Computers and Concrete, Vol. 6, No. 3 (2009) 203-223
DOI: http://dx.doi.org/10.12989/cac.2009.6.3.203

Modeling strength of high-performance concrete using genetic operation trees with pruning techniques

Chien-Hua Peng†

Department of Civil Engineering and Engineering Informatics, Chung Hua University, Taiwan, R.O.C

I-Cheng Yeh‡

Department of Information Management, Chung Hua University, Taiwan, R.O.C

Li-Chuan Lien‡†

Department of Construction Engineering, National Taiwan University of Science and Technology, Taiwan, R.O.C

(Received June 11, 2008, Accepted May 12, 2009)

Abstract. Regression analysis (RA) can establish an explicit formula to predict the strength of High-Performance Concrete (HPC); however, the accuracy of the formula is poor. Back-Propagation Networks (BPNs) can establish a highly accurate model to predict the strength of HPC, but cannot generate an explicit formula. Genetic Operation Trees (GOTs) can establish an explicit formula to predict the strength of HPC that achieves a level of accuracy in between the two aforementioned approaches. Although GOT can produce an explicit formula but the formula is often too complicated so that unable to explain the substantial meaning of the formula. This study developed a Backward Pruning Technique (BPT) to simplify the complexity of GOT formula by replacing each variable of the tip node of operation tree with the median of the variable in the training dataset belonging to the node, and then pruning the node with the most accurate test dataset. Such pruning reduces formula complexity while maintaining the accuracy. 404 experimental datasets were used to compare accuracy and complexity of three model building techniques, RA, BPN and GOT. Results show that the pruned GOT can generate simple and accurate formula for predicting the strength of HPC.

Keywords: back-propagation networks; genetic operation trees; high-performance concrete; backward pruning technique; median constant.

1. Introduction

In addition to conventional concrete ingredients (water, fine and coarse aggregates), High-Performance Concrete (HPC) (Huang 1999) requires the addition of supplementary cementitious materials (e.g., fly ash, blast furnace slag) and chemical admixtures (e.g., superplasticizer). Besides

† Research Assistant
‡ Professor
‡† Research Assistant, Corresponding Author, E-mail: lclien@gmail.com

improving strength and workability, such additives improve the durability, safety, economy and environmental demand. Traditionally, concrete strength models are established using statistical methods such as regression analysis (RA). Although RA can build an explicit formula, its accuracy is relatively low. Therefore, it is not feasible to apply such formula to establish a complex non-linear strength model for HPC.

There are many research studies have been done on the application of artificial neural networks (ANN) (Yeh 2004) in the field of material science. Although many studies (Yeh 1998, 1999, Kim *et al.* 2004, Chen 2003, Chen *et al.* 2004, Hamid-Zadeh *et al.* 2007, Ahmet *et al.* 2006, Lien *et al.* 2006) have proposed complex non-linear strength models which can accurately predict the strength behavior of HPC with a high level of accuracy, these "black boxes" models are unable to provide explicit formula to explain the substance of strength models.

Some researchers have employed genetic operation trees (GOTs) that comprise genetic algorithms (GA) and operation trees (OTs) in order to build strength models that can accurately predict HPC strength behavior and explain the substance of strength models (Chen 2003, Lien *et al.* 2006). Operation tree is a tree structure that expresses a mathematical formula. Optimizing the operation tree can produce a self-organized regression formula. However, this optimization problem is a discrete optimization problem, which mathematical programming cannot solve. GA can solve discrete optimization problem, representing one paradigm of evolution computation that is based on natural evolution and derived from the ideas of "the survival of the fittest" (Davis 1991), such as inheritance, selection, crossover, and mutation. GA has some advantages, such as global optimization, non-linear ability, flexibility, and parallelism (Goldberg 1989).

In general, the accuracy of GOT generated models are lower than those produced by artificial neural networks, but more accurate than those produced by RA (Chen 2003, Chen *et al.* 2004, Lien *et al.* 2006). Although GOT can produce explicit strength formulas but these formulas are often too complicated so that unable to explain the substantial meaning of the strength model. To make these complicated formulas (operation trees) simpler, understandable, and more meaningful, this study developed a Backward Pruning Technique (BPT) to simplify the complexity of the operation tree produced by GOT. This technique tries to replace each variable of the tip node of the operation tree with the median in the training dataset belonging to the node and then pruning the node with the most accurate test dataset. Such pruning reduces formula complexity while maintaining accuracy. A large number of experimental datasets were used to compare accuracy and complexity of the three model building techniques (RA, BPN and GOT) and evaluate whether GOT can produce simpler and understandable but accurate operation trees (formulas) to predict HPC strength.

2. Operation tree, genetic algorithm and pruning technique

2.1. Operation tree

Operation tree is a tree structure that expresses a mathematical formula. Fig. 1 shows an operation tree model, where the root X_1 denotes a mathematical operation (+, –, ×, ÷, ln or exp), each branch X_2~X_7 denotes a variable, constant, or mathematical operation, and each leaf X_8~X_{15} denotes a variable or constant (Lien *et al.* 2006). When X_1~X_{15} are set with specific mathematical operations, variables, or constants, the operation tree can express a specific mathematical formula.

Fig. 2 shows an example of operation tree model. The model used mathematical operations +, –,

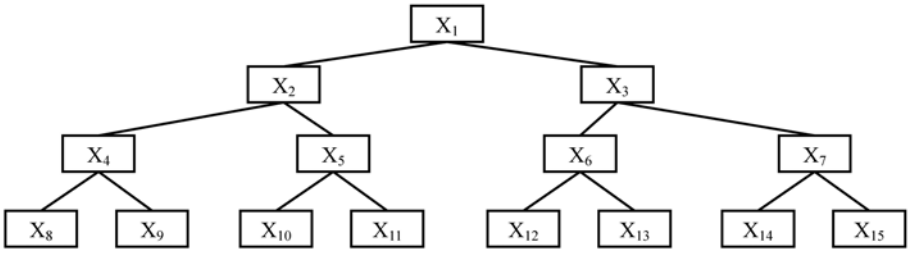

Fig. 1 Diagram of operation tree model

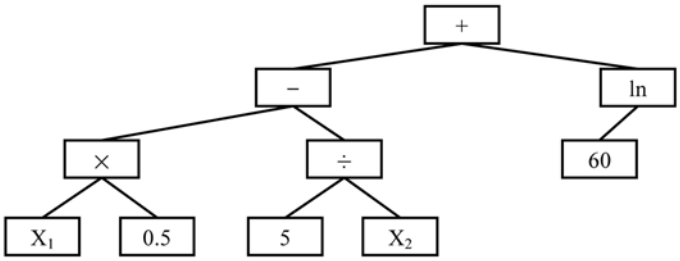

Fig. 2 Diagram of a specific example of operation tree model

×, ÷, and ln, variables X_1 and X_2, and constants 0.5, 5, and 60. The operation tree can express the following formula.

$$Y = \left(X_1 \times 0.5 - \frac{5}{X_2}\right) + \ln(60) \tag{1}$$

Operation tree can be employed to build a regression model by setting up appropriate mathematical operations, variables and constants in the root, branches, and leaves of the operation tree. When the tree-style structure is set up to represent a specific mathematical formula, it can produce predicted output value for each data by substituting their inputs to the variables on the branches or leave of the tree-style structure. The root mean squared error (RMSE) between predicted and actual output values can be used to evaluate operation tree performance. The operation tree with the minimum RMSE is the one that fit the dataset best.

Using operation tree can overcome the disadvantages of conventional regression analysis, which requires the regression formula structure to be predetermined and only allow the regression coefficients of the predetermined structure to be adjusted. On the other hand, operation tree is a tree-style data structure which expresses a flexible mathematical formula, and optimizing the structure to fit data is a discrete optimization problem. Therefore, conventional mathematical programming cannot optimize the operation tree. This study adopted genetic algorithm with discrete optimization ability to optimize the operation trees to fit the data best.

2.2. Genetic algorithm (GA)

Genetic Algorithm (GA) is an optimization method that employs a unique searching algorithm that can jump from local optimum and close to the global optimum. This method was first developed in 1975 by Holland (Davis 1991). The GA concept was derived from Darwin's Theory "natural selection, survival of the fittest." In fact, any organism which is capable of reproducing

itself on an ongoing basis will survive as a species, not just the "fittest" ones. A more accurate characterization of evolution would be "survival of the fit enough". Under the mechanism of competition for existence, initial generation individuals produce future generation individuals through crossover and mutation. By combining genes, future generation individuals have similar but different characteristics from their former generation individuals. Moreover, gene mutation also produces some variation. The new generation also produces the next generation under the mechanism of competition for existence. The mechanism reproduces one generation to next generation, and species constantly evolving to adapt to a changing environment.

GA can produce an optimal solution by treating the "optimization problem" as an "evolution problem", the "solution" as an "individual", and the "optimization process" as an "evolutionary process". The basic concepts of GA are described as follow (Lien *et al.* 2006):

- Encoding and decoding: In GA, a solution is composed of a large number of genes. When a solution is represented as a set of genes, it is called encoding. On the other hand, when a set of genes is expressed to a solution, it is called decoding. In GOT, a solution is a mathematical formula represented as an operation tree, which consists of a set of genes. Therefore, when a mathematical formula is represented as an operation tree, it can be considered as a form of encoding, and when an operation tree is expressed to a mathematical formula, it can be considered as a form of decoding.
- Fitness function and fitness: Each organism has a different capacity to adapt to its environment. Only the fittest survive. Those who cannot will be eliminated. A fitness function is function designed to mimic the role of the natural environment (optimization problem) to measure the level of adaptation of an individual (solution). After decoding the individual's genes into a solution, the fitness function can evaluate the fitness of the individual. The purpose of such an evolution is to maximize the fitness function.
- Selection: The GA selection mechanism simulates the "survival of the fittest" phenomenon in nature. The higher the individual fitness, the higher the probability of survival. In other words, the lower the individual fitness, the lower the probability of survival. If the fitness of certain individuals exceeds that of others, they are more likely to produce offspring successfully and see their genes increasingly represented in the population mainstream.
- Crossover: Crossover is an operation that exchanges and combines the genes of two individuals to form two offspring with similar but different structures and features.
- Mutation: Mutation is an operation that arbitrarily alters one or more genes of an individual to increase population variability.
- Stop criterion: The process of GA is a loop. Therefore, a rule is required to decide when it should be stopped. The four kinds of common stop criteria include: (1) a better solution cannot be found in a predetermined number of generations (e.g., 30 generations). (2) The number of evolution generation has reached the predetermined number (e.g., 200 generations). (3) The fitness value has satisfied a predetermined target value. (4) Population variability has satisfied a predetermined standard (e.g., the difference of fitness between the best and the worst ethnic is lower than target value).
- Elitist strategy: In the GA selection mechanism, the higher the individual fitness, the higher the probability of survival. However, it does not guarantee that the individual with the highest fitness would survive. In other words, the best of the next generation will not necessarily be better than the best of the current generation. Under elitist strategy, the best individual is retained in the next generation even though it is not selected via the selection mechanism. This strategy can guarantee that the best individual of final generation is the best individual of all individuals produced in the whole evolution process.

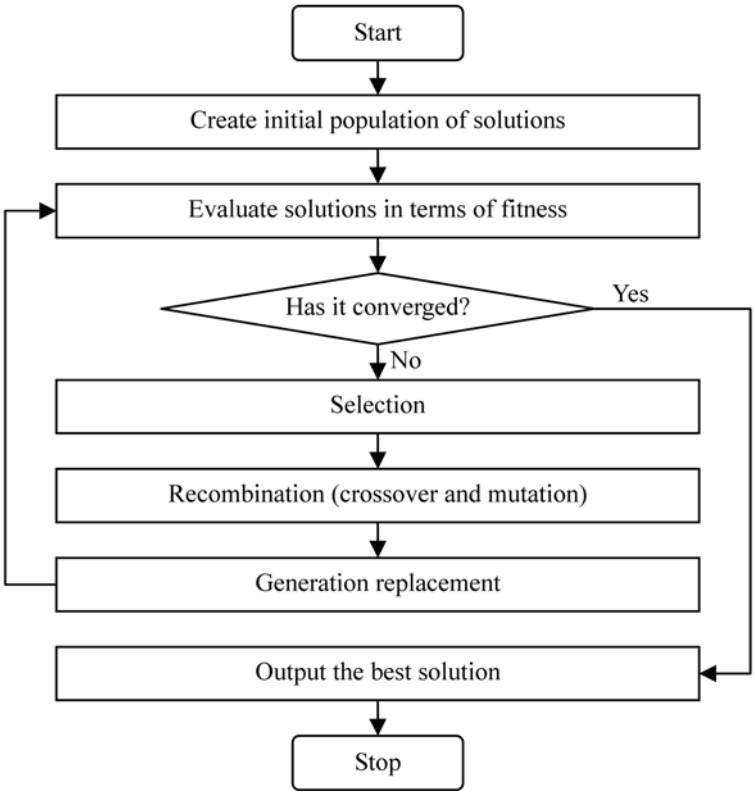

Fig. 3 Flowchart of genetic algorithm (GA)

The comprehensive procedure of GA is shown in Fig. 3.

2.3. Pruning operation tree

Although the GOT can produce an explicit formula, the formula is often so complicated that it is unable to explain the substantial meaning of the formula. Therefore, this study developed a Backward Pruning Technique (BPT) to simplify the complexity of the formula produced by GOT. The central idea of this technique is to try to replace each variable of the tip node of operation tree with the median of the variable in the training dataset belonging to the node, and then pruning the node with the highest test data accuracy. This pruning can decrease the complexity of the formula while maintaining a similar level of accuracy.

The state of each node is binary, either preserved as a variable or replaced with a median constant. The key problem of the technique is how to determine the state of each node. The problem can be considered as a search problem, for which this study proposed two kinds of search methods, as follows:

(1) Exhaustive search: When an operation tree has N nodes, and each node is binary (preserved or replaced), then there are 2^N alternatives for the tree pruning. When N is small (e.g., below ten), the exhaustive search can be carried on.

(2) Best-first search (backward pruning method): When N is not small, the exhaustive search can

not be carried on. Under this situation, the best-first search, a process similar to stepwise regression, can be conducted. First, each variable at the tip node of operation tree should be replaced with the median of the variable in the training dataset that belongs to the node. Then prune the node whose accuracy of test dataset is the highest. This process is repeated until no improvement can be reached in all the tip nodes of the operation tree. Detailed steps are described as follows:

Step 1: Try to replace each variable of the tip node of operation tree with the median of the variable in the training dataset belonging to the node; then prune the node whose accuracy of test dataset is the highest.

Step 2: Repeat step 1 until no improvement can be reached by pruning another tip node.

Step 3: Output the pruned operation tree.

3. Modeling strength of high-performance concrete

3.1. Experimental data

There are eleven input variables in the HPC strength model include cement (C), fly ash (FL), slag (SL), water (W), superplasticizer (SP), coarse aggregate (CA), fine aggregate (FA), water-cement ratio (W/C), water-binder ratio (W/B), water-solid ratio (W/S) and total aggregate binder ratio (TA/B). The output of the model is 28-day compressive strength (f'_c, MPa). This study collected 404 experimental strength data, 300 data were randomly selected as the training set, and the remaining 104 data as the testing set (Lien *et al.* 2006). The training set was employed to build the model and the testing set was employed to evaluate model generalizations. Table 1 presents some descriptive statistics of the strength dataset.

Table 1 Descriptive statistics of the data set

Variable	Unit	Minimum	Maximum	Mean	Standard deviation
cement (C)	kg/m^3	71.00	896.00	293.35	129.80
fly ash (FL)	kg/m^3	0.00	200.10	48.30	63.07
slag (SL)	kg/m^3	0.00	359.40	76.45	88.52
water (W)	kg/m^3	118.00	314.00	180.98	25.04
superplasticizer (SP)	kg/m^3	0.00	32.20	5.49	5.66
coarse aggregate (CA)	kg/m^3	595.00	1820.00	963.67	130.07
fine aggregate (FA)	kg/m^3	486.00	1300.00	779.61	99.51
water cement ratio (W/C)	ratio	0.24	2.73	0.79	0.41
water binder ratio (W/B)	ratio	0.24	0.90	0.48	0.14
water solid ratio (W/S)	ratio	0.04	0.13	0.09	0.01
total aggregate binder ratio (TA/B)	ratio	2.18	9.85	4.46	1.29
28-day compressive strength f'_c	MPa	8.5	122.0	42.9	22.5

Note: W/C=(W+SP)/C; W/B=(W+SP)/(C+FL+SL); W/S=(W+SP)/(C+FL+SL+CA+FA); TA/B=(CA+FA)/(C+FL+SL)

3.2. Gene encoding of operation tree

This study adopted operation tree to express a regression formula and employed genetic algorithms to optimize the tree to produce a self-organized formula. In this study, a five-layered operation tree was adopted, as shown in Fig. 4. The gene encoding rule of mathematical operations and variables and constants are listed in Table 2 and Table 3, respectively. The gene encoding rule was designed to adhere to the following rules:

- Gene X_1 on the first (top) layer must be mathematical operations. Therefore, the gene must be between integer 1 and integer 6 (see Table 2).
- Genes X_2 to X_{15} on the second, third and fourth layers may be mathematical operations, variables, or constants. Therefore, these genes may be integers between 1 and 18. When the gene encoding is 18, it represents a constant K, and a constant between -100 and 100 would be assigned to the gene.
- The fifth layer (X_{16} to X_{31}) must be a variable or constant. Therefore, genes must be integers between 7 and 18.

Moreover, the operation tree adheres to the following decoding rules:

- When a node (gene) is assigned with logarithm mathematical operation, the right node is neglected.
- When a node (gene) is assigned with variable or constant, the lower node is neglected.

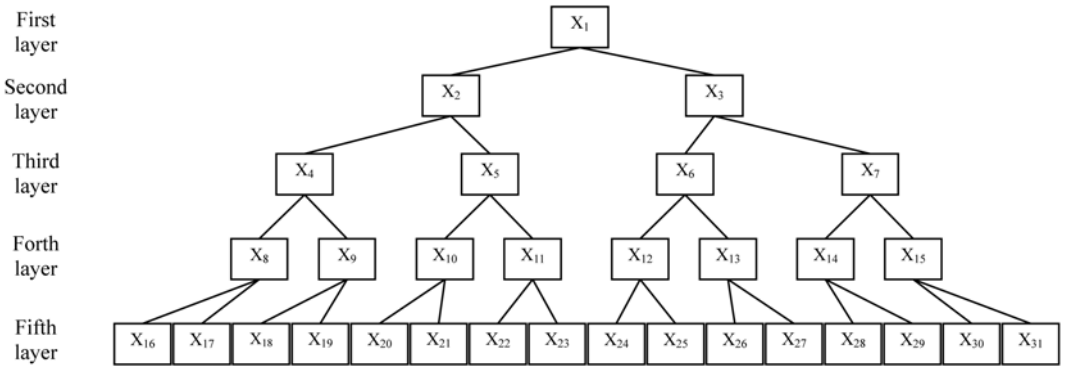

Fig. 4 Diagram of five-layered operation tree

Table 2 Genetic code of mathematical operations

Code	1	2	3	4	5	6
Operator	+	-	×	÷	x^y	ln

Table 3 Genetic code of variables and constants

Code	7	8	9	10	11	12	13	14	15	16	17	18
Variable	C	FL	SL	W	SP	CA	FA	W/C	W/B	W/S	TA/B	K

3.3. Modified predicted output value of operation tree

The predicted output values of operation tree usually have oblique phenomenon that there are high linear correlation but high root mean squared error (RMSE) between the predicted values and the actual values in dataset. Therefore, in this study, the single linear regression analysis was employed to modify the oblique phenomenon:

$$y = \alpha + \beta \cdot f \tag{2}$$

where

f = predicted output values of operation tree;
y = actual output values in dataset;
α and β = regression coefficients.

According to single linear regression analysis,

$$\alpha = \bar{y} - \beta \cdot \bar{f} \tag{3}$$

$$\beta = \frac{\sum_{i=1}^{n} (f_i - \bar{f}) \times (y_i - \bar{y})}{\sum_{i=1}^{n} (f_i - \bar{f})^2} \tag{4}$$

where:

$\bar{y}$ = the mean of the actual output values in the dataset;
$\bar{f}$ = the mean of the output values of the dataset predicted by the operation tree;
y_i = the actual output value of the ith data in the dataset;
f_i = the output value of the ith data point predicted by the operation tree.

3.4. Fitness function

As the objective of this study is to produce an accurate model to predict compressive strength, the Root Mean Squared Error (RMSE) was adopted as the evaluation function (fitness function) of solutions:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n} (\hat{y}_i - y_i)^2}{n}} \tag{5}$$

where $\hat{y}_i$ represents the modified predicted output value of the ith data point; y_i represents the actual output value of the ith data; and n represents the number of data.

3.5. GA parameters

This study adopted genetic algorithms to optimize the operation tree to produce the self-organized regression formula. There are some parameters may affect the performance of GA. Reference (Lei *et al*. 2005) suggested the following parameters: (1) population size=10~1000; (2) crossover rate=

0.4~0.99; (3) mutation rate=0.0001~0.1. In this study, GA parameters were set as: (1) population size=100; (2) crossover rate=0.9; (3) mutation rate=0.001; (4) stop criterion: the optimization process stops when the process cannot find a better solution in 1000 generations; (5) the elitist strategy was adopted.

4. Results

4.1. Operation tree before pruning

There is some randomness identified in the GA search process. Therefore, this study carried out the GOT process three times with different random seeds to produce three solutions. The results are shown in Figs. 5 through 7, and corresponding mathematical formulas are listed as follows:

$$y = 434.51 + 64.48 \times \frac{\ln(C \times W/S)}{W/S \times W/B} \tag{6}$$

$$y = 1.94 - 84635.21 \times \frac{\ln(C)}{(TA/B/\ln(C) - W - SP) \times W/B} \tag{7}$$

$$y = 263.401 - 871.06 \times \frac{\ln(W/C \times W/S)}{W/B} \tag{8}$$

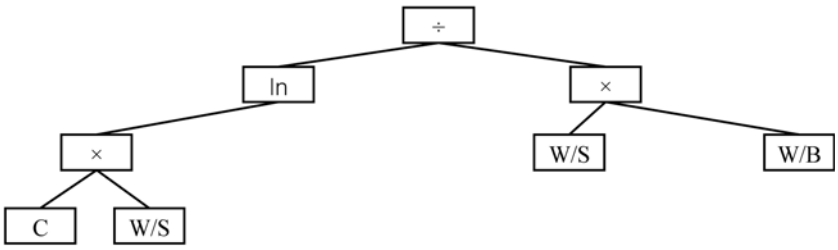

Fig. 5 Solution 1 of operation tree produced by GOT

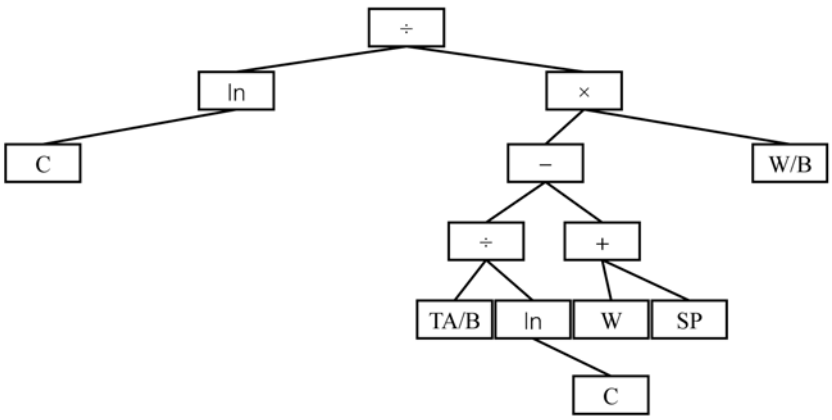

Fig. 6 Solution 2 of operation tree produced by GOT

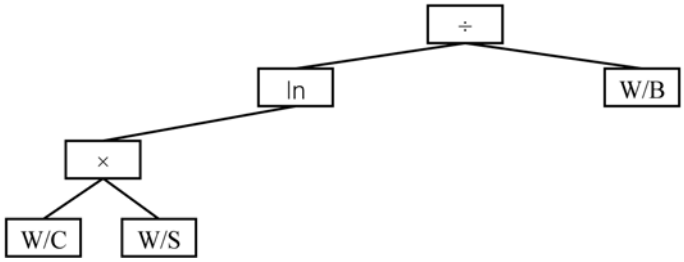

Fig. 7 Solution 3 of operation tree produced by GOT

Table 4 GOT data statistic before pruning

Group	Training RMS(MPa)	Testing RMS(MPa)
1	9.10	10.76
2	9.29	10.11
3	9.10	10.61

The RMSE error of training data and testing data of these formulas are presented in Table 4. The scatter diagram of the predicted value and the actual value of training data and testing data are shown in Figs. 8 through 13.

4.2. Operation tree after pruning

Detailed pruning processes are shown in Figs. 14 through 16, and the evaluation results of each pruned operation tree are listed in Tables 5 through 7. In order to discuss the effects of the number of variables, the pruning process would be conducted until only one variable remained in the tree instead of ending it according to the stop criterion. After pruning the three operation trees until only one variable remained, their corresponding mathematical formulas are listed as follows:

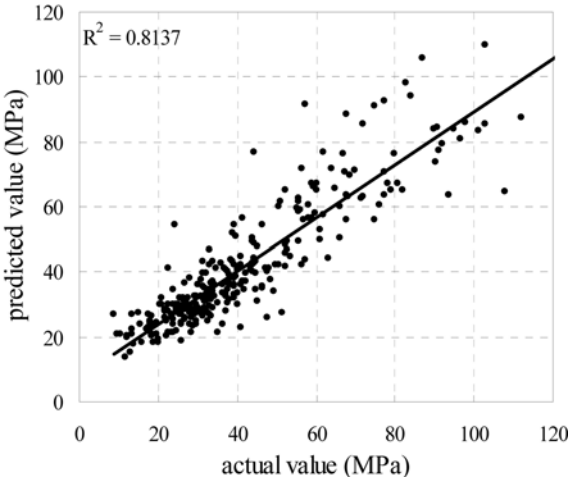

Fig. 8 Scatter diagram of training data of solution 1

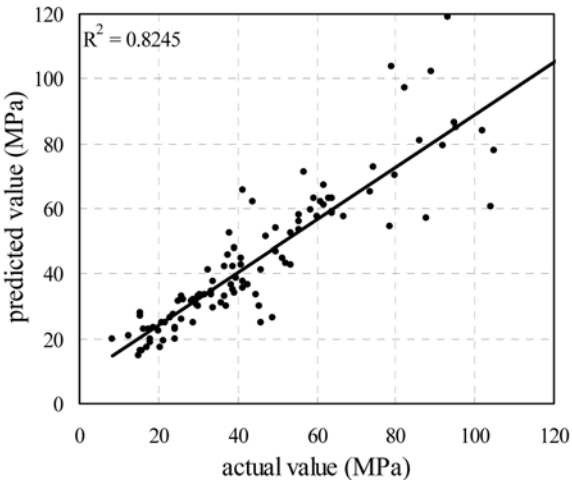

Fig. 9 Scatter diagram of testing data of solution 1

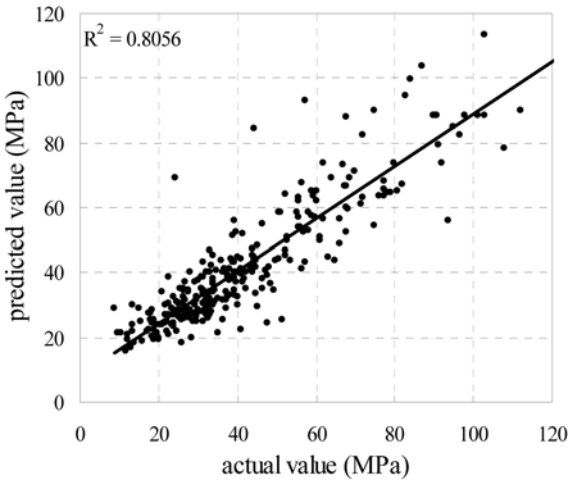

Fig. 10 Scatter diagram of training data of solution 2

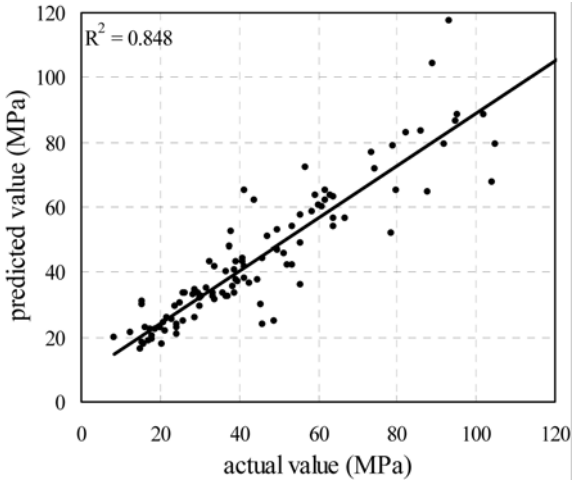

Fig. 11 Scatter diagram of testing data of solution 2

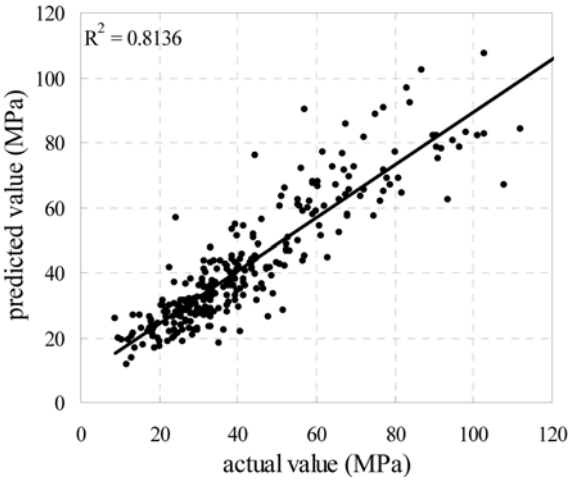

Fig. 12 Scatter diagram of training data of solution 3

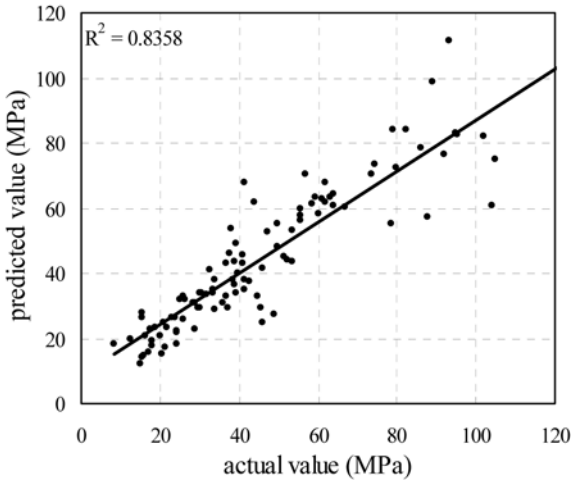

Fig. 13 Scatter diagram of testing data of solution 3

$$y = -3313.95 + \frac{4131.42}{W/B} \tag{9}$$

$$y = -3313.95 + \frac{4135.76}{W/B} \tag{10}$$

$$y = -3313.95 + \frac{4134.29}{W/B} \tag{11}$$

We can find that they have the same form:

$$y = \alpha + \beta \frac{1}{W/B} \tag{12}$$

where $\alpha < 0$ and $\beta > 0$.

The substantial meaning of the formula is that the most important variable in predicting 28-day

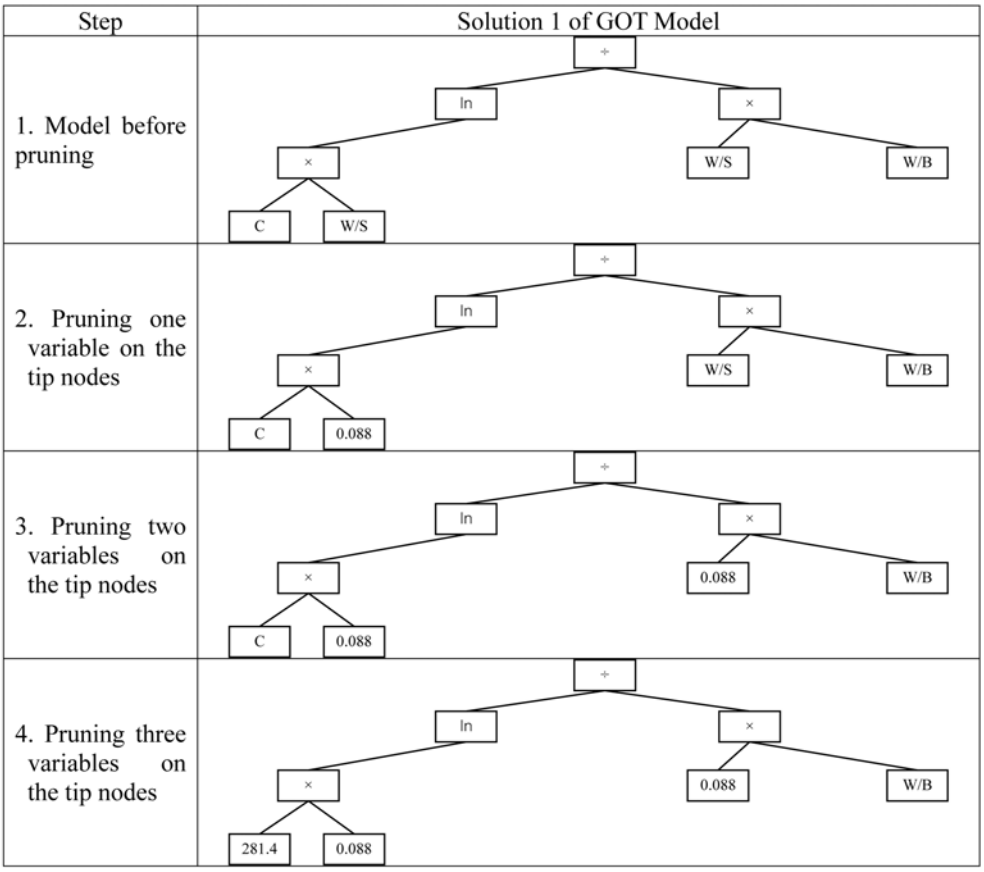

Fig. 14 Pruning process for solution 1

strength of HPC is water-binder ratio (W/B), and the compressive strength is in reverse proportion to W/B, which is consistent to the recognized knowledge in concrete material science. Moreover, regression coefficients in these formulas are almost the same, which shows that GOT is rather stable in combination with the pruning technique.

Moreover, the formulas in the last second step in Tables 5 through 7 are as follows:

$$y = -865.83 + 942.84 \times \frac{\ln(C) - 2.43}{W/B} \tag{13}$$

$$y = -1888.57 + 618.84 \times \frac{\ln(C)}{W/B} \tag{14}$$

$$y = -272.60 - 965.86 \times \frac{\ln(W/C) - 2.43}{W/B} = -272.60 + (-965.86 \cdot \ln(W/C) + 2347) \times \frac{1}{W/B} \tag{15}$$

Formula (9) and formula (10) show that under the same water-binder ratio (W/B), the greater the amount of cement, the higher the compressive strength. In formula (11), the smaller the W/C, the higher the compressive strength. These phenomena might represent a substantial new finding in the concrete material science field.

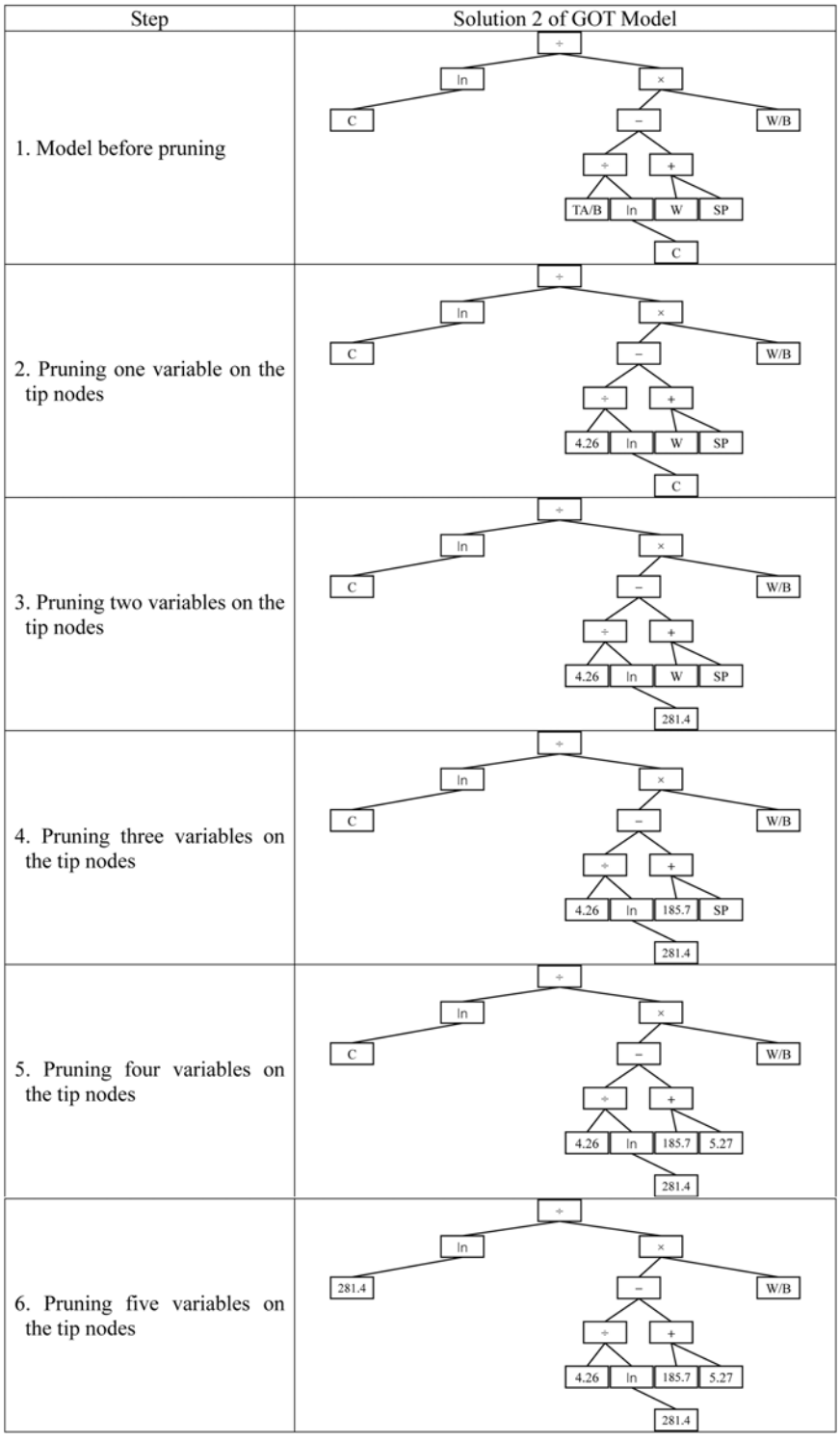

Fig. 15 Pruning process for solution 2

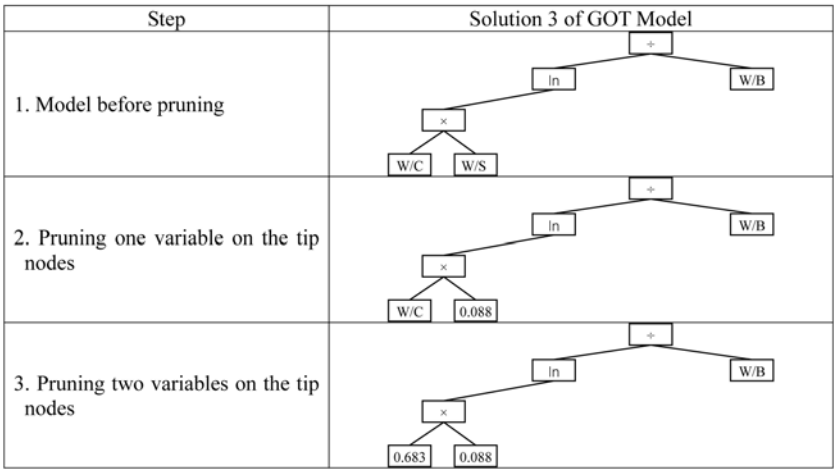

Fig. 16 Pruning process for solution 3

Table 5 Evaluation of the pruning process of solution 1

Step	Formula	Training R^2	Testing R^2	Training RMS(MPa)	Testing RMS(MPa)
1	$y = 434.51 + 64.48 \times \frac{\ln(C \times W/B)}{W/S \times W/B}$	0.81	0.82	9.10	10.80
2	$y = 958.13 + 57.19 \times \frac{\ln(C) - 2.43}{W/S \times W/B}$	0.81	0.81	9.27	11.19
3	$y = -865.83 + 942.84 \times \frac{\ln(C) - 2.43}{W/B}$	0.79	0.77	9.74	12.74
4	$y = -3313.95 + \frac{4131.42}{W/B}$	0.76	0.76	10.22	12.96

Table 6 Evaluation of the pruning process for solution 2

Step	Formula	Training R^2	Testing R^2	Training RMS(MPa)	Testing RMS(MPa)
1	$y = 1.94 - 84635.21 \times \frac{\ln(C)}{(TA/B/\ln(C) - W - SP) \times W/B}$	0.81	0.85	9.29	10.11
2	$y = 11.28 - 84494.88 \times \frac{\ln(C)}{(4.26/\ln(C)W - SP) \times W/B}$	0.81	0.85	9.29	10.11
3	$y = 14.83 - 84440.97 \times \frac{\ln(C)}{(0.755 - W - SP) \times W/B}$	0.81	0.85	9.29	10.11
4	$y = -1963.75 - 118962.63 \times \frac{\ln(C)}{(0.755 - 185.7 - SP) \times W/B}$	0.80	0.81	9.40	11.76
5	$y = -1888.57 + 618.84 \times \frac{\ln(C)}{W/B}$	0.79	0.78	9.58	12.51
6	$y = -3313.95 + \frac{4135.76}{W/B}$	0.76	0.76	10.22	12.96

Table 7 Evaluation of the pruning process of solution 3

Step	Formula	Training R^2	Testing R^2	Training RMS (MPa)	Testing RMS (MPa)
1	$y = 263.40 - 871.06 \times \frac{\ln(W/C \times W/S)}{W/B}$	0.81	0.84	9.10	10.60
2	$y = -272.60 - 965.86 \times \frac{\ln(W/C) - 2.43}{W/B}$	0.81	0.81	9.23	11.47
3	$y = -3313.95 + \frac{4134.29}{W/B}$	0.76	0.76	10.22	12.96

4.3. Stepwise regression analysis

This study adopted stepwise regression analysis with backward elimination technique to make a comparison with GOT. The steps are as follows:

Step 1: Use all N input variables to set up a regression analysis model, then try to eliminate one input variable and employ the remaining N-1 input variables to set up a regression analysis model. Choose the simplified model with minimum testing data error.

Step 2: Repeat step 1 until no improvement can be reached by eliminating another variable in the simplified model.

Step 3: Output the simplified model.

The detailed elimination process and the evaluation results of each regression model are listed in Table 8. To discuss the effects of the number of variables, this study did not follow the stop criterion that the process would be repeated until no improvement can be reached by eliminating a variable in the simplified model. The eliminating process was allowed to continue until only one variable remained. From the results in Table 8, we find that:

4.3.1. Evaluation of training dataset

In the variable elimination process, we find that after eliminating six variables, the accuracy of the simplified model becomes slightly lower for the training dataset. The five variables remaining include cement (C), fly ash (FL), slag (SL), water (W), and superplasticizer (SP), which are the components of the water-binder ratio.

$$W/B\ ratio = \frac{W + SP}{C + FL + SL} \tag{16}$$

While eliminating more variables, the accuracy sharply becomes lower on training data set. This phenomena shows that when the variable set does not contain all water-binder ratio components, the corresponding regression model cannot predict the strength accurately. This finding is consistent with recognized concrete material science knowledge and GOT results.

As shown in Table 8, the elimination process sequentially eliminates SP, FL, SL, W, and C, and their corresponding regression model becomes rapidly less accurate. The phenomena shows that the most important variables in building the strength model, from least to most important, are SP, FL, SL, W, and C, with the contribution significance of the three kinds of binder to form strength (also ranked from least to most important) including fly ash (FL), slag (SL), and cement (C). This finding is also consistent with recognized concrete material science knowledge.

Table 8 Stepwise regression with backward elimination technique

Step	Formula	Training R^2	Testing R^2	Training RMS (MPa)	Testing RMS (MPa)	Eliminated variable
1	$f'_c = -36390.90 + 41.46C + 35.00FL + 36.46SL - 341.27W - 378.06SP + 23.99CA + 24.39FA - 340.44W/C + 7539.82W/B + 561452.70W/S - 993.69TA/B$	0.82	0.68	9.05	15.45	None
2	$f'_c = -35514.50 + 41.21C + 33.71FL + 35.2SL - 335.87W - 370.94SP + 23.39CA + 23.70FA + 5889.38W/B + 555362.9W/S - 880.69TA/B$	0.82	0.69	9.06	15.24	W/C
3	$f'_c = -36644.20 + 41.09C + 33.20FL + 34.99SL - 333.89W - 369.50SP + 22.84CA + 23.31FA + 579143.2W/S - 341.44TA/B$	0.82	0.68	9.07	15.40	W/B
4	$f'_c = 11417.09 + 21.45C + 12.28FL + 14.69SL - 113.23W - 143.78SP + 0.99CA + 91990.31W/S - 214.787TA/B$	0.80	0.81	9.47	11.60	FA
5	$f'_c = 13127.86 + 20.44C + 11.37FL + 13.59SL - 104.64W - 140.13SP + 72477.77W/S - 264.72TA/B$	0.80	0.81	9.49	11.80	CA
6	$f'_c = 10648.46 + 23.86C + 14.96FL + 17.14SL - 114.35W - 148.43SP + 91558.32W/S$	0.80	0.81	9.51	11.79	TA/B
7	$f'_c = 10386.55 + 21.79C + 14.41FL + 16.38SL - 66.52W - 101.53SP$	0.78	0.81	9.79	11.58	W/S
8	$f'_c = 9274.43 + 20.78C + 8.82FL + 14.32SL - 59.42W$	0.77	0.76	10.22	12.91	SP
9	$f'_c = 10871.31 + 18.50C + 11.75SL - 61.11W$	0.74	0.77	10.75	12.64	FL
10	$f'_c = 12238.92 + 14.83C - 57.89W$	0.65	0.73	12.47	13.82	SL
11	$f'_c = 1254.752 + 16.30C$	0.45	0.47	15.57	19.44	W

4.3.2. Evaluation of the testing dataset

In the process of eliminating variables, we can find that after eliminating six variables, the accuracy of the simplified model progressively becomes higher in the testing dataset. As more

Table 9 Stepwise neural network

Step	Training R^2	Testing R^2	Training RMS(MPa)	Testing RMS(MPa)	Eliminated variable
1	0.85	0.88	8.09	8.83	None
2	0.87	0.83	7.56	10.71	W/C
3	0.87	0.87	7.56	9.50	W/B
4	0.87	0.87	7.62	9.51	CA
5	0.87	0.88	7.70	9.01	FA
6	0.84	0.87	7.81	9.16	TA/B
7	0.85	0.87	8.23	9.32	W/S
8	0.81	0.79	9.18	11.75	SP
9	0.77	0.82	10.05	11.15	FL
10	0.69	0.76	11.81	12.76	SL
11	0.50	0.50	15.00	18.30	W

variables are eliminated, the accuracy sharply lowers for the testing dataset. This phenomena shows that the model with more than five variables happens to be over-fitted, while the model with less than five variables happens to be under-fitted, and the model with the most important five variables happens to be right-fitted and form the most accurate strength regression model with the highest generalization prediction ability.

4.4. Stepwise neural network

This study adopted back-propagation neural networks (BPNs) to make a comparison with GOT. In this study, network parameters such as number of hidden neurons, learning rate, momentum factor, and number of learning cycles were determined according to minimizing the RMSE error on the testing dataset. To make a comparison with GOT and stepwise regression, this study proposed a stepwise neural network similar to stepwise regression analysis with backward elimination technique.

The detailed elimination process and the evaluation results of each BPN model are listed in Table 9. From these results, we see that the variable elimination sequence is almost the same as stepwise regression (only the positions of the third and the fourth eliminated variables are exchanged), and the accuracy progress of stepwise neural network is rather similar that of stepwise regression, which also, in sequence, showed the over-fitting, right-fitting, and under-fitting processes. The model with the most important five variables happened to be right-fitting and formed the most accurate strength model with the highest generalization prediction ability.

4.5. Comparisons of methods

The model accuracy refers to the generalization prediction ability, i.e., the smaller the RMSE or the higher the R^2 in the testing dataset, the higher the accuracy. The model complexity is defined as the number of variables in the model. The larger the number, the more complex and less easily comprehended the model is. The comparisons of model accuracy and complexity of above-mentioned three methods are shown in Tables 10 and 11 as well as Figs. 17 through 20. We can find that:

Table 10 Relationship between R^2 and number of variable

Number of variable	Solution 1 of GOT		Solution 2 of GOT		Solution 3 of GOT		Stepwise regression		Stepwise BPN	
	Training set	Testing set	Training set	Testing set	Training set	Testing set	Training set	Testing set	Training set	Testing set
11	NA	NA	NA	NA	NA	NA	0.82	0.68	0.85	0.88
10	NA	NA	NA	NA	NA	NA	0.82	0.69	0.87	0.83
9	NA	NA	NA	NA	NA	NA	0.82	0.68	0.87	0.87
8	NA	NA	NA	NA	NA	NA	0.80	0.81	0.87	0.87
7	NA	NA	NA	NA	NA	NA	0.80	0.81	0.87	0.88
6	NA	NA	0.81	0.85	NA	NA	0.80	0.81	0.84	0.87
5	NA	NA	0.81	0.85	NA	NA	0.78	0.81	0.85	0.87
4	0.81	0.82	0.81	0.85	NA	NA	0.77	0.76	0.81	0.79
3	0.81	0.81	0.80	0.81	0.81	0.84	0.74	0.77	0.77	0.82
2	0.79	0.77	0.79	0.78	0.81	0.81	0.65	0.73	0.69	0.76
1	0.76	0.76	0.76	0.76	0.76	0.76	0.45	0.47	0.5	0.5

Table 11 Relationship between RMSE and number of variable

Number of variable	Solution 1 of GOT		Solution 2 of GOT		Solution 3 of GOT		Stepwise regression		Stepwise BPN	
	Training set	Testing set	Training set	Testing set	Training set	Testing set	Training set	Testing set	Training set	Testing set
11	NA	NA	NA	NA	NA	NA	9.05	15.45	8.09	8.83
10	NA	NA	NA	NA	NA	NA	9.06	15.24	7.56	10.71
9	NA	NA	NA	NA	NA	NA	9.07	15.40	7.56	9.50
8	NA	NA	NA	NA	NA	NA	9.47	11.60	7.62	9.51
7	NA	NA	NA	NA	NA	NA	9.49	11.80	7.70	9.01
6	NA	NA	9.29	10.11	NA	NA	9.51	11.79	7.81	9.16
5	NA	NA	9.29	10.11	NA	NA	9.79	11.58	8.23	9.32
4	9.10	10.80	9.29	10.11	NA	NA	10.22	12.91	9.18	11.75
3	9.27	11.19	9.40	11.76	9.10	10.60	10.75	12.64	10.05	11.15
2	9.74	12.74	9.58	12.51	9.23	11.47	12.47	13.82	11.81	12.76
1	10.22	12.96	10.22	12.96	10.22	12.96	15.57	19.44	15.00	18.30

- Model accuracy: When the number of variables or nodes of the three methods exceeds five, the model accuracy is highest in the stepwise neural network, followed by GOT and stepwise regression. When the number of variables or nodes is less than three, the accuracy of both the stepwise neural network and stepwise regression begin to drop sharply, while GOT accuracy falls only slightly. The result is that when less than three variables are used, model accuracy is highest in GOT, followed by the stepwise neural network and stepwise regression.
- Model complexity: Operation trees produced by GOT before pruning used three to six variables. Their average of R^2 and RMSE on the testing dataset was 0.837 and 10.51 MPa, respectively. Stepwise regression was unable to reach the necessary level of accuracy, while neural network required to use more than five variables to obtain the same level of accuracy.

According to the above-mentioned evaluations, this study verified that the pruning technique can simplify the operation tree, while maintaining a similarly high level of accuracy in order to predict the 28-day compressive strength of high-performance concrete.

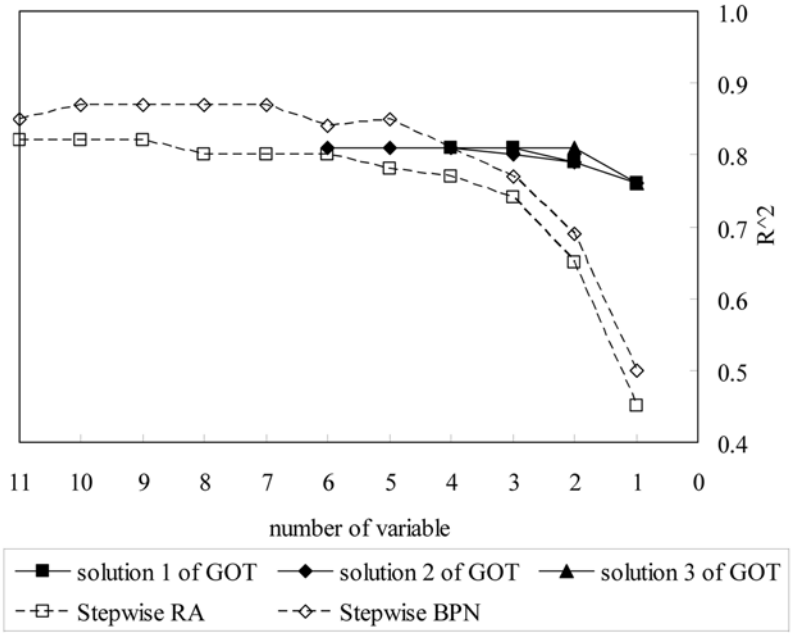

Fig. 17 Line chart of R^2 on training dataset and number of variable

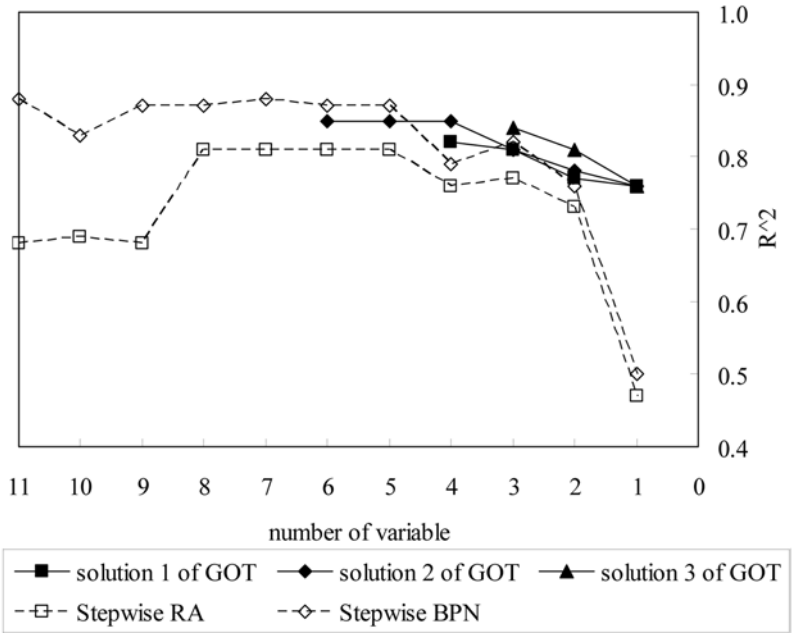

Fig. 18 Line chart of R^2 on testing dataset and number of variable

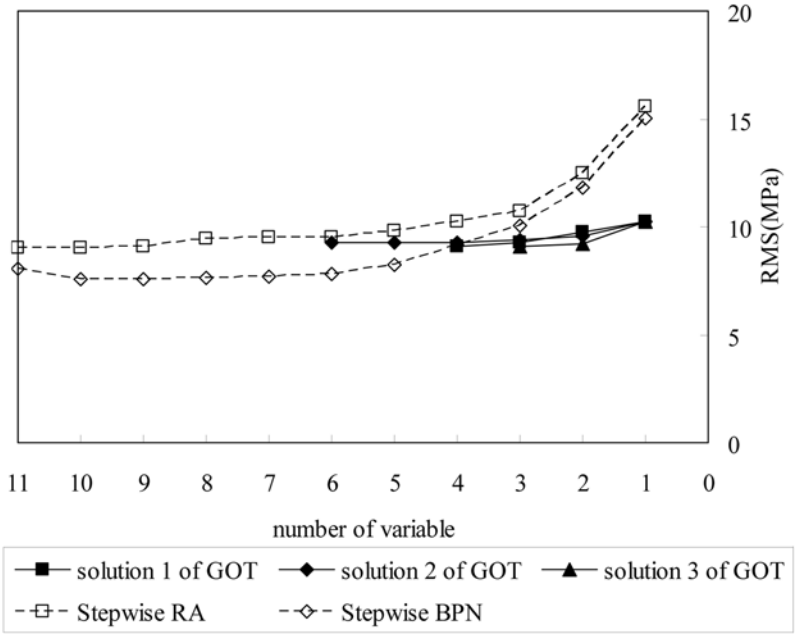

Fig. 19 Line chart of RMSE on training dataset and number of variable

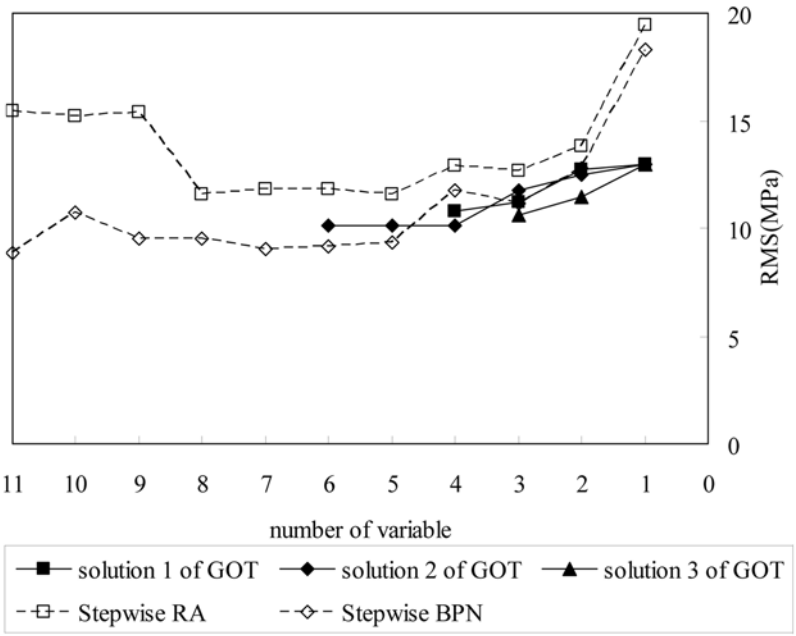

Fig. 20 Line chart of RMSE on testing dataset and number of variable

5. Conclusions

This study led to the following conclusions:

1. One-variable formulas generated by GOT with pruning technique are rather similar, which

showed technique stability and reliability. The most important variable in predicting 28-day compressive strength of HPC is shown to be the water-binder ratio (W/B). It also found that water-binder ratio strength is the reverse of material strength. Such results are consistent with recognized concrete material science knowledge.

2. In the process of eliminating stepwise regression analysis and stepwise neural network variables, the accuracy of model for training datasets always decreased, while the testing dataset slightly increased in the early stage to reach an early maximum level before sharply decreasing in the late stage. Such results showed there are over-fitting and under-fitting phenomenon in the early and late stages. With the elimination of certain unimportant variables, the regression analysis and neural network is able to build the model with the highest generalization prediction ability.

3. In comparing model accuracy and complexity, when the number of variables in the strength model exceeded five, the accuracy is highest in stepwise neural network, followed by GOT and stepwise regression. When the number of variables are less than three, the accuracy of stepwise neural network and stepwise regression decrease sharply, while GOT decrease only slightly. The accuracy for models using less than three variables is thus highest in GOT, followed by the stepwise neural network and stepwise regression.

From the four above-mentioned evaluations, this study proved that the proposed pruning technique can simplify the operation tree while maintaining a high level of accuracy in order to predict 28-day compressive strength of HPC.

References

Huang, C.L. (1999), *Characteristics and Behaviors of Concrete*, Chan's Arch Books, Taipei.

Yeh, I.C. (2004), *Applications of Artificial Neural Networks*, Scholars books, Taipei.

Yeh, I.C. (1998), "Modeling concrete strength with augment-neuron networks", *J. Mater. Civil Eng.*, **10**(4), 263-268.

Yeh, I.C. (1999), "Modeling of strength of high performance concrete using artificial neural networks", *Cement Concrete Res.*, **28**(12), 1797-1808.

Kim, J.I., Kim, D.K., Feng, M.Q., and Yazdani, F. (2004), "Application of neural networks for estimation of concrete strength", *J. Civil Eng.*, **16**(4), 257-264.

Chen, L. (2003), "A study of applying macroevlutionary genetic programming to concrete strength estimation", ASCE, *J. Comput. Civil Eng.*, **17**(4), 290-294.

Chen, L., Tasi, C.S., and Chen, H.M. (2004), "A study of applying grammar evolution to concrete strength estimation", *Chung Hua J. Sci. Eng.*, **2**(2), 55-62.

Hamid-Zadeh, N., Jamali, A., Nariman-Zadeh, N., and Akbarzadeh, H. (2007), "Prediction of concrete compressive strength using evolved polynomial neural networks", *World Sci. Eng. Acad. Soc. T. Syst.*, **6**(4), 802-807.

Ahmet, O., Murat, P., Erdogan, O., Erdogan, K., Naci C., and Bhatti, M.A. (2006), "Predicting the compressive strength and slump of high strength concrete using neural network", *Constr. Build. Mater.*, **20**(9), 769-775.

Lien, L.C., Yeh, I.C., and Cheng, M.Y. (2006), "Modeling strength of high-performance concrete using genetic algorithms and operation tree", *J. Tech.*, **21**(1), 41-54.

Davis, L. (1991), *Handbook of Genetic Algorithms*, Van Nostrand Reinhold, NY.

Goldberg, D.E. (1989), *Genetic Algorithms in Search Optimization and Machine Learning*, Addison-Wesley Publishing Company, Massachusetts.

Lei, I.J., Chang, S.C., Li, H.W., and Chou, C.M. (2005), *An Introduce of GA Toolbox in MATLAB*, Xidian University Press.