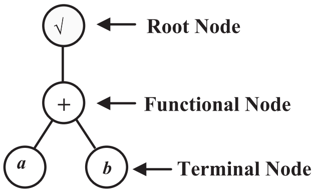

Fig. 1. The tree representation of a GP model ($\sqrt{(a+b)}$).

Once a population of models has been created at random, the GP algorithm evaluates the individuals, selects them for reproduction, generates new individuals, and finally creates the new generation in all iterations [2,15]. New individuals are created using mutation, crossover and direct reproduction. Fig. 2 shows a typical crossover operation in GP. During this operation, a point on a branch of each program is selected at random and the set of terminals and/or functions from each program are then swapped to create two new programs. During the mutation process, a function or terminal from a model is occasionally selected at random and is mutated (see Fig. 3). The best program that appeared in any generation defines the output of the GP algorithm [2,15].

GEP is a linear variant of GP. The individuals created by linear variants of GP are represented as linear strings that are decoded and expressed like nonlinear entities (trees) [20,26].

2.1. Gene expression programming

GEP is first invented by Ferreira [19]. Most of the genetic operators used in GAs can also be implemented in GEP with minor changes. GEP consists of five main components including: (1) function set, (2) terminal set, (3) fitness function, (4) control parame-

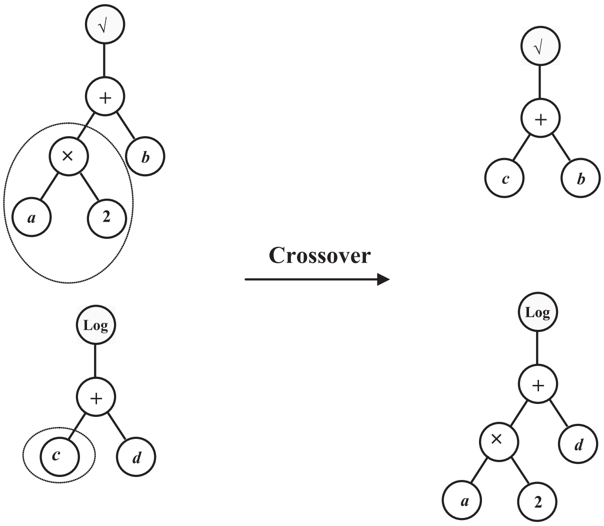

Fig. 2. Typical crossover operation in GP.

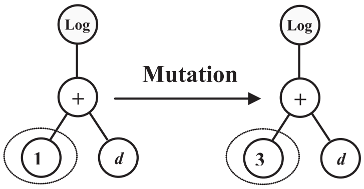

Fig. 3. Typical mutation operation in GP.

ters, and (5) termination condition. GEP uses a fixed length of character strings to represent solutions to problems, which are afterwards expressed as parse trees of different sizes and shapes [24,25]. These trees are called GEP expression trees (ETs). One advantage of the GEP technique is that the creation of genetic diversity is extremely simplified as genetic operators work at the chromosome level. Another strength of GEP refers to its unique, multi-genic nature which allows the evolution of more complex programs composed of several subprograms [24,25]. Each GEP gene contains a list of symbols with a fixed length that can be any element from a function set like {+, −, ×} and the terminal set like {*a*, *b*, *c*, 1}. The function set and terminal set must have the closure property: each function must able to take any value of data type which can be returned by a function or assumed by a terminal [24,25]. A typical GEP gene with the given function and terminal sets can be as follows:

$$\underline{+\cdot\times\cdot b\cdot a}\cdot-\cdot+\cdot\times\cdot c\cdot 1\cdot b\cdot c \qquad (1)$$

where x_1, x_2 and x_3 are variables and 3 is a constant; "·" is element separator for easy reading. The above expression is termed as Karva notation or *K*-expression [19,27]. A *K*-expression can be represented by a diagram which is an ET. For instance, the above sample gene can be presented as Fig. 4.

The conversion starts from the first position in the *K*-expression, which corresponds to the root of the ET, and reads through the string one by one. The above GEP gene can also be expressed in a mathematical form as:

$$a((c+1)-(b\times c))+b \qquad (2)$$

An ET can inversely be converted into a *K*-expression by recording the nodes from left to right in each layer of the ET, from root layer down to the deepest one to form the string. As previously mentioned, GEP genes have fixed length, which is predetermined for a given problem. Thus, what varies in GEP is not the length of genes but the size of the corresponding ETs. This means that there exist a certain number of redundant elements, which are not useful for the genome mapping. Hence, the valid length of a *K*-expression may be equal or less than the length of the GEP gene. To guarantee the validity of a randomly selected genome, GEP employs a head–tail method. Each GEP gene is composed of a head and a tail. The head may contain both function and terminal symbols, whereas the tail may contain terminal symbols only [19,24,25]. A basic representation of the GEP algorithm is presented in Fig. 5. In GEP, the individuals are selected and copied into the next generation according to the fitness by roulette wheel sampling with elitism. This guarantees the survival and cloning of the best individual to the next generation. Variation in the population is introduced by conducting single or several genetic operators on selected chromosomes, which include crossover, mutation and rotation. The rotation operator is used to rotate two subparts of element sequence in a genome with respect to a randomly chosen point. It can also drastically reshape the ETs [24,25].

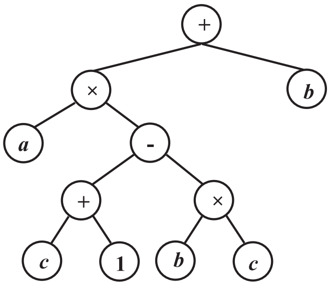

Fig. 4. Typical representation of expression trees (ETs).

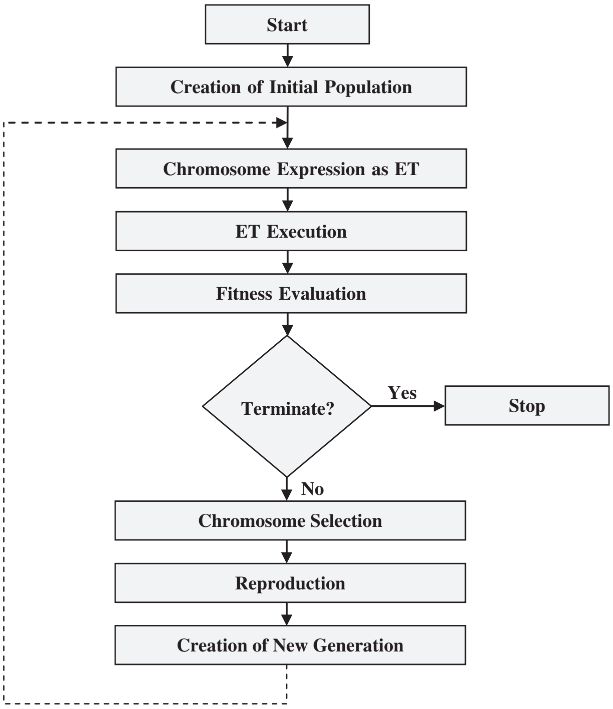

Fig. 5. A basic representation of the GEP algorithm [25].

3. Modeling of compressive strength of HPC mixes

In general, the enhanced performance characteristics of HPC are obtained by addition of various cementitious materials and chemical and mineral admixtures to the conventional concrete mix designs. According to the Abrams' well-known rule, the correlation of the strength of concrete with the water to cement ratio is negative. This rule indicates that only the quality of the cement paste controls the strength of comparable cement. Based on a variety of experimental studies, this is not quite true. For example, if two comparable concrete mixtures have the same water to cement ratio, the strength of the concrete with the higher cement content is lower [28]. Advances in recent years have been assisted by the use and understanding of chemical admixtures (e.g., super plasticizers and cement replacement materials such as fly ash and blast furnace slag). Fly ash and slag have essential effect on the workability and low slump loss rates of HPC. This is due to the mutual containment with surface lubrication and the ball-bearing effects among the fly ash and micro fine materials [2]. In many cases, there is an economic benefit of price differential between cement and supplementary cementitious material. Additionally, partial replacement of cement allows a significant reduction in the dosage of the superplasticizer [2,4]. Behavior modeling of the compressive strength of HPC is essentially more difficult than conventional concrete. In this study, the GEP approach is utilized to obtain meaningful relationships between the compressive strength (σ) of HPC mixes and the influencing variables as follows:

$$\sigma = f(W, C, B, F, S, CA, FA, A) \qquad (3)$$

where W (kg/m^3), C (kg/m^3), B (kg/m^3), F (kg/m^3), S (kg/m^3), CA (kg/m^3), FA (kg/m^3) and A (day) are the water, cement, blast furnace slag, fly ash, superplasticizer, coarse aggregate, fine aggregate and age of specimens, respectively. The above variables are chosen as the input variables on the basis of both an extensive trial study and literature review [2,4,5,9,29–31].

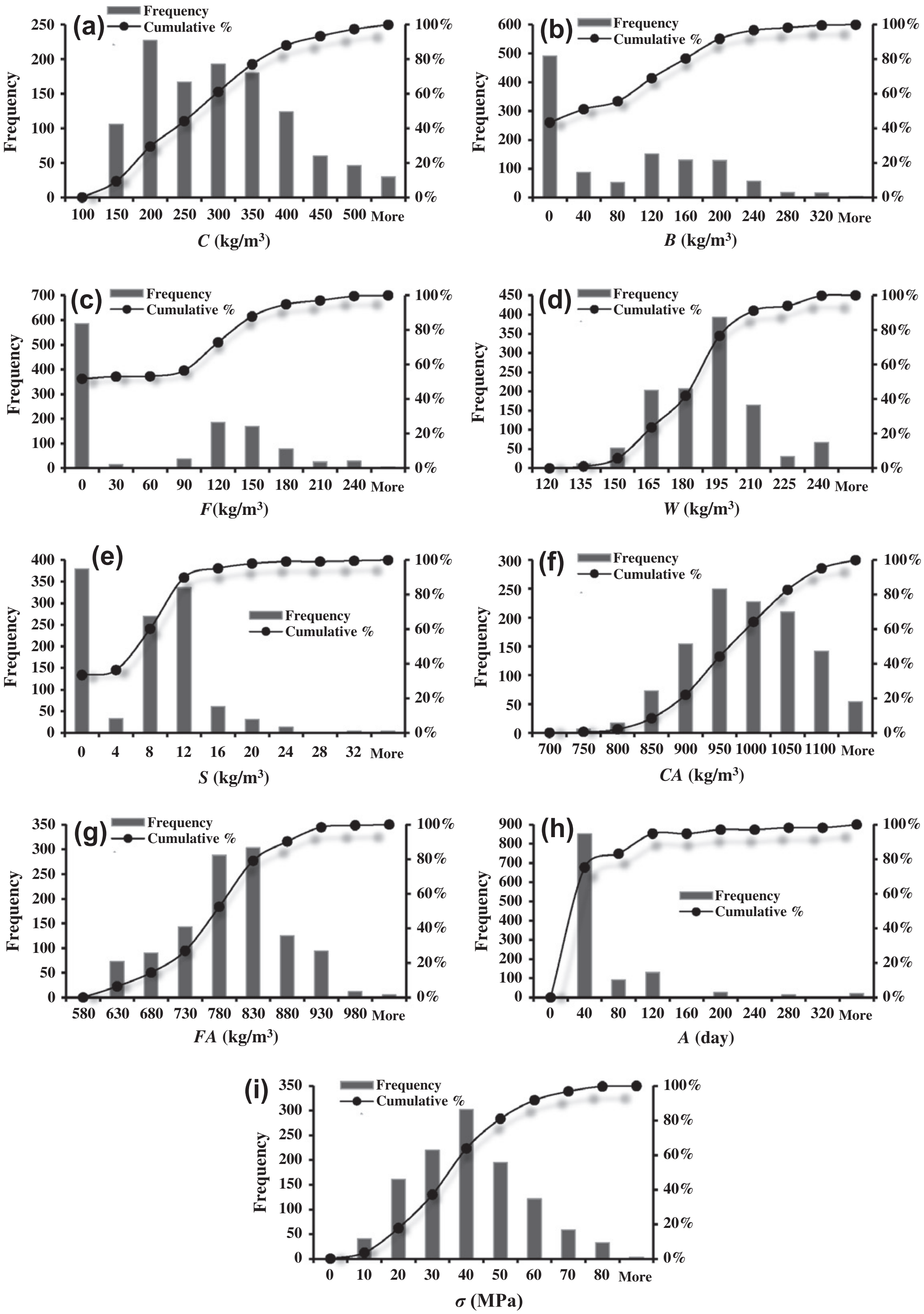

Fig. 6. Histograms of the variables.

3.1. Experimental database

It is known that the models derived using the GEP, ANNs or other similar approaches, in most cases, have a predictive capability within the data range used for their development. The amount of data used for the training process of the GEP technique bears heavily on the reliability of the final models. The only way to overcome this limitation is to employ comprehensive data sets for training their algorithms. Thus, a reliable database consisting of tests on mixtures with a wide range of aggregate gradation and properties is obtained from the literature to develop generalized models. The database contains 1133 compressive strength (σ) of HPC test results presented by Yeh [9,29]. The database includes measurements of *W*, *C*, *B*, *F*, *S*, *CA*, *FA*, *A*, and σ of HPC mixes.

Table 1
Descriptive statistics of the variables.

Parameter	*W* (kg/m^3)	*C* (kg/m^3)	*B* (kg/m^3)	*F* (kg/m^3)	*S* (kg/m^3)	*CA* (kg/m^3)	*FA* (kg/m^3)	*A* (day)	σ (MPa)
Mean	0.0785	0.1178	0.0319	0.0270	0.0027	0.4126	0.3295	44.0565	35.8380
Standard error	0.0003	0.0013	0.0011	0.0009	0.0001	0.0010	0.0010	1.7956	0.4783
Median	0.0789	0.1148	0.0107	0.0000	0.0028	0.4181	0.3300	28.0000	34.6737
Mode	0.1023	0.1493	0.0000	0.0000	0.0000	0.4181	0.2665	28.0000	33.3982
Standard deviation	0.0111	0.0427	0.0361	0.0309	0.0024	0.0325	0.0330	60.4413	16.1005
Sample variance	0.0001	0.0018	0.0013	0.0010	0.0000	0.0011	0.0011	3653.154	259.226
Kurtosis	0.0746	−0.5786	−0.5435	−0.8526	1.0700	−0.4130	0.0455	13.8117	−0.1564
Skewness	0.2561	0.4731	0.7555	0.6297	0.7403	−0.3986	−0.1940	3.4696	0.4224
Range	0.0608	0.1806	0.1503	0.1127	0.0131	0.1625	0.1662	364.000	80.2674
Minimum	0.0514	0.0448	0.0000	0.0000	0.0000	0.3173	0.2480	1.000	2.3318
Maximum	0.1122	0.2254	0.1503	0.1127	0.0131	0.4798	0.4141	365.000	82.5992
Sum	88.957	133.495	36.127	30.552	3.084	467.423	373.363	49916.00	40604.43
Confidence level (95.0%)	0.0006	0.0025	0.0021	0.0018	0.0001	0.0019	0.0019	3.5232	0.9385

Table 2
Correlation coefficients between all pairs of the explanatory variables.

Variable	*W*	*C*	*B*	*F*	*S*	*CA*	*FA*	*A*
W	1.000	−0.252	−0.425	−0.078	0.053	−0.091	−0.194	0.085
C	−0.252	1.000	−0.291	0.110	0.054	−0.273	−0.303	−0.059
B	−0.425	−0.291	1.000	−0.174	0.375	−0.082	−0.002	−0.169
F	−0.078	0.110	−0.174	1.000	−0.569	−0.294	−0.407	0.253
S	0.053	0.054	0.375	−0.569	1.000	−0.250	0.155	−0.213
CA	−0.091	−0.273	−0.082	−0.294	−0.250	1.000	−0.130	−0.119
FA	−0.194	−0.303	−0.002	−0.407	0.155	−0.130	1.000	−0.119
A	0.085	−0.059	−0.169	0.253	−0.213	0.019	−0.119	1.000

Fig. 6 presents the frequency histograms of the variables. This database has been already employed by Mousavi et al. [2] to develop prediction models for the compressive strength of HPC mixes using the GP/OLS method. The descriptive statistics of the variables are given in Table 1.

Some of the HPC property variables may be fundamentally interdependent. The first step in the analysis of interdependency of the data is to make a careful study of what it is that these variables are measuring, noting any highly correlated pairs. High positive or negative correlation coefficients between the pairs may lead to poor performance of the models and difficulty in interpreting the effects of the explanatory variables on the response. This interdependency can cause problems in analysis as it will tend to exaggerate the strength of relationships between variables. This is a simple case commonly known as the problem of multicollinearity [32]. Thus, the correlation coefficients between all possible pairs are determined and shown in Table 2. As can be seen in this table, there are not high correlations between the predictor variables.

For the GEP and regression-based analyses, the database is randomly divided into training and validation subsets. The training data are used for learning (genetic evolution). The validation data are used to measure the performance of the models on data that play no role in building them. In order to obtain a consistent data division, several combinations of the training and validation sets are considered. For the analysis, 907 values (80%) of the data are taken for the training process and the rest of the values (20%) are used for validation of the generalization capability of the models.

3.2. Parameters for performance evaluation

The best model was chosen on the basis of a multi-objective strategy as follows:

i. Selecting the simplest model, although this is not a predominant factor.
ii. Providing the best fitness value on the training data.
iii. Providing the best fitness value on the validation data.

The first objective can be controlled by the user through the parameter settings (e.g., head size or number of genes). For the other objectives, the following objective function (OBJ) is used as a measure of how well the model predicted output agrees with the experimentally measured output. Selection of the best GEP model is deduced by the minimization of the following function [17]:

$$\mathrm{OBJ} = \left(\frac{\mathrm{No.}_{\mathrm{Train}} - \mathrm{No.}_{\mathrm{Validation}}}{\mathrm{No.}_{\mathrm{All}}}\right)\frac{\mathrm{MAE}_{\mathrm{Train}}}{R^2_{\mathrm{Train}}} + \frac{2\mathrm{No.}_{\mathrm{Validation}}}{\mathrm{No.}_{\mathrm{All}}} \times \frac{\mathrm{MAE}_{\mathrm{Validation}}}{R^2_{\mathrm{Validation}}} \qquad (4)$$

where $\mathrm{No.}_{\mathrm{Train}}$, $\mathrm{No.}_{\mathrm{Validation}}$ and $\mathrm{No.}_{\mathrm{All}}$ are respectively the number of training, validation and whole of data; *R* and MAE are, respectively, correlation coefficient and mean absolute error calculated using the following equations:

Table 3
Parameter settings for the GEP algorithm.

Parameter settings		
General	Chromosomes	200
	Genes	3-January
	Head size	3, 5, 8
	Tail size	9
	Dc size	9
	Gene size	26
	Linking function	Multiplication
Complexity increase	Generations without change	2000
	Number of tries	3
	Max. complexity	5
Genetic operators	Mutation rate	0.044
	Inversion rate	0.1
	IS transposition rate	0.1
	RIS transposition rate	0.1
	One-point recombination rate	0.3
	Two-point recombination rate	0.3
	Gene recombination rate	0.1
	Gene transposition rate	0.1

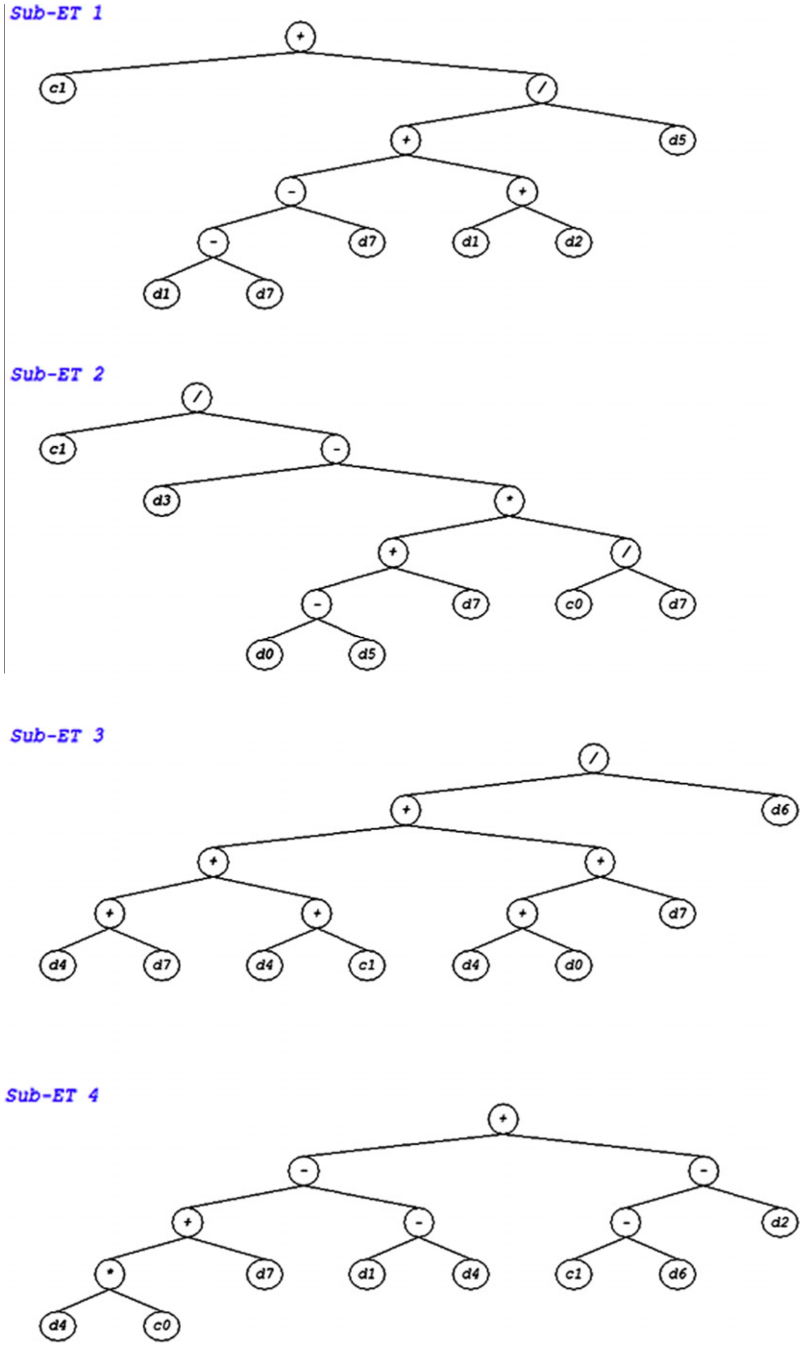

Fig. 7. Expression tree for the compressive strength (ETs) ($d_0, \ldots, d_7$ denote C, B, F, W, S, CA, FA, and A, respectively. c_0 and c_1 are constants).

$$R = \frac{\sum_{i=1}^{n}(h_i - \overline{h_i})(t_i - \overline{t_i})}{\sqrt{\sum_{i=1}^{n}(h_i - \overline{h_i})^2 \sum_{i=1}^{n}(t_i - \overline{t_i})^2}} \tag{5}$$

$$\mathrm{MAE} = \frac{\sum_{i=1}^{n}|h_i - t_i|}{n} \tag{6}$$

in which h_i and t_i are respectively actual and calculated outputs for the ith output, $\overline{h}$ and $\overline{t_i}$ are respectively the averages of the actual and predicted outputs, and n is the number of samples. It is well known that the R value alone is not a good indicator of prediction accuracy of a model. This is because, on equal shifting of the output values of a model, the R value does not change. The constructed objective function simultaneously takes into account the changes of R and MAE. Higher R and lower MAE values result in lowering OBJ, and hence indicate a more precise model. In addition, the above function takes the effects of different data divisions into account for the training and validation data [2,17].

3.3. Development of empirical model using GEP

W, C, B, F, S, CA, FA and A are used to create the GEP model. Various parameters involved in the GEP algorithm are shown in Table 3. In this study, basic arithmetic operators (+, −, ×, /) and some basic mathematical functions ($\sqrt{}$, exp) are utilized to get the optimum GEP model. The population size (number of chromosomes) sets the number of programs in the population. A run will take longer with a larger population size. The proper number of population depends on the number of possible solutions and complexity of the problem. A large number of chromosomes are tested

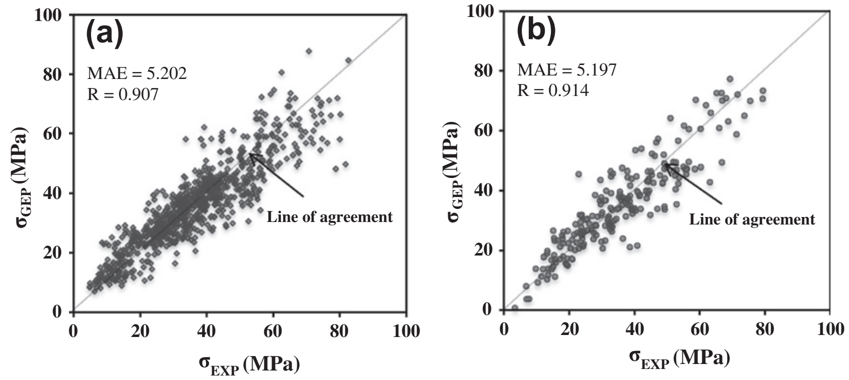

Fig. 8. Experimental versus predicted compressive strength using the GEP model: (a) training and (b) validation data.

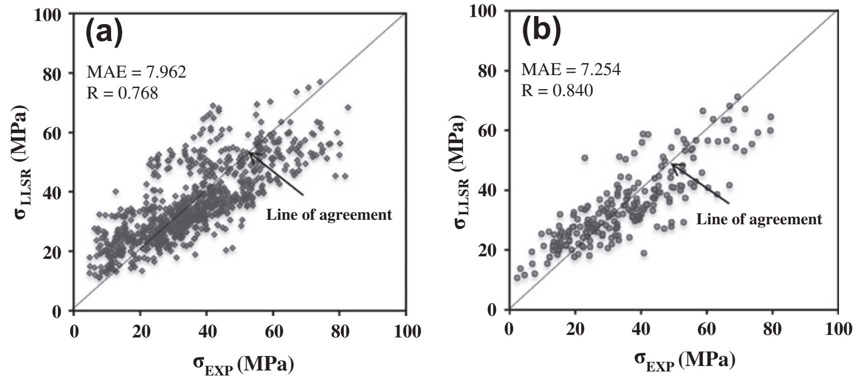

Fig. 9. Experimental versus predicted compressive strength using the LLSR model: (a) training and (b) validation data.

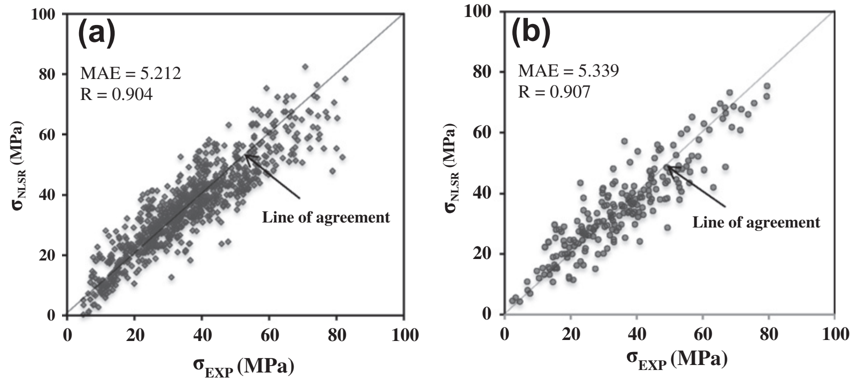

Fig. 10. Experimental versus predicted compressive strength using the NLSR model: (a) training and (b) validation data.

to find models with minimum error. The program was run until there was no longer significant improvement in the performance of the models or the runs automatically terminated. The chromosome architectures of the models evolved by GEP include head size and number of genes. The head size determines the complexity of each term in the evolved model. The number of terms in the model is determined by the number of genes per chromosome. Each gene codes for a different sub-expression tree or sub-ET. Different values are tested for the head size and number of genes. For the number of genes greater than one, the multiplication linking function is used to link the mathematical terms encoded in each gene. The MAE function is used to calculate the overall fitness of the evolved programs. The period of time acceptable for evolution to occur without improvement in best fitness is set via the generations without change parameter. After 2000 generations considered herein, a mass extinction or a neutral gene is automatically added to the model [24,25]. The values of the other involved parameters are selected on the basis of both previously suggested values [21–25] and a trial and error approach. For developing the GEP-based empirical models, a computer software, called GeneXproTools [33] is used.

3.3.1. GEP-based formulation for compressive strength of HPC mixes

The GEP-based formulation of the compressive strength of HPC mixes, for the best OBJ value, is as given below:

$$\sigma = \frac{-5.69(2.89 + (2.64B - 3.19A + F)/CA)(8.72S + 1.93A - 33.60 + C)(16.08S + A - B - 196.26 - FA - F)}{FA(W - (C - CA + A)/A)} \quad (7)$$

in which compressive strength σ is in MPa, the water content W, the cement C, the blast furnace slag B, the fly ash F, the superplasticizer

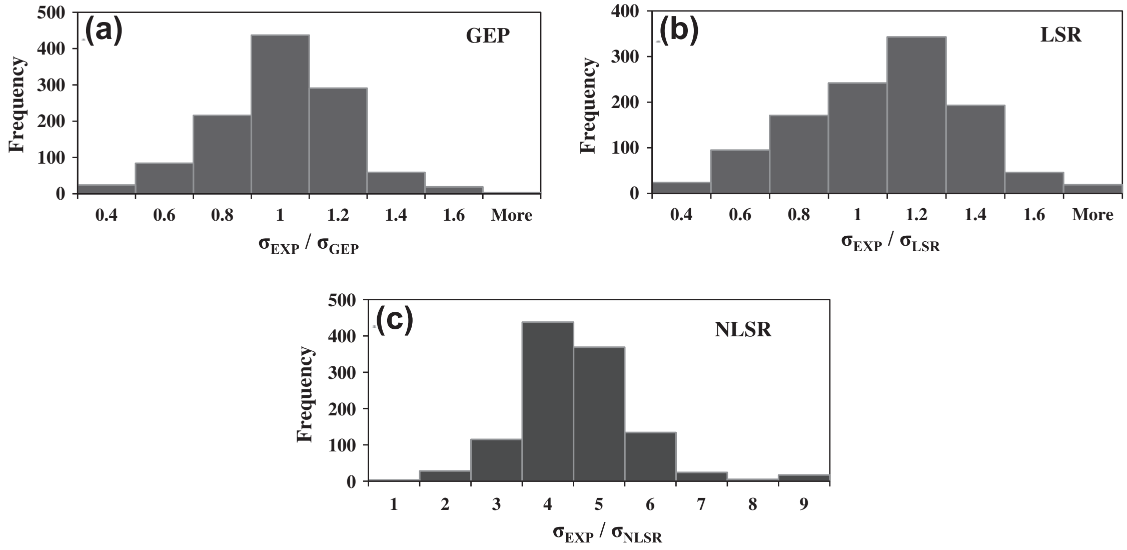

Fig. 11. A comparison of the ratio of the experimental to predicted compressive strength values using histograms of the models: (a) GEP, (b) LLSR, (c) NLSR.

Table 4
Statistical parameters of the GEP models.

Item	Formula	Condition	GEP
1	R	0.8 < R	0.914
2	$k = \frac{\sum_{i=1}^{n}(h_i \times t_i)}{h_i^2}$	0.85 < K < 1.15	1.002
3	$k' = \frac{\sum_{i=1}^{n}(h_i \times t_i)}{t_i^2}$	0.85 < K' < 1.15	0.969
where	$Ro^2 = 1 - \frac{\sum_{i=1}^{n}(t_i - h_i^o)^2}{\sum_{i=1}^{n}(t_i - \overline{t_i})^2}, \quad h_i^o = k \times t_i$		1.000
	$Ro'^2 = 1 - \frac{\sum_{i=1}^{n}(h_i - t_i^o)^2}{\sum_{i=1}^{n}(h_i - \overline{h_i})^2}, \quad t_i^o = k' \times h_i$		0.994

h_i: Actual output value for the ith output; t_i: predicted output value for the ith output; $\overline{h_i}$: average of actual outputs; $\overline{t_i}$: average of predicted outputs; n: number of samples.

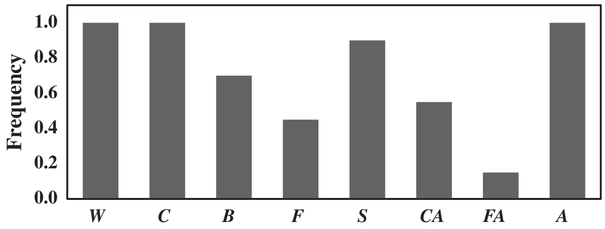

Fig. 12. Contributions of the predictor variables in the GEP analysis.

S, the coarse aggregate CA, and the fine aggregate FA are in kg/m^3; and the age of specimens A is in day. The expression tree of the above formulation is shown in Fig. 7. A comparison of the experimental compressive strength of HPC versus values predicted by GEP is shown in Fig. 8. The GEP model yields low OBJ value equal to 6.281.

3.4. Development of empirical model using regression analysis

In the conventional material modeling process, regression analysis is an important tool for building a model. In this study, multivariable linear least squares regression (LLSR) and nonlinear least squares regression (NLSR) [34] analyses are performed to have an idea about the predictive power of the GEP technique, in comparison with a classical statistical approach. The LLSR method is extensively used in regression analysis primarily because of its interesting nature. LLSR minimizes the sum-of-squared residuals for each equation, accounting for any cross-equation restrictions on the parameters of the system. NLSR extends LLSR for use with a much larger and more general class of functions. Eviews software package [35] was used to perform the LLSR and NLSR analysis. The LLSR-based formulation of σ in terms of W, C, B, F, S, CA, FA, and A is as follows

$$\sigma = -0.0902W + 0.1279C + 0.1089B + 0.1065F + 0.3079S + 0.0229CA + 0.03116FA + 0.1089A - 50.7715 \tag{8}$$

Several NLSR-based equations are established for the prediction of σ. Selection of the best NLSR model is made by considering the minimum OBJ value. The best NLSR prediction model for σ is as given below:

$$\sigma = -187.2684 + \exp\Big(-2.2380 + 1.0346W^{-0.2896} + 0.0066C^{0.6647} + 0.00057B + 0.0005F + 0.0005S + 0.0159CA^{0.4517} + 0.7943FA^{0.1038} + 5.0006A^{0.0078}\Big) \tag{9}$$

Comparisons of the experimental σ of HPC versus values predicted by LLSR and NLSR are shown in Figs. 9 and 10, respectively. The LLSR and NLSR models yield OBJ values equal to 12.221 and 6.320, respectively.

4. Performance analysis and model validity

A new prediction equation is developed for the compressive strength of HPC upon a reliable database. Based on a rational hypothesis, Smith [36] suggested the following criteria for judging performance of a model:

- If a model gives $|R| > 0.8$, a strong correlation exists between the predicted and measured values.

In all cases, the error values (e.g., MAE) should be at the minimum. It can be observed from Fig. 8 that the GEP model with high R and low MAE, and therefore low OBJ, values is able to predict the target values to an acceptable degree of accuracy. Meanwhile, the RMSE and MAE values for the model are not only low but also as similar as possible for the training and testing sets. This suggests that the proposed model have a very good predictive ability (low values) and generalization performance (similar values) [37]. The GEP model outperforms the LLSR and NLSR models. Fig. 11 visualizes a comparison of the compressive strength predictions made by the models on the whole of data. No rational model to predict the compressive strength of HPC mixes has been developed yet that would encompass the influencing variables considered in this study. Therefore, it is not possible to conduct a comparative study between the results of this research and those of previous studies.

According to Frank and Todeschini [38], the minimum ratio of the number of objects over the number of selected variables for model acceptability is 3. A safer value of 5 is suggested to be more reasonable [38]. In the present study, this ratio is much higher and is equal to 1133/8 = 141.6. Furthermore, new criteria recommended by Golbraikh and Tropsha [39] are checked for external validation of the GEP model on the validation data sets. It is suggested that at least one slope of regression lines (k or k') through the origin should be close to 1. Either the squared correlation coefficient (through the origin) between predicted and experimental values (Ro^2), or the coefficient between experimental and predicted values (Ro'^2) should be close to 1 [24,25]. Models are considered valid, if they satisfy the conditions. The validation criteria and the relevant results obtained by the GEP model are presented in Table 4. The results ensure that the derived model is strongly valid, has the prediction power and is not established by chance.

Although the proposed nonlinear regression model yields relatively good results for the current database, empirical modeling based on statistical regression techniques has significant limitations. Most commonly used regression analyses can have large uncertainties. In most cases, the best models developed using the commonly used regression approach are obtained after controlling only few equations established in advance. Thus, they cannot efficiently consider the interactions between the dependent and independent variables. A major advantage of GEP lies in its powerful ability to model the mechanical behavior without any prior assumptions. The best solutions (equations) evolved by this technique are determined after controlling numerous preliminary models, even millions of linear and nonlinear models [25]. Also, the user physical insight and the shape of the classical relationships can be regarded in making propositions on the elements and structure of the evolved equations. As more data become available, including those for other HPC mixtures and test conditions, the proposed model can be improved to make more accurate predictions for a wider range. However, one of the goals of introducing GP-based methods into the design processes is better handling of the information in pre-design phase [40]. It is idealistic to have some initial estimates of the outcome before performing any extensive laboratory or field work. The GEP approach employed in this research is based on the data alone to determine the structure and parameters of the model. Therefore, the derived model is considered to be mostly valid for use in preliminary design stages and should cautiously be used for final decision-making [25].

5. Sensitivity analysis

The contribution of the most relevant predictor variables (W, C, B, F, S, CA, FA and A) in the best GEP models is evaluated through a sensitivity analysis. For this aim, frequency values of the input variables are obtained. A frequency value equal to 1.00 for an input indicates that this variable has been appeared in 100% of the best thirty programs evolved by GEP. This methodology is a common approach in the GP-based analyses [24]. The frequency values of the predictor variables are presented in Fig. 12. According to these results, it can be found that W, C and A exert dominant influence on the variations of the compressive strength compared with the other inputs. The sensitivity analysis results are expected cases from the structural viewpoint.

6. Conclusions

In this study, a robust variant of GP, namely GEP is utilized to formulate the compressive strength of HPC mixes. An accurate empirical model for the prediction of the compressive strength is obtained. A widely dispersed database of previously published compressive strength of HPC test results is used for developing the prediction model. The GEP model is capable of predicting the compressive strength of HPC mixtures with high accuracy. The validity of the model is tested for a part of test results beyond the training data domain. The validation phase confirms the efficiency of the model for its general application to the compressive strength estimation of HPC mixes. Furthermore, the GEP prediction model efficiently satisfies the conditions of different criteria considered for its external validation. The derived model was benchmarked against multivariable linear and nonlinear regression models. Due to nonlinearity in the compressive strength behavior, the nonlinear GEP model produces better outcomes than the regression-based models. The proposed model simultaneously takes into account the effect of several important factors representing the compressive strength behavior. Using the derived model, the compressive strength can easily be estimated from the HPC mixture basic properties. Thus, there is no need to go through sophisticated and time-consuming laboratory tests. This can be regarded as a major advantage of the GEP model. The developed model can reliably be used for practical pre-planning and pre-design purposes in that it is derived from tests on mixtures with a wide range of aggregate gradation and properties.

Acknowledgements

The authors are thankful to the respected authorities of the Islamic Azad University of Shahrood (IAU-Shahrood). The authors are also thankful to Prof. I-Cheng Yeh for providing the experimental database.

References

[1] Goodspeed CH, Vanikar S, Cook R. High-performance concrete (HPC) defined for highway structures. Concr Int 1996;18:62–7.

[2] Mousavi SM, Gandomi AH, Alavi AH, Vesalimahmood M. Modeling of compressive strength of HPC mixes using a combined algorithm of genetic programming and orthogonal least squares. Struct Eng Mech 2010;36:225–41.

[3] Domone P, Soutsos M. An approach to the proportioning of high-strength concrete mixes. Concr Int 1994;16:26–31.

[4] Yeh IC. Modeling of strength of high-performance concrete using artificial neural networks. Cem Concr Res 1998;28:1797–808.

[5] Yeh IC. Modeling slump flow of concrete using second-order regressions and artificial neural networks. Cem Concr Compos 2007;29:474–80.

[6] Basma AA, Barakat S, Oraimi SA. Prediction of cement degree of hydration using artificial neural networks. Mater J 1999;96:166–72.

[7] Ji T, Lin T, Lin X. A concrete mix proportion design algorithm based on artificial neural networks. Cem Concr Res 2006;36:1399–408.

[8] Jepsen MT. Predicting concrete durability by using artificial neural network. Published in a special NCR-publication 2002: ID 5268; 2002.

[9] Yeh IC. Exploring concrete slump model using artificial neural networks. J Comput Civil Eng 2006;20:217–21.

[10] Kasperkiewicz J, Racz J, Dubrawski A. HPC strength prediction using artificial neural network. J Comput Civil Eng 1995;9:279–84.

[11] Raghu Prasad BK, Eskandari H, Venkatarama Reddy BV. Prediction of compressive strength of SCC and HPC with high volume fly ash using ANN. Constr Build Mater 2009;23:117–28.

[12] Rajasekaran S, Amalraj R. Predictions of design parameters in civil engineering problems using SLNN with a single hidden RBF neuron. Comput Struct 2002;80:2495–505.

[13] Rajasekaran S, Suresh D, Pai GAV. Application of sequential learning neural networks to civil engineering modeling problems. Eng Comput 2002;18:138–47.

[14] Rajasekaran S, Lavanya S. Hybridization of genetic algorithm with immune system for optimization problems in structural engineering. Struct Multidiscip Optim 2007;34:415–29.

[15] Koza JR. Genetic programming: on the programming of computers by means of natural selection. Cambridge: MIT Press; 1992.

[16] Gandomi AH, Alavi AH. Multi-stage genetic programming: a new strategy to nonlinear system modeling. Inf Sci 2011;181:5227–39.

[17] Gandomi AH, Alavi AH. A new multi-gene genetic programming approach to nonlinear system modeling. Part I: materials and structural engineering problems. Neural Comput Appl 2011; in press. doi: 10.1007/s00521-011-0734-z.

[18] Gandomi AH, Alavi AH. A new multi-gene genetic programming approach to nonlinear system modeling. Part II: geotechnical and earthquake engineering problems. Neural Comput Appl 2011; in press. doi: 10.1007/s00521-011-0735-y.

[19] Ferreira C. Gene expression programming: a new adaptive algorithm for solving problems. Complex Syst 2001;13:87–129.

[20] Oltean M, Grosan C. A comparison of several linear genetic programming techniques. Complex Syst 2003;14:1–29.

[21] Baykasoglu A, Gullub H, Canakci H, Ozbakir L. Prediction of compressive and tensile strength of limestone via genetic programming. Expert Syst Appl 2008;35:111–23.

[22] Cevik A, Cabalar AF. Modelling damping ratio and shear modulus of sand–mica mixtures using genetic programming. Expert Syst Appl 2009;36:7749–57.

[23] Alavi AH, Gandomi AH. A robust data mining approach for formulation of geotechnical engineering systems. Eng Comput 2011;28:242–74.

[24] Gandomi AH, Alavi AH, Mirzahosseini MR, Moqhadas Nejad F. Nonlinear genetic-based models for prediction of flow number of asphalt mixtures. J Mater Civil Eng ASCE 2011;23:248–63.

[25] Gandomi AH, Tabatabaie SM, Moradian MH, Radfar A, Alavi AH. A new prediction model for load capacity of castellated steel beams. J Constr Steel Res 2011;67:1096–105.

[26] Gandomi AH, Alavi AH, Sadat Hosseini SS. A discussion on Genetic programming for retrieving missing information in wave records along the west coast of India. Appl Ocean Res 2008;30:338–9.

[27] Ferreira C. Gene expression programming: mathematical modeling by an artificial intelligence. 2nd ed. Germany: Springer-Verlag; 2006.

[28] Popovics S. Analysis of the concrete strength versus water–cement ratio relationship. ACI Mater J 1990;87:517–29.

[29] Yeh IC. Analysis of strength of concrete using design of experiments and neural networks. J Mater Civil Eng 2006;18:597–604.

[30] Yeh IC, Lien L. Knowledge discovery of concrete material using Genetic Operation Trees. Expert Syst Appl 2009;36:5807–12.

[31] Chen L, Wang T. Modeling strength of high-performance concrete using an improved grammatical evolution combined with macro genetic algorithm. J Comput Civil Eng 2009;24:281–8.

[32] Dunlop P, Smith S. Estimating key characteristics of the concrete delivery and placement process using linear regression analysis. Civil Eng Environ Syst 2003;20:273–90.

[33] GEPSOFT. GeneXproTools, Version 4.0.; 2006. <http://www.gepsoft.com>.

[34] Ryan TP. Modern regression methods. New York: Wiley; 1997.

[35] Maravall A, Gomez V. EViews Software, Ver. 5. Irvine, CA: Quantitative Micro Software, LLC; 2004.

[36] Smith GN. Probability and statistics in civil engineering. London: Collins; 1986.

[37] Pan Y, Jiang J, Wang R, Cao H, Cui Y. A novel QSPR model for prediction of lower flammability limits of organic compounds based on support vector machine. J Hazard Mater 2009;68:962–9.

[38] Frank IE, Todeschini R. The data analysis handbook. Amsterdam, The Netherland: Elsevier; 1994.

[39] Golbraikh A, Tropsha A. Beware of q^2. J Mol Graph Model 2002;20:269–76.

[40] Kraslawski A, Pedrycz W, Nystrom L. Fuzzy neural network as instance generator for case-based reasoning system: an example of selection of heat exchange equipment in mixing. Neural Comput Appl 1999;8:106–13.