

Prediction of creep in concrete using genetic programming hybridized with ANN

Osama A. Hodhod¹, Tamer E. Said*² and Abdulaziz M. Ataya³

¹Department of Structural Engineering, Faculty of Engineering, Cairo University, Giza, Egypt

²Engineering Division, National Research Centre, Cairo, Egypt

³Structural Engineer, Stockholm, Sweden

(Received June 16, 2017, Revised September 13, 2017, Accepted January 11, 2018)

Abstract. Time dependent strain due to creep is a significant factor in structural design. Multi-gene genetic programming (MGGP) and artificial neural network (ANN) are used to develop two models for prediction of creep compliance in concrete. The first model was developed by MGGP technique and the second model by hybridized MGGP-ANN. In the MGGP-ANN, the ANN is working in parallel with MGGP to predict errors in MGGP model. A total of 187 experimental data sets that contain 4242 data points are filtered from the NU-ITI database. These data are used in developing the MGGP and MGGP-ANN models. These models contain six input variables which are: average compressive strength at 28 days, relative humidity, volume to surface ratio, cement type, age at start of loading and age at the creep measurement. Practical equation based on MGGP was developed. A parametric study carried out with a group of hypothetical data generated among the range of data used to check the generalization ability of MGGP and MGGP-ANN models. To confirm validity of MGGP and MGGP-ANN models; two creep prediction code models (ACI209 and CEB), two empirical models (B3 and GL 2000) are used to compare their results with NU-ITI database.

Keywords: Multi-Gene genetic programming; artificial neural network; artificial intelligence; hybrid; creep; concrete

1. Introduction

Concrete is a composite material with an extremely heterogeneous and complicated structure. This makes troubles in creating exact models of concrete structure from which its behavior can be surely predicted. An adequate description and knowledge of the mechanical properties of the material as the prediction of the time-dependent strains are important to assess the strength and serviceability of the elements in the reinforced and prestressed concrete structures. The time-dependent strains can be classified as being due to creep and/or shrinkage. These components of strains should be considered through the design phases of the structure.

Generally, when a load is applied to a concrete specimen, the specimen initially displays a direct elastic deformation followed by a slow continual inelastic deformation. This slow rate of increasing of inelastic deformation was discovered in 1907 by Hatt (Bazant 1982) and was named creep. The creep is one of the important strains that affect the serviceability, durability and long period reliability of concrete structures.

Creep of concrete is both an attractive phenomenon because it imparts a degree of necessary ductility to the concrete. On the other hand time-dependent phenomena of creep usually modify service stresses and could lead the

concrete structures to be on the unsafe side (Lozano-Galant and Turmo 2014). Creep is often responsible for extreme deflections at service loads, which can result in the instability of shell structures, or arch, cracking, creep buckling of long columns and loss of prestress. Oftentimes the detrimental effects of creep are more damaging to non-load-bearing elements related with the structure, for instance cladding panels, window frames and partitions, than they are to the structure itself (Fanourakis and Ballim 2003).

The most essential factors that affect the creep of concrete is its initial humidity and the speed of removal of moisture (Klovanych 2015). So, creep can be divided into basic creep and drying creep: Basic creep can be understood as time-dependent increase in strain under continued constant load of a concrete specimens that are sealed to inhibit the ingress or egress of humidity from or to its environment. It is considered a material essential property and independent of the effects of the specimen size and shape factor. Drying creep as the name implies, is the strain remaining after deducting shrinkage, elastic, and basic creep strains from the entire measured strain on nominally identical specimens in drying environmental conditions. The average creep of a cross section at drying is highly size-dependent.

Over the last 35 years, a lot of models have been suggested for the estimate of compliance in concrete. Several factors impact the value and rate of development of creep, containing the properties of the concrete mix, aggregate fraction, aggregate stiffness (elastic modulus), fine aggregate-to total aggregate ratio, volume-to-surface

*Corresponding author, Ph.D.
E-mail: SPORTNOLT@yahoo.com

ratio (V/S), humidity, age of concrete at loading, temperature and stress level.

Compliance $J(t, t_0)$: The total load induced strain (elastic strain in addition to creep strain) at time t per unit stress affected by a unit uniaxial sustained load applied since loading time t_0 , (ACI-Committee209 2008) as presented in Eq. (1).

$$\text{Compliance} = \frac{(\text{elastic strain} + \text{basic creep} + \text{drying creep})}{\text{stress}} \quad (1)$$

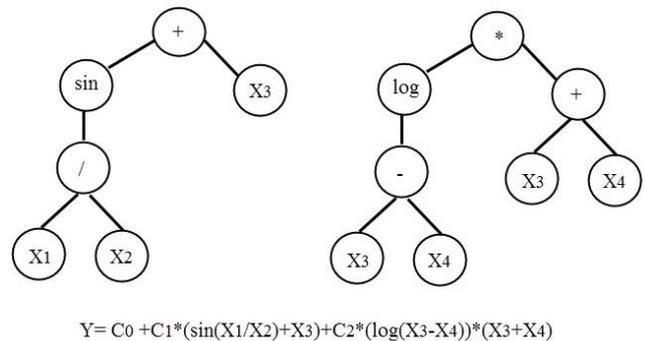
Designers usually use one of the next two code models to assess creep strain in concrete. The first model is the CEB MC90-99 Model (CEB 1999) suggested by the Euro-international concrete committee and the second is ACI 209 (Branson and Christiason 1971), suggested by the American Concrete Institute. With the presence of computers and the more experimental data from a lot of countries, new prediction models have been suggested by researchers to compute creep. Two famous models -the Bazant-Baweja B3 (Bažant and Baweja 2000) model and the GL2000 model (Gardner 2004) are differ in their level of complexity.

Artificial neural networks are used in the last ten years to predict creep in concrete. (Taha *et al.* 2003) have developed an artificial neural network (ANN) to predict creep in structural masonry in the year 2003. They used neural network with one hidden layer that contain six neurons. (Abed *et al.* 2010) presented a creep model based on focused time-delay neural network (FTDNN) for prediction of creep in brickwork structure. Then, (Abed and Osman 2013) introduced a model depended on non-linear auto-regression with exogenous inputs (Narx) which considers time dependency. (Karthikeyan *et al.* 2008) developed a neural network for prediction of creep coefficient in concrete by training the network with some experimental and CEB 90 predictions data. Lately (Bal and Buyle-bodin 2014) suggested a neural network for prediction of creep in concrete. This network consists of two hidden layers, each one contains eight hidden neurons. They used a RILEM data base that consider an earlier version of NU-ITI (Bažant and Li 2008) database.

2. Multi-Gene Genetic Programming (MGGP)

Genetic programming (GP) is specialization subclass of genetic algorithms which are based on the principles of genetics and natural selection (Holland 1992). GP considered as an evolutionary computation method that have capability to solve problem without external interference to guide the computer exactly how to solve it (Fulcher and Lakhmi 2008).

Genetic programming has been successfully used for solving a number of nonlinear civil engineering problems such as Predicting of compressive strength of recycled aggregate concrete (Abdollahzadeh *et al.* 2016), Predicting of Shear strength of RC beams (Cladera *et al.* 2014), and Prediction of the bond strength of ribbed steel bars in concrete (Golafshani *et al.* 2014).



$$Y = C_0 + C_1 * (\sin(X_1/X_2) + X_3) + C_2 * (\log(X_3 - X_4)) * (X_3 + X_4)$$

Fig. 1 Example of a multigene symbolic model

Multi objective genetic programming based creep model has been established by Gandomi *et al.* (2016). The proposed multi objective genetic programming technique can be used to predict creep compliance in concrete with higher accuracy.

Multi-gene genetic programming (MGGP) is a promising technique that extended from genetic programming. Each individual in MGGP consists of a number of traditional genetic programming trees. Fig. 1 shows an individual in MGGP that contains two trees (Genes). The weights of the multiple gene model C_0 , C_1 and C_2 are determined by least square. The model structure in Fig. 1 contains non-linear terms (sin and log). However, this model is linear in the parameters with regard to the coefficient C_0 , C_1 and C_2 , (Ao *et al.* 2011). Subsequently, Multigene genetic programming integrates the advantage of classical linear regression with the capability to represent non-linear behavior. GPTIPS is a free, open source MATLAB based software platform for symbolic data mining (SDM) that use the Multigene genetic programming as the engine that drives the automatic model discovery process (Searson 2009, Searson 2014).

The population contains number of individuals that are randomly generated in the initial population by several methods. In this work, the Ramped half-and-half method is used in initializing the population. The individuals are subjected to some of evolutionary operation. As the GP algorithm is continues, the mean fitness of the individuals in addition to fitness of best individuals are improved.

The higher the fitness individual, the higher is the probability to be selected in selection operator. This is similar to Darwin's theory of continued existence of the fittest. The tournament selection technique is used in this work. This is repeated a number of times for the same number of selections. The scheme of the GP is demonstrated in Fig. 2.

The following steps are performed to create an individuals in genetic programming (GP), (Negnevitsky 2005):

1. Specify the probabilities of some operations like, crossover, mutation and cloning which should be equal to one. In addition, specify the maximum number of generations.
2. An initial population of size N of individuals (programs) is created by randomly mixed functions and terminals.

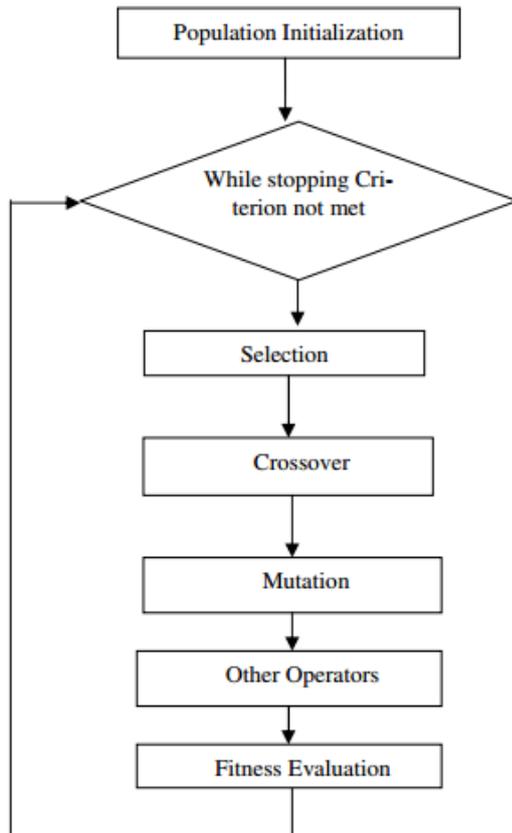


Fig. 2 Genetic programming scheme (Shukla *et al.* 2010)

3. Return the best individuals of the run by calculate the fitness of each program. The fitness is calculated by using a suitable fitness function. The fitness function is essential and important in the GP algorithm. The fitness function is defined accordingly to the problem type and complexity. The fitness function aims to evaluate the fitness and performance of each individual in the generations. The evolutionary operations are implemented on the selected individuals after evaluate it by the fitness function.
4. According to the specified probabilities, one of the genetic operations of crossover, mutation and cloning are selected. In the cloning operator, one individual is selected and moved to next generation without any modification. In the crossover operation, one pair of individuals that give one pair of offspring is moved to next generation. In the mutation operation, one individual is selected and subjected to mutation operation. Then, the mutated individual is moved to next generation. All individuals are selected with a probability corresponding to their fitness (i.e., the higher the fitness, the more probability the individual is to be chosen).
5. Step 4 is repeated until the size (number of individuals) of new population becomes equal to the size (number of individuals) of initial population.
6. The present population (parent) is replaced by the new population (offspring's).
7. Go back to step 3 and the process is repeated until one of termination is verified.

3. Artificial neural networks

Artificial neural networks considered as a kind of artificial intelligence that work to simulate the approach of how human brain stores and manages data. It depends on making connections between processing components, called neurons (Taylor 2006). Learning is a procedure by which the free parameters of a neural network are adjusted through a process of stimulation by the environment in which the network is embedded (Hakin 1999). In the supervised learning, the neural network is updated its weights by comparison between output of the network and the target until the output of network agrees of the target. The backpropagation algorithm is a supervised learning method. Many different backpropagation training algorithms are available, they have a different computation and time processing by computers. The Levenberg-Marquardt backpropagation algorithm is the fastest algorithm, so it was used in this study.

A feedforward neural network has a layered architecture. Each layer contains a number of neurons which get their input from neurons in the preceding layer straight and send their output signals to the following layer. The movement of information is unidirectional. There are no connections inside a layer. The most two public kind of feedforward networks is Multilayer Perceptron (MLP) and General Feedforward (GFF) networks. Thus, several researchers utilized a multilayer feedforward network to construct their models. The multilayer feedforward neural network distinguishes itself by the presence of one or more hidden layers, whose computation nodes are correspondingly called hidden neurons or hidden units, the function of hidden neurons is to intervene between the external input and the network output in some useful manner.

4. Database

The data sets used in this study are extracted from the NU-ITI database (Bazant and Li 2008). Some of data sets in NU-ITI database were excluded in this study due to lack of some information about average compressive strength at 28 days, volume to surface ratio, relative humidity, cement type and age at start of loading values. Later, all data sets that have specimen temperature out of the range $20\pm 3^{\circ}\text{C}$ were excluded. Only three additives were existing in few data sets used in this study: water reducer, silica fume (SiO_2) and Fly ash, the content of this additives were not exceeded 4.8%, 15.2% and 26.6% of cement weight respectively. After this filtering of NU-ITI database, the number of data sets used in this study are 187 experimental data sets including 4242 creep data points.

5. Models inputs and output

There is disagreement about the information should be utilized to calculate the compliance creep of concrete. According to (ACI-Committee209 2008), the creep models should contain at least the following parameters: Strength of

Table 1 Inputs and output parameters with their range of variation

Parameters type	Description	Variation range
Input	f_{cm28} : Average compressive strength at 28 days	$19 \leq f_{cm28} \leq 136$ MPa
	V/S: Volume/surface exposed to air	$12 \leq V/S \leq 128.57$ mm
	RH: Relative humidity	$40\% \leq RH \leq 100\%$
	CM: Cement type	I, II and III
	t_0 : Age at start of loading	$0.66 \leq t_0 \leq 3300$ day
	$t - t_0$: Age of beginning of creep measurement	$t \geq t_0$
Output	$J(t, t_0)$: The compliance function representing the strain at age t caused by sustained uniaxial stress applied at age t_0 .	$18.11 \leq J \leq 270.6$ (Microstrain/MPa)

concrete, Specimen size, Ambient relative humidity, Age at loading and Duration of loading.

Therefore, in this study, the inputs for creep prediction models will include the following inputs: Average compressive strength at 28 days (f_{cm28}), Relative humidity (RH) and V/S volume to surface ratio (V/S), Cement type (CT), Age at start of loading (t_0) and Age of beginning of creep measurement (t). The output is compliance $J(t, t_0)$. The inputs and output of data sets with their description and variation range are presented in Table 1.

To construct the MGGP and MGGP-ANN models, we should express some parameters as numerical codes so as to be input of the used models. Based on that, we supposed the following codes in this study for cement type: I=0.85, II=1 and III=1.1.

In this work, we used $\ln(t_0+1)$ and $\ln(t-t_0+1)$ as inputs for training MGGP and MGGP-ANN models instead of t_0 and $(t-t_0)$ respectively. The natural logarithm of time inputs provides better results of performance.

In addition, the relative humidity and volume to surface ratio are used as one input: $\left(\frac{100-RH}{V/S}\right)$. This input is equal to zero when the relative humidity is equal to 100 (i.e., basic creep, so the compliance is independent from the volume to surface ratio).

6. Application of MGGP technique in prediction of compliance in concrete

The MGGP technique needs a various parameters, some of these parameters are selected depend on some formerly recommended values (Searson *et al.* 2010, Muduli and Das 2013) and other after doing a lot of investigative

runs of GPTIPS toolbox on MATLAB software and evaluating the performance state (Searson 2009), (Searson 2014). The parameters values are summarized in Table 2. In this work, mathematical functions and main arithmetic operators are utilized to have the best MGGP models.

The population size determines the number of individuals that contain functions and terminals of the studied problem. This population is constantly improved

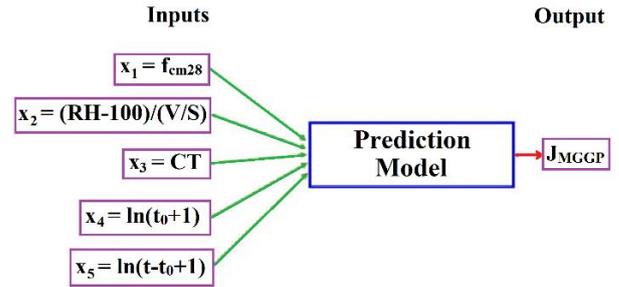


Fig. 3 Inputs and output used for construct MGGP and MGGP-ANN models

Table 2 Parameter settings for the MGGP

Parameter	Settings
Population size	200, 500, 800, 1200
Number of generations	50, 100, 200
Maximum number of genes allowed in an individual	1, 3, 6, 10
Maximum tree depth	2, 4, 7, 10
Tournament size	3, 10
Crossover events	80%
Mutation events	10%
Probability of GP tree direct copy	10%
Elitism	5% of population
High level crossover	20%
Low level crossover	80%
Function set	$+, -, \times, /, \sqrt{\quad}, \wedge^2, \exp, \sin, \tanh$

over a certain number of generations. The sufficient number of population and generation is related to the complication of problem and the amount of feasible solutions. Bearing in mind the increasing of numbers of population and generation not necessarily guide to better performance and results because of increasing level of noise.

The runs of individuals are automatically terminated when the number of generation or termination value are reached. The higher number of genes and higher depth of tree, the higher is the chance to fit data. However, increasing of these parameters leads to more complex solutions.

The individuals are selected by tournament selection method and subjected to various genetic operation to form a population in specific generation.

The data points are divided randomly into three parts such as 70% for the training process, 15% for test stage, and 15% for validation stage. The number and structure of the trees is evolved automatically through a run utilizing training data. Testing data are utilized to evaluate the evolved models. The validation data set can be specified to assistance mitigate against overfitting (Searson 2009).

There are $4 \times 3 \times 4 \times 4 \times 2 = 384$ various combinations of the parameters. Fitness function estimate the performance of individuals in each generation to get best solution. In this study the default fitness function of GPTIPS is utilized to reduce the error. GPTIPS toolbox is utilized to carry out MGGP technique, (Searson 2009, Searson 2014). 384 various runs of MGGP were applied by using GPTIPS tool with combination of parameters that specified in Table 2,

the mathematical formula for each model was discovered. The RMSE, R and ω_{BP} values were calculated. The lower the RMSE and ω_{BP} values, the higher is performance of model. On the other hand, the higher the R values, the higher is performance of model. Optimal MGGP Prediction Model for the concrete creep

The best MGGP was selected on the basis of providing the best fitness on the data (minimum of ω_{BP}) as well as the simplicity and smoothness of the models. The MGGP-800-50-6-7-10 is selected as the best model. The final formulation of the proposed model for the prediction of total creep compliance of concrete is obtained as shown through Eq. (2) to Eq. (8).

$$\text{Bias} = +36.6 \quad (2)$$

$$\text{Gene 1} = +1.7 * (X_3 * (2 * X_1 + X_5)) / X_4 \quad (3)$$

$$\text{Gene 2} = -0.1 * (X_1 * X_4 * X_5) / \text{SQRT}(X_1) \quad (4)$$

$$\text{Gene 3} = +140 * (\text{SQRT}(X_1 / (X_3 * (X_4)^2))) / (X_1 * X_4) \quad (5)$$

$$\text{Gene 4} = -3.8 * X_1 / X_4 \quad (6)$$

$$\text{Gene 5} = +5.6 * \text{SQRT}(\text{ABS}(5.2 - X_2)) * (X_3 * (X_3 / X_4 - X_3^2) + X_2 * (X_5 + 12.6)) / (X_4 * \text{SQRT}(\text{ABS}(X_4 - X_1))) \quad (7)$$

$$\text{Gene 6} = +448 * X_5 / (\text{SQRT}(\text{ABS}(X_4 + \text{SQRT}(X_1) - (X_1 * X_2 / X_4)))) + X_1 * X_3 \quad (8)$$

Where, $\text{SQRT}()$ refers to square root function. $\text{ABS}()$ refers to absolute value function.

The mathematical formula of optimal MGGP model is given in Eq. (9).

Variation of the best fitness (in log values) and mean fitness with the number of generations for optimal MGGP model are illustrated in Fig. 4. It can be observed from this figure that the fitness (RMSE) value decreases whenever the number of generations is increased. The best fitness is found at the 49th generation.

$$J_{MGGP} = \left(36.6 + 1.7 * \frac{CT * (2 * f_{cm28} + \ln(t - t_0 + 1))}{\ln(t_0 + 1)} - 0.1 * \frac{f_{cm28} * \ln(t_0 + 1) * \ln(t - t_0 + 1)}{\sqrt{f_{cm28}}} + 140 * \frac{\sqrt{f_{cm28}}}{\sqrt{CT * (\ln(t_0 + 1))^2}} - 3.8 * \frac{f_{cm28}}{\ln(t_0 + 1)} + 5.6 * \sqrt{\left| 5.189 - \frac{100 - RH}{V} \right|} * \left(CT * \left(\frac{CT}{\ln(t_0 + 1)} - CT^2 \right) + \frac{100 - RH}{V} \right) * \frac{\ln(t - t_0 + 1) + 12.6}{\ln(t_0 + 1) * \sqrt{|\ln(t_0 + 1) - f_{cm28}|}} \right) + 448$$

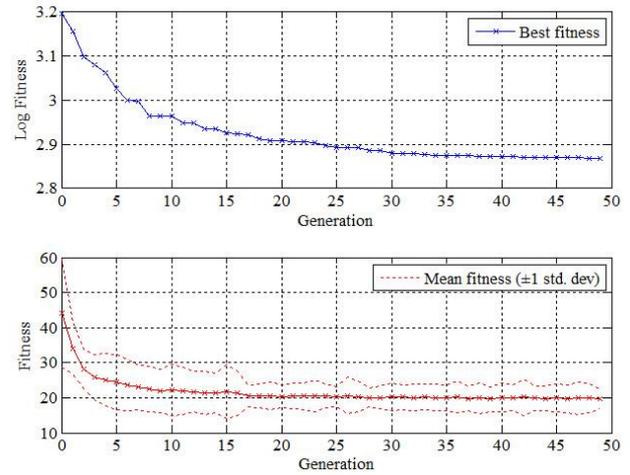


Fig. 4 Variation of the best and mean fitness with the number of generations for optimal MGGP model

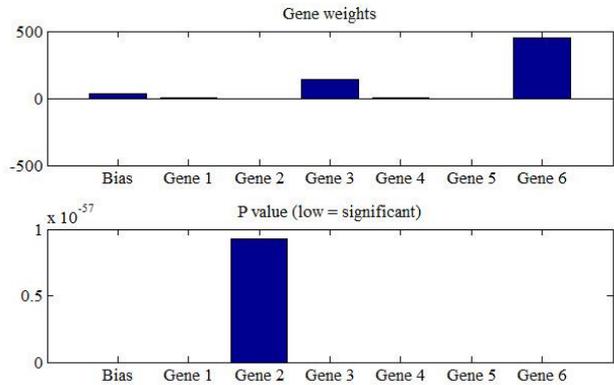


Fig. 5 Statistical properties of the optimal MGGP model

$$* \frac{\ln(t - t_0 + 1)}{\sqrt{\left| \ln(t_0 + 1) + \sqrt{f_{cm28}} - \frac{f_{cm28} * (100 - RH)}{\ln(t_0 + 1) * \frac{V}{S}} \right|} + f_{cm28} * CT} \quad (9)$$

The statistical importance of each of the six genes of the optimal model can be observed in Fig. 5.

It can be noticed from this figure the importance and weight of each gene evaluated using (P) values. The influence of the genes is not the same, the statistical importance of the sixth gene (Gene 6) is higher than the other genes and the bias term.

7. Application of MGGP-ANN technique in prediction of compliance in concrete

An improvement of the optimal MGGP model can be done by combine ANN with MGGP technique. The neural network is used to predict the errors (E_{ANN}) in optimal MGGP model. Fig. 6 presents the hybrid system of MGGP and ANN techniques which are working in parallel together. The inputs of the ANN are the same inputs used to develop the optimal MGGP model. The output of the ANN is error

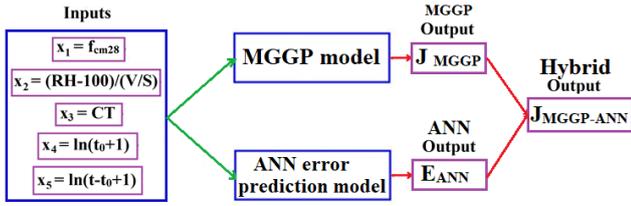


Fig. 6 Hybrid MGGP-ANN model

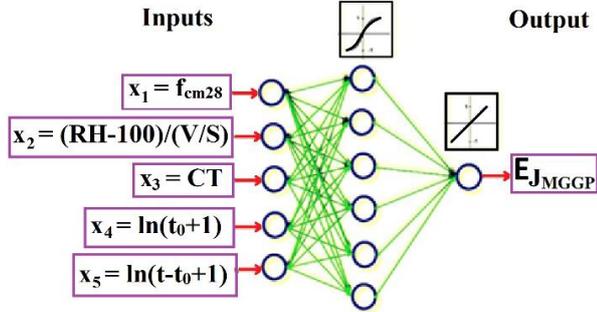


Fig. 7 Structure of the optimal neural network

(E_{ANN}), i.e., the difference between the predicted compliance by optimal MGGP model and the measured compliance by the experimental tests. The Target of neural network model is calculated the predicted value (E_{ANN}) of the Error (E) demonstrated in Eq. (10).

$$E = J_{MGGP} - J_{measured} \quad (10)$$

The inputs and output should be scaled to increase the ability and performance of neural network training. By default, the multilayer network creation functions have default processing function in MATLAB. This processing function map minimum and maximum values for inputs and output to $[-1 \ 1]$, (MathWorks 2015). The number of hidden layers and the number of neurons in each hidden layer depend on many factors, including the number of data points in training set, the performance goal and the complication of the problem. A network that contains one hidden layer can approximate any continuous function if suitable connection weight are used, (Shahin *et al.* 2003). So, a neural network consist of one hidden layer is used in this study. A number of trials is accomplished using the default parameters of the MATLAB software with one hidden layer and trials of 1, 2, 3...11 neurons in hidden layer. The number of neurons in hidden layer equal to " $2*n+1$ " is enough to represent any continuous function for a network with " n " inputs. The upper limit of neurons used in hidden layer is 11.

The Fig. 7 shows the architecture of optimal neural network including inputs, number of hidden units, type of transfer function in hidden layer and type of transfer function in output layer.

Table 3 show the results of RMSE and R for training (T), testing (S) and validation (V) for the best neural network that working in parallel with the optimal MGGP model.

The training, testing and validation progress of optimal neural network is shown in Fig. 8. The mean square error of training, testing and validation sets have similar

Table 3 Performance of optimal MGGP with optimal ANN

No. of neurons in hidden layer	Transfer function in hidden layer	RMSE%			R%		
		T	S	V	T	S	V
6	Tansig	23.3	23.2	24.2	93.6	93.7	93.2

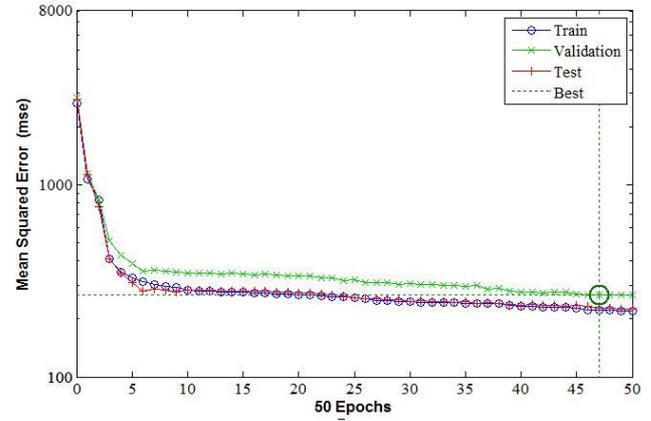


Fig. 8 Training, testing and validation progress of optimal neural network

Table 4 Performance of optimal MGGP-ANN & MGGP models

Model code	ω_{BP} %	RMSE%	R%
MGGP-800-50-7-6-10	29.94	27.45	91.05
MGGP-800-50-7-6-10-ANN-Tansig-6	26.57	23.43	93.57

characteristic and it doesn't appear that any important over fitting has occurred.

The result from optimal MGGP-ANN model can be calculated by Eq. (11).

$$J_{MGGP-ANN} = J_{MGGP} - E_{ANN} \quad (11)$$

Table 4 shows the performance of optimal MGGP-ANN model and optimal MGGP model. The (RMSE), (R) and (ω_{BP}) values are calculated for all data points. In this table we can see the minimum (ω_{BP}) & (RMSE) and maximum (R) is for optimal MGGP-ANN model.

Fig. 9 shows a comparison between the predicted compliance and experimental compliance using optimal MGGP and MGGP-ANN model. It can be noticed that the correlation coefficients (R) of testing data points is better than training data points which means good generalization.

8. Parametric study

The objective of parametric study is to confirm the generalization ability of optimal MGGP-ANN model and to quantify the effect of each input when all the others are fixed with mean values.

8.1 Influence of f_{cm28}

Fig. 10 shows the relationship between the average

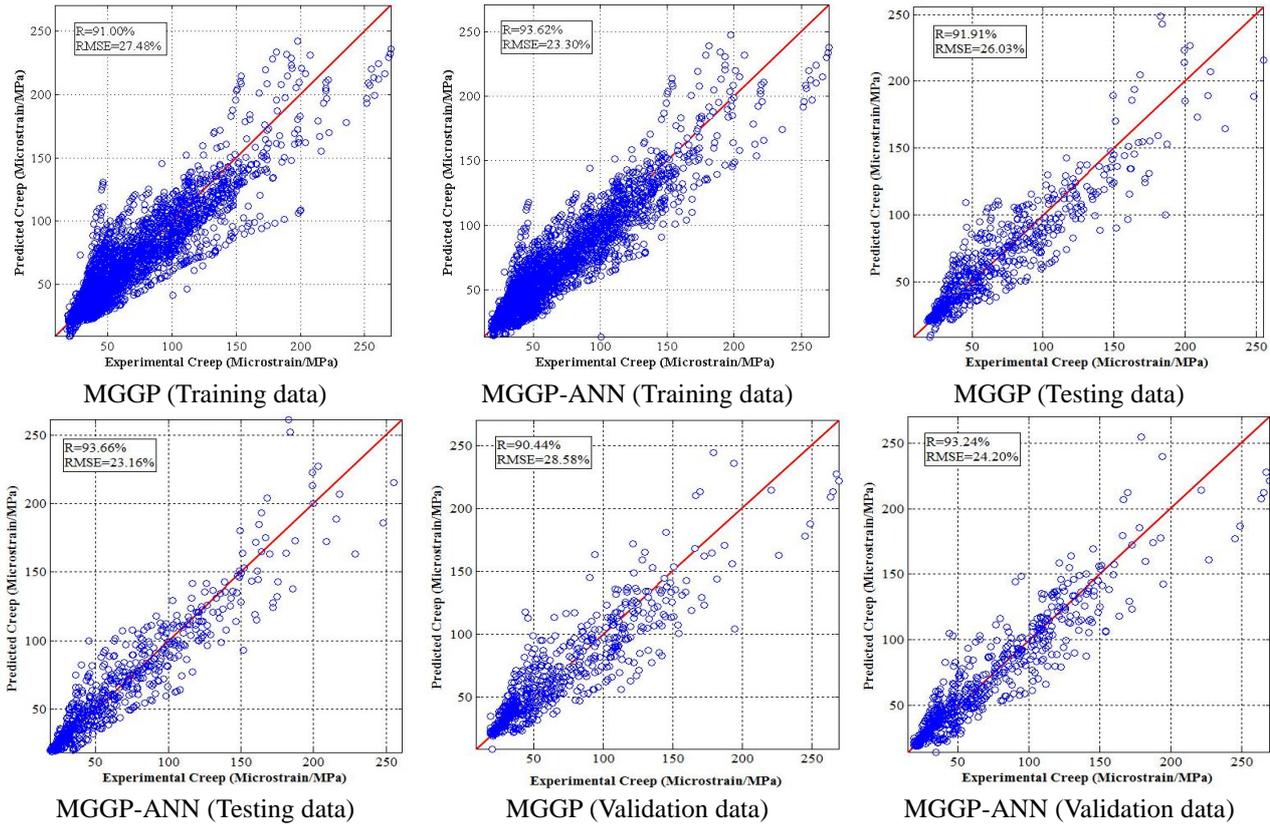


Fig. 9 Comparison of predicted and measured compliance using the optimal MGGP and MGGP-ANN models

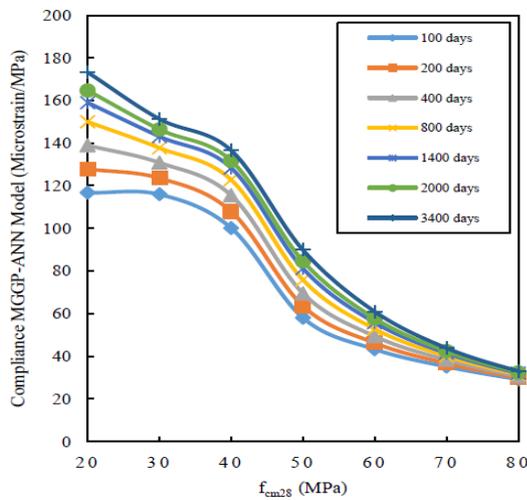


Fig. 10 f_{cm28} influence on the compliance of concrete at various ages for optimal MGGP-ANN model

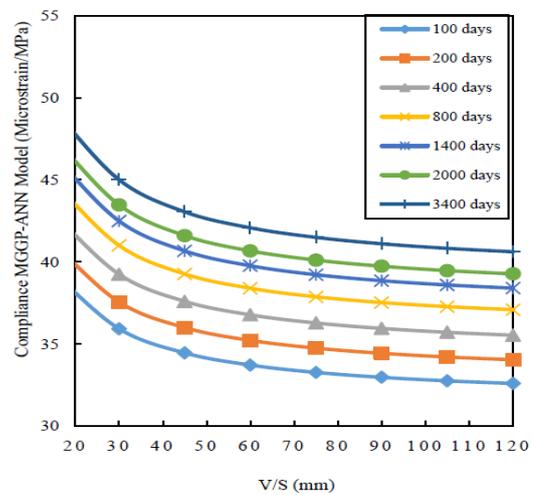


Fig. 11 Volume to surface ratio (V/S) influence on the creep of concrete at various ages for optimal MGGP-ANN model

compressive strength f_{cm28} and the compliance in concrete at various ages. It can be noticed that the compliance decreases as the f_{cm28} increases (ranging between 20 and 80 MPa). One of the important factors on the f_{cm28} and the quality of concrete is W/C ratio. When the W/C ratio is high, in this situation more of water will evaporates, the f_{cm28} will decrease and the value of creep will increase.

8.2 Influence of V/S

The sample size of concrete affects the value of creep.

The creep increases as the cross section decreases. The moisture equilibrium will take hundreds of days in big depth of concrete. In the contrary, the concrete balance will accrue after a few months in small depth of concrete. It depends on the long of the path that water molecules pursue to move from the inside the element to outside (Karthikeyan *et al.* 2008).

The relationship between the volume to surface ratio V/S and the compliance in concrete at various ages are demonstrated in Fig. 11. It can be noticed that the compliance decreases as the V/S increases (ranging

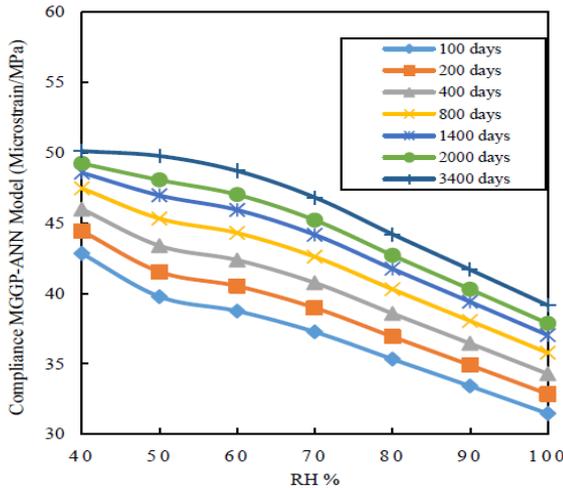


Fig. 12 Relative humidity (RH) influence on the creep of concrete at various ages for optimal MGGP-ANN model

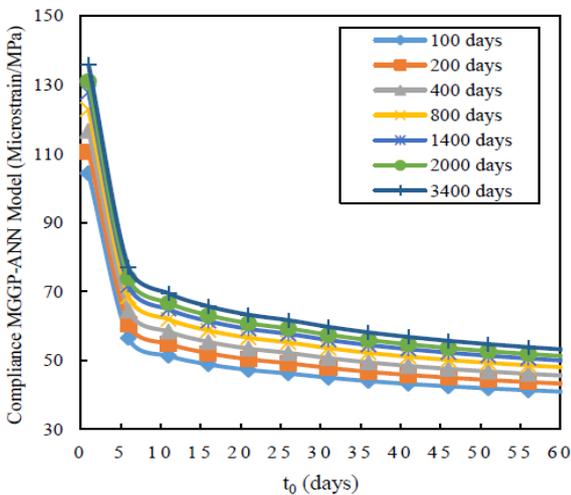


Fig. 13 t_0 influence on the creep of concrete at various ages for optimal MGGP-ANN model

between 20 and 120 mm).

8.3 Influence of RH

One of the most significant parameters is relative humidity that affecting the creep of the concrete. Fig. 12 presents the relationship between the relative humidity RH and the compliance in concrete at various ages. It can be noticed that the compliance decreases as the RH increases (ranging between 40% and 100%). The relative humidity of the ambience considers an external parameter. The value of final creep in cement paste and consequently creep in concrete is influence by relative humidity. The higher the relative humidity, the lower is water loss and therefore the lower creep.

8.4 Influence of t_0

Fig. 13 shows the relationship between the age at start of loading t_0 and the compliance in concrete in various ages. It can be noticed that the compliance decreases as the t_0

Table 5 Root mean square error, correlation coefficient and B3 coefficient of variation for compliance

Model	ACI 209	B3	CEB MC90-99	GL 2000	MGGP	MGGP-ANN
RMSE%	34.47	35.66	39.39	31.05	33.36	27.63
R%	61.32	66.39	60.07	70.97	67.56	78.61
ω_{BP} %	56.60	40.91	48.04	51.47	42.49	37.82

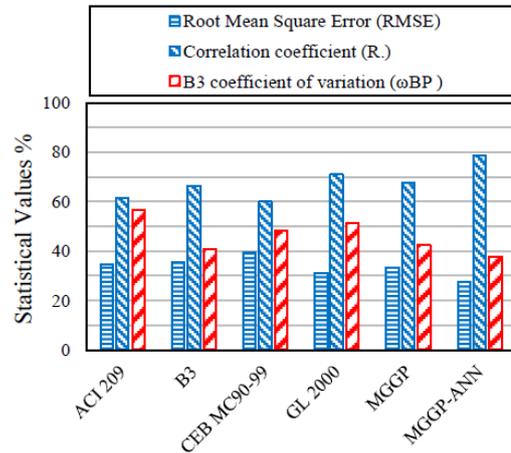


Fig. 14 Root mean square error, correlation coefficient and B3 coefficient of variation for compliance

increases (ranging between 1 day and 60 days). The compliance decreased grossly within the first 7 days. Then the compliance gradually decreases as the t_0 increases.

9. Evaluation of creep prediction models

The optimal MGGP and MGGP-ANN creep prediction models are evaluated for their accuracy. The ACI 209 Model (Branson and Christiason 1971) the B3 Model (Bažant and Baweja 2000) the CEB MC90-99 Model (CEB 1999, Abed, El-Shafie *et al.* 2010) the GL 2000 Model (Gardner 2004, Abed and Osman 2013) the MGGP model and the MGGP-ANN model are used to compare their predicted results against NU-ITI database by using six statistical approaches as follows:

1. Root mean square error (RMSE).
2. The correlation coefficient (R).
3. B3 coefficient of variation (ω_{BP}) : A coefficient of variation ω_{BP} that developed by Bažant and Panula (1978). The less accurate the prediction model, the higher the value of ω_{BP} .
4. Residual method: The compliance residuals were calculated as the different between the predicted and experimental values for all six models, and then decide how much different those predicted values are far from measured values.
5. Average of residuals: the average mean values for every model's residuals.
6. Standard deviation of residuals: the standard deviation values for every model's residuals.

The root mean square error, correlation coefficient and B3 coefficient of variation for compliance for all six models

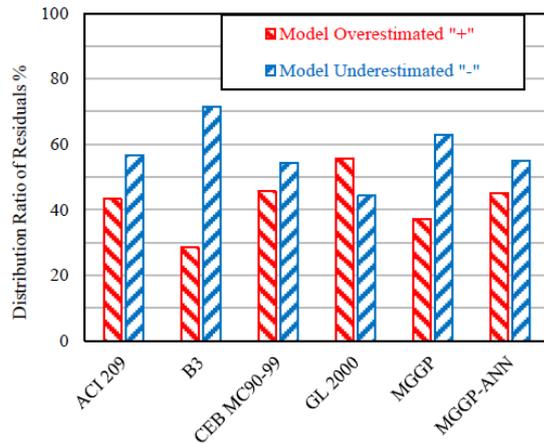


Fig. 15 Distribution of residual for compliance for all six models

Table 6 Distribution of residuals in different ranges for compliance in time range 0 to 1000 days

Residual Range (Microstrain MPa)	Number of Residual Points (percentage)					
	0 to 1000 days					
	ACI 209	B3	CEB MC90-99	GL 2000	MGGP	MGGP-ANN
0 to +33	270 (30.47%)	213 (24.04%)	345 (38.94%)	410 (46.28%)	289 (32.62%)	368 (41.53%)
0 to -33	442 (49.89%)	497 (56.09%)	340 (38.37%)	348 (39.28%)	441 (49.77%)	410 (46.28%)
Over +33	116 (13.09%)	40 (4.51%)	63 (7.11%)	87 (9.82%)	43 (4.85%)	34 (3.84%)
Over -33	58 (6.55%)	136 (15.35%)	138 (15.58%)	41 (4.63%)	113 (12.75%)	74 (8.35%)
0 to ±33	712 (80.36%)	710 (80.14%)	685 (77.31%)	758 (85.55%)	730 (82.39%)	778 (87.81%)
Over ±33	174 (19.64%)	176 (19.86%)	201 (22.69%)	128 (14.45%)	156 (17.61%)	108 (12.19%)

Note: The values in the brackets imply to percentage of residuals in that specific range to the overall number of residual points in its time range.

were calculated and summarized in Table 5 and illustrated in Fig. 14.

The compliance residuals were calculated as the different between the predicted and experimental values for all six models. The positive values of residuals imply that compliance values are overestimated by corresponding model. The negative values of residuals imply that compliance values are underestimated by corresponding model. The compliance residuals in different ranges were calculated as percentages as illustrated in Fig. 15.

The percentages and numbers of residuals distribution were summarized for the time interval [0-1000] days in Table 6

Table 7 summarizes the average of compliance residuals and also the mean averages is calculated for six time intervals.

Table 8 summarizes the standard deviation of compliance residuals and also the mean standard deviations

Table 7 The average of compliance residuals for six models

Time ranges (days)	The average of residuals for models (Microstrain/MPa)					
	ACI 209	B3	CEB	GL 2000	MGGP	MGGP-ANN
0-10	-0.78	-4.67	-1.81	2.47	-5.16	-3.08
11-100	-0.69	-13.49	-5.61	1.85	-6.39	-2.68
101-365	-0.42	-12.5	-4.45	3.37	-8.07	-3.61
366-730	1.25	-13.1	-10.55	5.58	-8.8	-5.44
731-1095	-7.89	-21.24	-22.12	-7.01	-14.29	-9.86
Above 1095	-40.53	-27.93	-34.47	-8.39	-45.83	-42.00
Mean Average	-8.18	-15.49	-13.17	-0.35	-14.76	-11.11

Table 8 The standard deviation of compliance residuals for six models

Time ranges (days)	The standard deviation of residuals for models (Microstrain/MPa)					
	ACI 209	B3	CEB	GL 2000	MGGP	MGGP-ANN
0-10	15.1	16.95	16.67	15.05	14.94	12.76
11-100	22.94	22.67	26.02	20.92	21.65	18.12
101-365	30.62	28.45	36.3	26.75	28.40	23.73
366-730	40.84	37.28	44.24	36.12	37.27	31.11
731-1095	33.43	32.89	34.94	30.45	30.31	22.75
Above 1095	24.55	16.23	16.99	13.77	27.81	26.05
Mean standard deviation	27.91	25.75	29.19	23.84	26.73	22.42

Table 9 Summary of six compliance indicators

Model	ACI 209	B3	CEB MC90-99	GL2000	MG GP	MGGP-ANN
RMSE%	34.47	35.66	39.39	31.05	33.36	27.63
R%	61.32	66.39	60.07	70.97	67.56	78.61
ω_{BP} %	56.6	40.91	48.04	51.47	42.49	37.82
Residuals Over ±33%	19.64	19.86	22.69	14.45	17.61	12.19
Mean Average of residuals (Microstrain/MPa)	8.18	15.49	13.17	0.35	14.76	11.11
Mean Standard deviation of residuals (Microstrain/MPa)	27.91	25.75	29.19	23.84	26.73	22.42

is calculated for six time intervals.

Table 9 summarizes the results of different statistical indicators for evaluations of compliance residuals and compliance. This table summarizes the root mean square error, the correlation coefficient and the B3 coefficient of variation of compliance. In addition, this table demonstrated the absolute average of compliance residuals for time interval [0-3500] days, the standard deviation of compliance residuals in time interval [0-3500] days and the percentage of residuals over ±33 Microstrain/MPa.

The RMSE, ω_{BP} , average of residuals, standard deviation of residuals and residuals over ±33 Microstrain/MPa decrease as the model performance and fitness increase. While, the R coefficient increases as the model performance and fitness increase.

It can be observed from Table 9 that the MGGP-ANN model had a highest score and best performance followed

by the GL 2000 model, the MGGP model, the B3 model, ACI 209 model and at the last CEB-MC 90-99.

10. Conclusions

Based on the results and investigations obtained in this study, the following conclusions and characteristics were summarized:

- The MGGP model is developed and a practical equation is extracted to predict the compliance in concrete with very good degree of generalization and accuracy within the given range of training data.
- An enhancement of MGGP model is accomplished by hybridized with ANN by predicting and reducing the error.
- In general, increasing the population, the maximum depth of tree, the maximum number of genes and the number of generation in MGGP technique causes an increase in performance and time consuming. Bearing in mind this increasing -for mentioned parameters- will not necessarily guide to better performance and good results because of increasing in level of noise and due to random nature of MGGP in initializing population and performing crossover and mutation operations.
- In the ANN technique, the increasing of neurons in hidden layer usually causes an increase in performance and time consuming. However, after certain number of neurons the neural network lose its ability to generalize.
- The results of parametric study of the MGGP-ANN model are compatible with the literature and properties of creep. The compliance is increased as the mean compressive strength of concrete, the volume to surface ratio, the relative humidity and the age at loading are decreased. Contrariwise, the compliance is increased as the age at creep measurement is increased.
- The evaluation procedure and rating of models accomplished by using six statistical indicators. According to these indicators, the MGGP-ANN model has a highest score and best performance followed by the GL 2000 model, the MGGP model, the B3 model, ACI 209 model and CEB-MC 90-99.

References

- Abdollahzadeh, G., Jahani, E. and Kashir, Z. (2016), "Predicting of compressive strength of recycled aggregate concrete by genetic programming", *Comput. Concrete*, **18**(2), 155-163.
- Abed, M. and Osman, S. (2013), "Dynamic versus static artificial neural network model for masonry creep deformation", *Proceedings of the Institution of Civil Engineers Structures and Buildings*, 355-366.
- Abed, M., El-Shafie, A. and Osman, S. (2010), "Creep predicting model in masonry structure utilizing dynamic neural network", *J. Comput. Sci.*, **6**(5), 597-605.
- ACI-Committee209 (2008), Guide for Modeling and Calculating Shrinkage and Creep in Hardened Concrete (ACI 209.2R-08), American Concrete Institute, Farmington Hills, MI.
- Ao, S., Castillo, O. and Hang, X. (2011), *Intelligent Control and Computer Engineering*, Springer Netherlands.
- Bal, L. and Buyle-bodin, F. (2014), "Artificial neural network for predicting creep of concrete", *Neur. Comput. Appl.*, **25**(6), 1359-1367.
- Bazant, Z.P. (1982), *Mathematical Models for Creep and Shrinkage of Concrete, Creep and Shrinkage in Concrete Structures*, Wittmann, John Wiley & Sons Ltd.
- Bazant, Z.P. and Baweja, S. (2000), "Creep and shrinkage prediction model for analysis and design of concrete structures: model B3", *The Adam Neville Symposium: Creep and Shrinkage-Structural Design Effects*, SP-194, Ed. A. Al-Manaseer, American Concrete Institute, Farmington Hills, MI.
- Bazant, Z.P. and Li, G.H. (2008), "Comprehensive database on concrete creep and shrinkage", *Ac. Mater. J.*, **105**(6), 635-637.
- Bazant, Z.P. and Panula, L. (1978), "Practical prediction of time dependent deformations of concrete", *Mater. Struct.*, **11**(5), 307-316.
- Branson, D.E. and Christason, M.L. (1971), "Time-dependent concrete properties related to design-strength and elastic properties, creep, and shrinkage designing for effects of creep, shrinkage, and temperature in concrete structures", SP-27, American Concrete Institute, 257-277.
- CEB (1999), Structural Concrete-Textbook on Behaviour, Design and Performance, Updated Knowledge of the CEB/FIP Model Code 1990, fib Bulletin2, Federation Internationale du Beton, Lausanne, Switzerland, **2**, 37-52.
- Cladera, A., Perez-Ordóñez, J.L. and Martínez-Abella, F. (2014), "Shear strength of RC beams. Precision, accuracy, safety and simplicity using genetic programming", *Comput. Concrete*, **14**(4), 479-501.
- Fanourakis, G.C. and Ballim, Y. (2003), "Predicting creep deformation of concrete: a comparison of results from different investigations", *Proceedings of the 11th FIG Symposium on Deformation Measurements*, Santorini, Greece.
- Fulcher, J. and Lakhmi, C.J. (2008), *Computational Intelligence: A Compendium*, Springer-Verlag Berlin Heidelberg.
- Gandomi, A.H., Sajedi, S., Kiani, B. and Huang, Q.D. (2016), "Genetic programming for experimental big data mining: A case study on concrete creep formulation", *Auto. Constr.*, **70**, 89-97.
- Gardner, N.J. (2004), "Comparison of prediction provisions for drying shrinkage and creep of normal-strength concretes", *Can. J. Civil Eng.*, **31**(5), 767-775.
- Golafshani, E.M., Rahai, A. and Kebria, S.S.H. (2014), "Prediction of the bond strength of ribbed steel bars in concrete based on genetic programming", *Comput. Concrete*, **14**(3), 327-345.
- Hakin, S. (1999), *NEURAL NETWORKS A Comprehensive Foundation*, Prentice Hall International, Inc.
- Holland, J. (1992), *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control and Artificial Intelligence*, MIT Press, Cambridge, MA.
- Karthikeyan, J., Upadhyay, A. and Bhandari, N.M. (2008), "Artificial neural network for prediction creep and shrinkage of high performance concrete", *J. Adv. Concrete Technol.*, **6**(1), 135-142.
- Klovanych, S. (2015), "Creep of concrete at variable stresses and heating", *Comput. Concrete*, **16**(6), 897-907.
- Lozano-Galant, J.A. and Turmo, J. (2014), "Creep and shrinkage effects in service stresses of concrete cable-stayed bridges", *Comput. Concrete*, **13**(4), 483-499.
- MathWorks (2015), "Matlab-help", <http://www.mathworks.com/help/nnet/ug/choose-neuralnetwork-input-output-processing-functions.html>.
- Muduli, P.K. and Das, S.K. (2013), "CPT-based seismic liquefaction potential evaluation using multi-gene genetic programming approach", *Indi. Geotech. J.*, **44**(1), 86-93.
- Negnevitsky, M. (2005), *Artificial Intelligence A Guide to Intelligent Systems*, Addison-Wesley.

- Searson, D. (2014), "GPTIPS symbolic data mining platform for MATLAB", <https://sites.google.com/site/gptips4matlab/7>.
- Searson, D.P. (2009), *GPTIPS: Genetic Programming & Symbolic Regression for MATLAB*, User Guide.
- Searson, D.P., Leahy, D.E. and Willis, M.J. (2010), "GPTIPS: An open source genetic programming toolbox for multigene symbolic regression", *Proceedings of the International Multi Conference of Engineers and Computer Scientists*, **1**, Citeseer.
- Shahin, M.A., Jaska, M.B. and Maier, H.R. (2003), "Application of artificial neural networks in foundation engineering", Australian Geomechanics.
- Shukla, A., Tiwari, R. and Kala, R. (2010), *Towards Hybrid and Adaptive Computing*, Springer-Verlag Berlin Heidelberg.
- Taha, M.M.R., Noureldin, A., Ei-Sheimy, N. and Shrive, N.G. (2003), "Artificial neural networks for predicting creep with an example application to structural masonry", *Can. J. Civil Eng.*, **30**(3), 523-532.
- Taylor, B.J. (2006), *Methods and Procedures for the Verification and Validation of Artificial Neural Networks*, Springer US.

HK