GROUTABILITY PREDICTION OF MICROFINE CEMENT BASED SOIL IMPROVEMENT USING EVOLUTIONARY LS-SVM INFERENCE MODEL

Min-Yuan CHENG[a], Nhat-Duc HOANG[b]

[a]*Department of Civil and Construction Engineering,*
National Taiwan University of Science and Technology, Taiwan
[b]*Center of Research and Development, Duy Tan University, Vietnam*

Abstract. Permeation grouting is a widely used technique for soil improvement in construction engineering. Thus, predicting the results of the grouting activity is a particularly interesting topic that has drawn the attention of researchers both from the academic field and industry. Recent literature has indicated that artificial intelligence (AI) approaches for groutability prediction are capable of delivering better performance than traditional formula-based ones. In this study, a novel AI method, evolutionary Least Squares Support Vector Machine Inference Model for groutability prediction (ELSIM-GP), is proposed to forecast the result of grouting activity that utilizes microfine cement grout. In the model, Least Squares Support Vector Machine (LS-SVM) is a supervised machine learning technique that is employed to learn the decision boundary for classifying high dimensional data. Differential Evolution (DE) is integrated into ELSIM-GP for automatically optimizing its tuning parameters. 240 historical cases of grouting process for sandy silt soil have been collected to train, validate, and test the inference model. Experimental results demonstrated that ELSIM-GP can overcome other benchmark approaches in terms of forecasting accuracy. Therefore, the proposed approach is a promising alternative for predicting groutability.

Keywords: groutability prediction, least squares support vector machine, differential evolution, machine learning, artificial intelligence.

Introduction

In construction industry, soil improvement by means of permeation grouting is a process that is often carried out to reduce the water movement through soils (Zebovitz *et al.* 1989). Particularly for underground construction, inflow of groundwater has always been a critical issue for engineers (Butron *et al.* 2009). Incidents of inflow can bring about construction delay, and even cause serious damages for the quality of structures. As a consequence, permeation grouting is a crucial task that is needed to be accomplished in a majority of excavation and tunnel projects.

Among the grouts used for permeation grouting, microfine cement has been increasingly applied in the industry. The reason is that it usually provides improved groutability of the target geomaterial and it does not cause groundwater pollution in surrounding environment (Perret *et al.* 2000; Zebovitz *et al.* 1989). Furthermore, the microfine cement based grout is proven to have the capacity of filling cracks with small openings as well as penetrating fine soils with very low permeability (Perret *et al.* 2002).

Noticeably, grouting activity is considered to be successful if the grout can be sufficiently injected into the soil. According to Liao *et al.* (2011), the grouting activity succeeds if the injected grout is at least two times the volume of the void space under the split pressure. In practice, an accurate prediction of this activity using microfine cement is by no means an easy task (Akbulut, Saglamer 2002). It is because the suitability of conventional predictive formulas, which are mostly based on the grain-size of the soil and the grout, is unreliable for semi-nanometer scale grout (Liao *et al.* 2011). Furthermore, the grain-size is not the sole factor that affects the consequence of the grouting process.

Various research works have investigated grouting predictability. Akbulut and Saglamer (2002) and Ozgurel and Cumaraswamy (2005) found that in addition to the size of the soil and the grout, the water-to-cement ratio of the grout (w/c), the void size in soil, and the fines content (FC) of the total soil should be taken into account (Akbulut, Saglamer 2002). Liao *et al.* (2011) pointed out that inclusion of soil gradation information, namely the coefficient of uniformity (C_u), which measures the particle size range, and the coefficient of gradation (C_z), which characterizes the particle size curve, can boost the overall predictive performance. Needless to say, it is beneficial to take into account these factors for estimation of the grouting process (Tekin, Akbas 2011).

Because construction projects are highly uncertain and inherently context-dependent, artificial intelligence (AI) methods may provide viable alternatives for groutability prediction problem. The inference model, which

Corresponding author: Nhat-Duc Hoang
E-mail: M9705805@mail.ntust.edu.tw; ducxd85@yahoo.com

Taylor & Francis
Taylor & Francis Group

composes of various AI techniques, can be utilized to derive new facts from historical data (Cheng, Wu 2009). The inference process changes adaptively in response to alteration in historical data. Used nomenclature is presented in Figure 1.

w/c	Water-to-cement ratio of the grout	γ	The regularization constant
FC	The fines content (FC) of the total soil	α_k	The Lagrange multipliers
C_u	Coefficient of uniformity	$\phi(x_k)$	A kernel function
C_z	Coefficient of gradation	σ	RBF kernel function parameter
N	Number of training data	D	Number of decision variables
x_k	Input data point	NP	Population size
y_k	Class label	g	Current generation of DE
e_k	LS-SVM prediction error	$X_{i,0}$	Decision variable at the first generation
w	The normal vector of the hyperplane	$X_{r,g}$	A randomly chosen vector
b	The bias of the hyperplane	$V_{i,g}$	A mutant vector
$U_{j,i,g+1}$	The trial vector	F	Mutation scale factor
Cr	Crossover probability	G_{max}	Maximum number of generations
X_o	Original data value	X_{max}	Maximum value of a data field
X_{min}	Minimum value of a data field	X_n	Normalized data value
R_{TR}	The accuracy rate for training set	R_{VA}	The accuracy rate for validating set

Fig. 1. Nomenclature

Notably, the problem of groutability prediction, in essence, can be modeled as a classification task that contains two class labels ("success" and "failure"). Therefore, AI based classifiers, such as Classification and Regression Trees (CART), Artificial Neural Network (ANN), and Least Squares Support Vector Machine (LS-SVM) can be feasible in coping with the aforementioned problem. The main motivation of selecting CART, BPNN, RBFNN, and LS-SVM is that they are popular and effective approaches for classifying data in high dimensional space (Brown *et al.* 1993; Olson *et al.* 2011). Furthermore, these techniques represent different learning mechanisms that can be worth investigating in the case of groutability prediction.

CART (Breiman *et al.* 1984) is a popular machine learning technique which utilizes historical data to construct decision trees. One major advantage of the decision tree based model is its ability to mitigate the negative effect of outliers because the model is capable of isolating the outliers in a separate node. However, one disadvantage of CART is that it may produce unstable decision trees (Timofeev 2004). The reason is that insignificant modification of learning sample could result in radical changes in the model. Moreover, as a replacement of stopping rules, CART generates a sequence of sub-trees by growing a large tree and pruning the tree back until only the root node is left (Loh 2011).

Artificial Neural Network (ANN) has been applied to deal with groutability prediction as well as with other problems in construction industry (Chen *et al.* 2009; Kalinli *et al.* 2011; Liao *et al.* 2011; Tekin, Akbas 2011). This approach eliminates the requirement to find a mapping relationship that mathematically describes the problem of interest. Studies have shown that ANN is a viable substitution for traditional deterministic methods in coping with sophisticated problems (Cheng *et al.* 2010; Wong *et al.* 1997). When the influence factors and the structure of ANN are all specified, the task boils down to collecting a reasonable number of data to train the ANN. There are two types of ANN that are commonly utilized for classification: Back-propagation Neural Network (BPNN) and Radial Basis Function Neural Network (RBFNN).

The implementation of BPNN for solving machine learning tasks is exposed to several drawbacks. This method suffers difficulties in selecting a large number of controlling parameters such as number of hidden layers, number of neurons in a hidden layer and learning rate (Bao *et al.* 2005). Notably, the training process of BPNN is notoriously time-consuming. Furthermore, one major disadvantage of this approach is that its training process is achieved through a gradient descent algorithm on the error space, which can be very complex and may contain many local minima (Kiranyaz *et al.* 2009). Thus, training process is likely to be trapped into a local minimum and this definitely hinders the forecasting performance.

RBFNN is also an efficient learning algorithm for carrying out pattern classification. In this learning mechanism, radial basis functions (RBF) are embedded into the network structure (Pendharkar 2011). The training process of RBFNN comprises of two stages which can be accomplished with low computational expenses (Bishop 1995). In the first stage, an unsupervised learning algorithm, such as the orthogonal least squares (OLS) method, is used to determine a suitable set of RBF centers from a large set of training data points (Chen *et al.* 1991). The second stage requires the determination of the network's weights which is achieved by solving a linear system. Notably, for this approach, back-propagation computation is not required in specifying the parameters of the hidden layer (Liao *et al.* 2011). Therefore, RBFNN can be trained much faster than BPNN (Bishop 1995; Wu, Liu 2012).

Proposed by Suykens *et al.* (De Brabanter *et al.* 2010; Suykens *et al.* 2002), LS-SVM is an advanced machine learning technique which possesses many advanced features reflected in its good generalization and fast computation. In LS-SVM's training process, a least squares cost function is proposed to obtain a linear set of equations in the dual space. Consequently, to derive the solution, it is required to solve a set of linear equations that can be efficiently solved by iterative methods such as conjugate gradient (Wang, Hu 2005). Despite of its superiority, the application of this approach in construction engineering is still very limited. Surprisingly, no studies have investigated the capability of LS-SVM on predicting the groutability.

Furthermore, the mechanism for setting models' control parameters is an important problem in the field of

AI. This crucial issue has been widely recognized by many scholars in a variety of disciplines (Cheng *et al.* 2011; Lean *et al.* 2009; Zhou *et al.* 2011). In practice, identifying the most suitable set of model's parameters, in essence, represents an optimization problem. Therefore, hybridizing the AI based techniques with an evolutionary optimization algorithm is a prevalent research direction (Cheng, Roy 2011; Cheng, Wu 2009; Yu 2011).

Evolutionary computation is characterized by iterative progresses used to guide the randomly initialized population to the final optimal solution. Among evolutionary optimization techniques, DE, proposed by Storn and Price (Price *et al.* 2005), is a population-based stochastic search engine which is efficient and effective for global optimization in the continuous domain. It uses mutation, crossover, and selection operators at each generation to move its population toward the global optimum. Superior performance of DE, in terms of accuracy and fast operation, has been verified in many reported research works (Price *et al.* 2005; Storn, Price 1997).

Thus, this article proposes to fuse LS-SVM and DE to construct an inference model for groutability prediction using microfine cement. The remaining part of this paper is organized as follows. The second and the third sections of this paper review related literature on LS-SVM and DE. The framework of the proposed model ELSIM-GP is depicted in the fourth section. The fifth section demonstrates the experimental results. Conclusion on our study is mentioned in the final section.

2. Least Squares Support Vector Machine for classification

Given a training dataset $\{x_k, y_k\}_{k=1}^N$ with input data $x_k \in R^n$ where N is the number of training data points, n is the data dimension, and the corresponding class labels is denoted as $y_k \in \{-1, +1\}$, the LS-SVM formulation for classification is presented as follows (Suykens *et al.* 2002; Suykens, Vandewalle 1999):

$$\text{Minimize } J_p(w, e) = \frac{1}{2} w^T w + \gamma \frac{1}{2} \sum_{k=1}^N e_k^2, \qquad (1)$$

$$\text{Subjected to } y_k(w^T \varphi(x_k) + b) = 1 - e_k, \ k = 1, ..., N, \qquad (2)$$

where $w \in R^n$ is the normal vector to the classification hyperplane and $b \in R$ is the bias; $e_k \in R$ are error variables; $\gamma > 0$ denotes a regularization constant.

The Lagrangian is given by:

$$L(w, b, e; \alpha) = J_p(w, e) -$$
$$\sum_{k=1}^N \alpha_k \{y_k(w^T \varphi(x_k) + b) - 1 + e_k\}, \qquad (3)$$

where: α_k are Lagrange multipliers; $\varphi(x_k)$ represents a kernel function. The conditions for optimality are given by:

$$\begin{cases} \dfrac{\partial L}{\partial w} = 0 \to w = \sum_{k=1}^N \alpha_k y_k \varphi(x_k) \\[2mm] \dfrac{\partial L}{\partial b} = 0 \to \sum_{k=1}^N \alpha_k y_k = 0 \\[2mm] \dfrac{\partial L}{\partial e_k} = 0 \to \alpha_k = \gamma e_k, \ k = 1, ..., N \\[2mm] \dfrac{\partial L}{\partial \alpha_k} = 0 \to y_k(w^T \varphi(x_k) + b) - 1 + e_k = 0, \ k = 1, ...N. \end{cases} \qquad (4)$$

After elimination of *e* and *w*, the following linear system is obtained:

$$\begin{bmatrix} 0 & y^T \\ y & \omega + \gamma^{-1} I \end{bmatrix} \begin{bmatrix} b \\ \alpha \end{bmatrix} = \begin{bmatrix} 0 \\ 1_v \end{bmatrix}, \qquad (5)$$

where $y = y_1, ..., y_N$, $1_v = [1; ...; 1]$, and $\alpha = [\alpha_1; ...; \alpha_N]$. And the kernel function is applied as follows:

$$\omega = y_i y_j \varphi(x_k)^T \varphi(x_l) = y_i y_j K(x_k, x_l). \qquad (6)$$

The resulting LS-SVM model for function estimation is expressed as:

$$y(x) = sign(\sum_{k=1}^N \alpha_k y_i K(x_k, x_l) + b), \qquad (7)$$

where α_k and *b* are the solution to the linear system (5). The kernel function that is often utilized is Radial Basis Function (RBF) kernel. Description of RBF kernel is given as follows:

$$K(x_k, x_l) = \exp(\frac{\|x_k - x_l\|^2}{2\sigma^2}), \qquad (8)$$

where σ is the kernel function parameter.

In the case of the RBF kernel, there are two tuning parameters (γ, σ) that are needed to be determined in LS-SVM. The regularization parameter (γ) is used to weight the importance of classification errors. Meanwhile, the kernel parameter (σ) affects the kernel width. It is worth noticing that proper setting of these tuning parameters is required to ensure desirable performance of the prediction model (Suykens *et al.* 2002).

3. Differential evolution

This section describes the algorithm of Differential Evolution (DE) proposed by Storn and Price (Price *et al.* 2005; Storn, Price 1997). The algorithm (Fig. 2) consists of five main stages: initialization, mutation, crossover, selection, and stopping condition verification. Given that the problem at hand is to minimize a cost function *f(X)*, where the number of decision variables is *D*, we can describe each stages of DE in details.
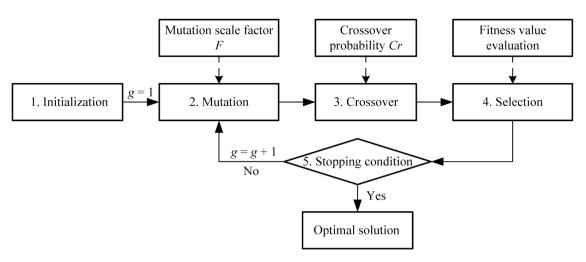
Fig. 2. Differential Evolution optimization algorithm

3.1. Initialization

DE begins the search process by randomly generating *NP* number of D-dimensional parameter vectors $X_{i,g}$, where $i = 1, 2, …, NP$ and *g* represents the current generation. In DE algorithm, *NP* does not change during the optimization process (Storn, Price 1997). Moreover, the initial population (at $g = 0$) ought to cover the entire search space in a uniform manner. Thus, we can simply generate these individuals as follows:

$$X_{i,0} = LB + rand[0,1] \times (UB - LB), \qquad (8)$$

where: $X_{i,0}$ is the decision variable *i* at the first generation; *rand*[0,1] denotes a uniformly distributed random number between 0 and 1. *LB* and *UB* are two vectors of lower bound and upper bound for any decision variable.

3.2. Mutation

A vector in the current population (or parent) is called a target vector. Hereafter, the terms parent and target vector are used interchangeably. For each target vector, a mutant vector is produced via the following equation (Storn, Price 1997):

$$V_{i,g+1} = X_{r1,g} + F(X_{r2,g} - X_{r3,g}), \qquad (9)$$

where: *r*1, *r*2, and *r*3 are three random indexes lying between 1 and *NP*. $X_{r1,g}$, $X_{r2,g}$, and $X_{r3,g}$ are three random vectors in the current generation *g*. These three randomly chosen integers are also selected to be different from the index *i of* the target vector. *F* denotes the mutation scale factor, which controls the amplification of the differential variation between $X_{r2,g}$ and $X_{r3,g}$. $V_{i,g+1}$ represents the newly created mutant vector.

3.3. Crossover

The purpose of the crossover stage is to diversify the current population by exchanging components of target vector and mutant vector. In this stage, a new vector,

named as trial vector, is created. The trial vector is also called the offspring. The trial vector can be formed as follows:

$$U_{j,i,g+1} = \begin{cases} V_{j,i,g+1}, & if \ \ rand_j \leq Cr \ \ or \ \ j = rnb(i) \\ X_{j,i,g}, & if \ \ rand_j > Cr \ \ and \ \ j \neq rnb(i) \end{cases}, \qquad (10)$$

where: $U_{j,i,g+1}$ is the trial vector; *j* denotes the index of element for any vector; $rand_j$ is a uniform random number lying between 0 and 1; *Cr* is the crossover probability, which is needed to be determined by the user. *rnb(i)* is a randomly chosen index of $\{1,2,...,NP\}$ which guarantees that at least one parameter from the mutant vector ($V_{j,i,g+1}$) is copied to the trial vector ($U_{j,i,g+1}$).

3.4. Selection

In this stage, the trial vector is compared to the target vector. If the trial vector can yield a lower objective function value than its parent, then the trial vector replaces the position of the target vector. The selection operator is expressed as follows:

$$X_{i,g+1} = \begin{cases} U_{i,g} & if \ \ f(U_{i,g}) \leq f(X_{i,g}) \\ X_{i,g} & if \ \ f(U_{i,g}) > f(X_{i,g}) \end{cases}, \qquad (11)$$

where: $X_{i,g}$ denotes the target vector in generation *g*; $X_{i,g+1}$ represents the target vector in the next generation *g*+1; $U_{i,g}$ is the trial vector.

3.5. Stopping criterion verification

The optimization process terminates when the stopping criterion is met. The type of this condition can be specified by users. Commonly, maximum generation (G_{max}) or maximum number of function evaluations (*NFE*) can be used as the stopping condition. When the optimization process terminates, the final optimal solution is readily presented.

4. Evolutionary Least Squares Support Vector Machine Inference Model for Groutability Prediction (ELSIM-GP)

This section describes the proposed model, named as ELSIM-GP, in detail. The model (Fig. 3) is established by a fusion of LS-SVM and DE. ELSIM-GP employs LS-SVM as the supervised learning algorithm for learning the decision boundary for carrying out classification task. Furthermore, the model incorporates the DE for automatically identifying the optimal values of tuning parameters. The construction of the prediction model is dependent on two tuning parameters: the regularization parameter (γ) and the RBF kernel parameter (σ).
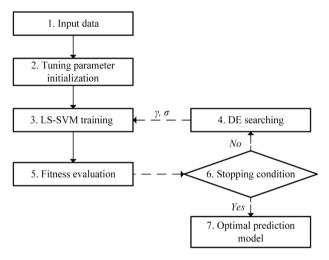
Fig. 3. Evolutionary Least Squares Support Vector Machine Inference Model for Groutability Prediction (ELSIM-GP)

(1) *Input Data*: The historical database used in this article contains 240 on-site permeation grouting data samples collected by Liao *et al.* (2011). 192 data cases are used for training (80%) and validating (20%). Meanwhile, 48 data cases are used for testing. All of the grouting activities were executed in the cities of Taipei and Kaohsiung, Taiwan. A mixture of microfine cement and micro-slag in equal proportions was utilized as the injected grout. The diameters through which 95%, 90%, and 85% of the total grout passes are 7.4 μm, 6.4 μm, and 4.5 μm, respectively. Moreover, the diameter through which 70% of the total grout passes is less than 1 μm. Thus, the grout is considered to be semi-nanometer material.

According to the previous research (Liao *et al.* 2011; Tekin, Akbas 2011), utilizing only the grain-size ratio of the soil to predict the groutability is not able of completely describing the behavior of the grouting mechanism. Furthermore, experimental results have indicated that using other parameters, including the w/c, the *e*, the *FC*, the C_z and the C_u can deliver superior prediction performance in both training and testing cases (Liao *et al.* 2011). Hence, in the current study, seven influencing factors (Table 1) are considered to determine the outcome of the grouting activity.

Table 1. Influencing factors (IF) of historical data

Factors	Description	Notation
IF1	The diameter through which 10% of the total soil mass passes	D_{10} (μm)
IF2	The diameter through which 15% of the total soil mass passes	D_{15} (μm)
IF3	Void ratio	e
IF4	The fines content of the total soil mass	FC (%)
IF5	The coefficient of gradation	C_z
IF6	The coefficient of uniformity	C_u
IF7	Water-to-cement ratio of grout	w/c

Table 2. Descriptive statistics of historical data

Factors	Notation	Max.	Min.	Mean	Std. Dev.
IF1	D_{10} (μm)	85	0.1	20.07	20.83
IF2	D_{15} (μm)	125	0.2	32.01	27.54
IF3	e	1.04	0.35	0.71	0.13
IF4	FC (%)	99.6	6.9	41.67	29.94
IF5	C_z	27.84	0.02	2.44	2.42
IF6	C_u	581.82	2.11	26.38	63.12
IF7	w/c	4.65	3.34	4[*]	0.53

[*] Median.

Moreover, in this research, the grout size is not taken into account as an influencing factor. It is because previous researches have demonstrated that the performance of AI based approaches is not affected by the size of the grouts (Liao *et al.* 2011; Tekin, Akbas 2011). Furthermore, for each data case, the corresponding output is either +1, which means that the grouting is successful, or −1, which indicates unsuccessful grouting. Table 2 provides descriptive statistics of the influencing factors of the historical data.

Before the training the model, the data set has been normalized into a (0, 1) range which helps prevent the situation in which inputs with greater magnitudes dominate those with smaller magnitudes. The historical data is illustrated in Table 3. The function used for normalizing data is shown as follows:

$$X_n = \frac{X_o - X_{\min}}{X_{\max} - X_{\min}}, \qquad (12)$$

where: X_n is the normalized data; X_o is the original data; $X_{\max}$ and $X_{\min}$ denote the maximum and minimum values of the data, respectively.

Table 3. Historical data for groutability prediction

Case	IF1	IF2	IF3	IF4	IF5	IF6	IF7	Y[*]
1	0.03	0.03	0.03	0.04	0.61	0.78	1.00	−1
2	0.02	0.02	0.04	0.02	0.74	0.97	1.00	−1
3	0.07	0.09	0.03	0.09	0.49	0.35	0.50	−1
...	...	...	...	...	...	...	...	...
238	0.11	0.14	0.01	0.18	0.61	0.39	0.50	−1
239	0.25	0.25	0.01	0.08	0.42	0.22	0.50	+1
240	0.14	0.22	0.01	0.14	0.51	0.25	0.50	+1

[*] Desired outputs.

(2) *Tuning Parameter Initialization*: The aforementioned tuning parameters of the model are randomly generated within the range of lower and upper boundaries (Table 4).

Table 4. Ranges of ELSIM-GP's tuning parameters

Tuning parameter	Notation	LB[*]	UB[**]
Regularization parameter	γ	0.001	10000
Kernel function parameter	σ	0.001	1000

[*] Lower bound; [**] Upper bound.

(3) *LSSVM Training*: In this step, LS-SVM is deployed to learn the decision boundary to separate input data into two classes of groutability (–1 and +1).

(4) *DE searching*: At each generation, the optimizer carries out the mutation, crossover, and selection processes to guide the population to the optimal solution.

(5) *Fitness evaluation*: In ELSIM-GP, in order to determine the optimal set of tuning parameters, the following objective function is used in the step of fitness function evaluation:

$$F_{fitness} = \frac{1}{R_{TR} + R_{VA}}, \qquad (13)$$

where: R_{TR} denotes the accuracy rate of classification for training set; R_{VA} represents the accuracy rate of classification for validating set. The accuracy rate of classification is calculated as the number of correct classification divided by the number of all data instances within a data set.

(6) *Stopping condition*: The DE's optimization process terminates when the maximum number of generation is achieved.

(7) *Optimal prediction model*: When the program terminates, the optimal set of tuning parameters has been successfully identified. The ELSIM-GP is ready to predict new input patterns.

5. Experimental result

As stated earlier, ELSIM-GP uses 192 data cases for training as well as validating and 48 data cases for testing. This means that 20% of historical data is reserved for testing process. However, due to the randomness in selecting testing cases, the evaluation of model error can be biased (Bishop 2006). To avoid such issue, the whole dataset is divided into five subsamples in which each subsample in turn serves as testing cases; and the model performance can be appraised via average predictive results of the five subsamples.

The process depicted above is the *k*-fold cross validation which is commonly used for model selection. In this process, the value of *k* is an unfixed parameter and it depends on the size of the data set at hand (Geisser 1993). For a large data set, even a small value of *k* (e.g. 3) can bring about relatively accurate result. Reversely, for a very sparse data set, a leave-one-out validation should be utilized in order to have as many training data as possible.

In general, with a large number of data folds, we can obtain a very good estimate of model performance. Nevertheless, the computational expense can be very costly especially for hybrid intelligence model like ELSIM-GP. Thus, it is beneficial to choose a value of *k* that can both estimate the model performance fairly and necessitate an acceptable computing effort. Arlot (2010) suggested that the number of folds should be between 5 and 10 because the statistical performance does not improve much for a larger one. Additionally, previous studies showed that 5-fold cross validation can bring about reliable estimates of prediction error (Breiman *et al.* 1984; Burman 1989).

Thus, considering the efficiency and the effectivity in evaluating model prediction performance, 5-fold cross validation is utilized. The whole data set, which includes 240 instances, is randomly separated into five folds. In each run, one data fold serves as testing set and the rest of the data folds are used for training the model (Fig. 4). Since all of the subsamples are mutually exclusive, this approach can diminish the bias in model assessment and can estimate the generalization property of each model.

After the training process, the proposed model, ELSIM-GP, is utilized to predict new input patterns from the testing data set. Furthermore, in order to verify the capability of ELSIM-GP, its performance is compared to results obtained from other benchmark approaches: Classification and Regression Trees (CART), Back-propagation Neural Network (BPNN), and Radial Basis Function Neural Network (RBFNN).

In the learning process, CART (Breiman *et al.* 1984; Loh 2011) dissipates the data into two subsets so that the records within each subset are more homogeneous than in the previous subset. Therefore, the algorithm operates by choosing a split at each node such that each sub-node created by the split is more pure than its parent node (Bevilacqua *et al.* 2003). Herein, purity refers to similarity of values of the target field. CART measures the impurity of a split at a node by defining an impurity measure. Since groutability prediction is a classification problem, Gini's diversity index is chosen in this study for measuring impurity. Another parameter that controls the tree structure is the tree depth. This parameter can be determined via the pruning algorithm (Timofeev 2004; Vega *et al.* 2009).

When using BPNN, it is needed to specify the number of hidden layers, the number of neurons in the hidden layer, and the learning rate (Samarasinghe 2006).
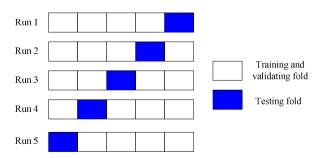

Fig. 4. Illustration of 5-fold cross validation

These parameters of BPNN are selected via repetitive trial-and-error processes. The network configuration is described as follows: the number of hidden layers is set to be 1; the number of neurons in the hidden layer is 14; and the learning rate is 1; and the number of training epochs is selected to be 2000.

Table 5. Comparison of classification accuracy (%)

Model	Fold	1	2	3	4	5	Avg.
ELSIM-GP	Train	95.3	97.9	97.9	93.2	94.3	95.7
	Test	97.9	93.8	93.8	93.8	95.8	95.0
RBF-NN	Train	96.4	94.3	93.2	94.8	94.8	94.7
	Test	91.7	85.4	85.4	95.8	95.8	90.8
CART	Train	98.4	96.4	97.9	96.4	96.9	97.2
	Test	91.7	89.6	87.5	91.7	91.7	90.4
BPNN	Train	93.8	92.7	93.8	93.2	91.2	92.9
	Test	89.6	91.7	87.5	93.8	85.4	89.6

For RBFNN, there are several parameters needed to be specified: the number of center points, the center locations, and the standard deviation of the RBF. Herein, the utilized RBF is the Gaussian function. In our study, at the first stage, the orthogonal least squares (OLS) method is used to determine the number and the locations of centers (Chen *et al.* 1991). The second stage requires the determination of the network's weights which is achieved by solving a linear system (Liao *et al.* 2011).

Table 5 provides the prediction results for ELSIM-GP as well as other approaches. Average prediction outcomes of RBFNN (90.83%), BPNN (89.58%), and CART (90.42%) are less accurate than that of ELSIM-GP (95%). CART seems to suffer the problem of over-fitting since it fits the training set very well, but performs poorly on the new data set. Noticeably, all of the three benchmark models have testing results in which classification rates are below 90%. This, to some degree, indicates unstable performances of these models.

Meanwhile, the average classification accuracy acquired from ELSIM-GP for training and testing sets are 95.73% and 95%, respectively. It can be observed that the proposed model has successfully overcome the issue of over-fitting since it yields relatively balanced performances between training and testing data sets. Notably, in the third fold, the accuracy rate for testing data can reach roughly 98%. Furthermore, predictive results obtained by ELSIM-GP of all data folds surpass 90%. These results demonstrate that the newly proposed approach is capable of delivering accurate predictive performance. The classification result of ELSIM-GP for one testing data fold is shown in the Table 6.

Conclusions

This paper has presented a new prediction model, named as ELSIM-GP, to assist construction engineers in assessing the possibility of a grouting process which utilizes micro-fine cement. The proposed model was developed by a fusion of LS-SVM and DE. ELSIM-GP utilizes LS-SVM to classify high dimensional input data so that the model can make prediction whenever a new input pattern is available. Meanwhile, DE searching algorithm is implemented to identify the most appropriate tuning parameters.

Table 6. Classification result of ELSIM-GP

	IF1	IF2	IF3	IF4	IF5	IF6	IF7	Y_A	Y_P
1	0.22	0.24	0.01	0.08	0.55	0.26	0.00	+1	+1
2	0.18	0.22	0.02	0.09	0.41	0.22	1.00	+1	+1
3	0.21	0.28	0.02	0.07	0.42	0.18	0.50	+1	+1
4	0.65	0.96	0.01	0.09	0.17	0.05	1.00	+1	+1
5	0.35	0.47	0.01	0.06	0.39	0.14	0.50	+1	+1
6	0.11	0.13	0.02	0.13	0.57	0.28	1.00	+1	+1
7	0.03	0.03	0.03	0.04	0.61	0.78	1.00	−1	−1
8	0.21	0.30	0.01	0.12	0.38	0.19	1.00	+1	+1
9	0.92	0.70	0.00	0.02	0.45	0.01	1.00	+1	+1
10	0.03	0.06	0.28	0.13	0.46	0.26	1.00	−1	−1
11	0.33	0.33	0.01	0.06	0.48	0.17	1.00	+1	+1
12	0.07	0.09	0.03	0.10	0.59	0.32	1.00	−1	−1
13	0.01	0.01	0.06	0.02	0.72	0.95	0.50	−1	−1
14	0.03	0.07	0.13	0.22	0.48	0.30	0.00	−1	−1
15	0.01	0.04	0.19	0.47	0.36	0.38	0.00	−1	−1
16	0.21	0.26	0.01	0.12	0.46	0.22	0.50	+1	+1
17	0.27	0.33	0.01	0.09	0.42	0.19	0.50	+1	+1
18	0.01	0.01	0.14	0.03	0.88	0.98	1.00	−1	−1
19	0.94	0.86	0.01	0.04	0.32	0.02	1.00	+1	+1
20	0.08	0.09	0.02	0.09	0.57	0.34	0.50	−1	−1
21	0.12	0.15	0.01	0.24	0.51	0.44	1.00	−1	−1
22	0.98	0.84	0.01	0.03	0.32	0.00	0.50	+1	+1
23	0.06	0.06	0.02	0.11	0.43	0.63	1.00	−1	−1
24	0.22	0.25	0.01	0.10	0.54	0.20	0.50	+1	+1

Continued Table 6

	IF1	IF2	IF3	IF4	IF5	IF6	IF7	Y_A	Y_P
25	0.02	0.02	0.02	0.01	0.90	0.99	1.00	−1	−1
26	0.02	0.02	0.07	0.05	0.32	0.86	0.50	−1	−1
27	0.01	0.01	0.08	0.01	0.86	0.99	1.00	−1	−1
28	0.56	0.57	0.01	0.05	0.41	0.10	0.50	+1	+1
29	0.15	0.23	0.01	0.11	0.55	0.26	0.50	+1	+1
30	0.72	0.64	0.01	0.03	0.39	0.07	0.50	+1	+1
31	0.28	0.39	0.00	0.09	0.39	0.19	0.50	+1	+1
32	0.15	0.22	0.01	0.14	0.52	0.22	1.00	+1	+1
33	0.59	0.61	0.00	0.04	0.33	0.09	0.50	+1	+1
34	0.01	0.01	0.10	0.02	0.93	0.99	0.50	−1	−1
35	0.86	0.66	0.00	0.02	0.33	0.04	0.50	+1	+1
36	0.60	0.56	0.00	0.03	0.43	0.11	1.00	+1	+1
37	0.11	0.14	0.01	0.18	0.61	0.39	0.50	−1	−1
38	0.84	0.72	0.01	0.03	0.29	0.04	1.00	+1	+1
39	0.56	0.58	0.01	0.04	0.36	0.09	0.50	+1	+1
40	0.08	0.06	0.02	0.12	0.59	0.63	1.00	−1	−1
41	0.16	0.25	0.01	0.14	0.49	0.21	0.50	+1	+1
42	0.40	0.50	0.00	0.06	0.25	0.13	0.50	+1	+1
43	0.03	0.02	0.02	0.01	0.87	0.98	1.00	−1	−1
44	0.15	0.22	0.01	0.14	0.58	0.32	0.50	+1	+1
45	0.19	0.24	0.01	0.08	0.54	0.23	0.50	+1	+1
46	0.08	0.09	0.02	0.18	0.55	0.31	0.50	+1	−1
47	0.22	0.27	0.01	0.11	0.48	0.20	1.00	+1	+1
48	0.85	0.70	0.01	0.02	0.27	0.04	0.50	+1	+1

Note: Y_A – actual output; Y_P – predicted output.

Since ELSIM-GP is a hybrid intelligence model, the approach can be quite complex for practical engineers. However, considering that permeation grouting is a complicated process and predicting its outcome is by no means an easy task. Therefore, it is very challenging to construct a simple model that yields highly accurate forecasting performance.

Although the proposed model is complicated to establish, ELSIM-GP has the advantage of operating autonomously because the model does not require any expertise in parameter setting. Model tuning parameters are all determined by DE. Hence, with more effort on software engineering, a user-friendly interface can be integrated into the model, and this enables ELSIM-GP to be a promising tool for practical engineers for dealing with groutability prediction problem.

In the current study, the soil type which is subjected to permeation grouting is sandy silt soil. Although simulation results have demonstrated that ELSIM-GP can deliver superior forecasting accuracy for this type of soil, more historical cases in which the targeted objects of the grouting process involve different soil types should be incorporated to enhance the generalization of the prediction model.

Additionally, if another type of microfine grout is applied, the characteristics of the grouting process can be changed. Hence, the current inference model may encounter difficulty in groutability forecasting. The reason is that geotechnical engineering is inherently uncertain and highly complex. Thus, it is required to collect new observations and to construct new prediction models by training processes. Nevertheless, the procedure of collecting new data cases are of great effort and time-consuming. Hence, we would like to consider this to be a promising future research direction.

References

Akbulut, S.; Saglamer, A. 2002. Estimating the groutability of granular soils: a new approach, *Tunnelling and Underground Space Technology* 17(4): 371–80. http://dx.doi.org/10.1016/S0886-7798(02)00040-8

Arlot, S. 2010. A survey of cross-validation procedures for model selection, *Statistics Surveys* 4: 40–79. http://dx.doi.org/10.1214/09-ss054

Bao, Y. K.; Tao, Z.; Guo, L. L.; Wang, W. 2005. Forecasting stock composite index by fuzzy Support Vector Machine regression, in *Proc. of the 4th International Conference on Machine Learning and Cybernetics,* 18–21 August 2005, Guangzhou, China, 8747489.

Bevilacqua, M.; Braglia, M.; Montanari, R. 2003. The classification and regression tree approach to pump failure rate analysis, *Reliability Engineering and System Safety* 79(1): 59–67. http://dx.doi.org/10.1016/s0951-8320(02)00180-1

Bishop, C. 1995. *Neural networks for pattern recognition.* Oxford: Clarendon Press. 482 p.

Bishop, C. 2006. *Pattern recognition and machine learning.* Singapore: Springer Science+Business Media. 738 p.

Breiman, L.; Friedman, J.; Olshen, R.; Stone, C. J. 1984. *Classification and regression trees.* London: Chapman and Hall/CRC. 358 p.

Brown, D. E.; Corruble, V.; Pittard, C. L. 1993. A comparison of decision tree classifiers with backpropagation neural networks for multimodal classification problems, *Pattern Recognition* 26(6): 953–961. http://dx.doi.org/10.1016/0031-3203(93)90060-A

Burman, P. 1989. A comparative study of ordinary cross-validation, *v*-fold cross-validation and the repeated learning-testing methods, *Biometrika* 76(3): 503–14. http://dx.doi.org/10.1093/biomet/76.3.503

Butron, C.; Gustafson, G.; Fransson, A.; Funehag, J. 2009. Drip sealing of tunnels in hard rock: a new concept for the design and evaluation of permeation grouting, *Tunnelling and Underground Space Technology* 25(2): 114–121. http://dx.doi.org/10.1016/j.tust.2009.09.008

Chen, S.; Cowan, C. F. N.; Grant, P. M. 1991. Orthogonal least squares learning algorithm for radial basis function networks, *IEEE Transactions on Neural Networks* 2(2): 302–309. http://dx.doi.org/10.1109/72.80341

Chen, Y.-L.; Azzam, R.; Fernandez-Steeger, T.; Li, L. 2009. Studies on construction pre-control of a connection aisle between two neighbouring tunnels in shanghai by means of 3D FEM, neural networks and fuzzy logic, *Geotechnical and Geological Engineering* 27(1): 155–167. http://dx.doi.org/10.1007/s10706-008-9220-5

Cheng, M.-Y.; Hoang, N.-D.; Roy, A. F. V.; Wu, Y.-W. 2011. A novel time-depended evolutionary fuzzy SVM inference model for estimating construction project at completion, *Engineering Applications of Artificial Intelligence* 25(4): 744–752. http://dx.doi.org/10.1016/j.engappai.2011.09.022

Cheng, M.-Y.; Roy, A. F. V. 2011. Evolutionary fuzzy decision model for cash flow prediction using time-dependent support vector machines, *International Journal of Project Management* 29(1): 56–65. http://dx.doi.org/10.1016/j.ijproman.2010.01.004

Cheng, M.-Y.; Tsai, H.-C.; Sudjono, E. 2010. Evolutionary fuzzy hybrid neural network for project cash flow control, *Engineering Applications of Artificial Intelligence* 23(4): 604–613. http://dx.doi.org/10.1016/j.engappai.2009.10.003

Cheng, M.-Y.; Wu, Y.-W. 2009. Evolutionary support vector machine inference system for construction management, *Automation in Construction* 18(5): 597–604. http://dx.doi.org/10.1016/j.autcon.2008.12.002

De Brabanter, K.; Karsmakers, P.; Ojeda, F.; Alzate, C.; De Brabanter, J.; Pelckmans, K.; De Moor, B.; Vandewalle, J.; Suykens, J. A. K. 2010. *LS-SVMlab toolbox user's guide version 1.8*. Internal Report 10–146, ESAT-SISTA, K. U. Leuven, Belgium.

Geisser, S. 1993. *Predictive inference*. New York: Chapman and Hall/CRC. 240 p.

Kalinli, A.; Acar, M. C.; Gunduz, Z. 2011. New approaches to determine the ultimate bearing capacity of shallow foundations based on artificial neural networks and ant colony optimization, *Engineering Geology* 117(1–2): 29–38. http://dx.doi.org/10.1016/j.enggeo.2010.10.002

Kiranyaz, S.; Ince, T.; Yildirim, A.; Gabbouj, M. 2009. Evolutionary artificial neural networks by multi-dimensional particle swarm optimization, *Neural Networks* 22(10): 1448–1462. http://dx.doi.org/10.1016/j.neunet.2009.05.013

Yu, L.; Chen, H.; Wang, S.; Lai, K. K. 2009. Evolving least squares support vector machines for stock market trend mining, *IEEE Transactions on Evolutionary Computation* 13(1): 87–102. http://dx.doi.org/10.1109/TEVC.2008.928176

Liao, K.-W.; Fan, J.-C.; Huang, C.-L. 2011. An artificial neural network for groutability prediction of permeation grouting with microfine cement grouts, *Computers and Geotechnics* 38(8): 978–986. http://dx.doi.org/10.1016/j.compgeo.2011.07.008

Loh, W.-Y. 2011. Classification and regression tree methods, *WIREs Data Mining and Knowledge Discovery* 1(1): 14–23. http://dx.doi.org/10.1002/widm.8

Olson, D. L.; Delen, D.; Meng, Y. 2011. Comparative analysis of data mining methods for bankruptcy prediction, *Decision Support Systems* 52(2): 464–473. http://dx.doi.org/10.1016/j.dss.2011.10.007

Ozgurel, H. G.; Cumaraswamy, V. 2005. Effect of grain size and distribution on permeability and mechanical behavior of acrylamide grouted sand, *Journal of Geotechnical and Geoenvironmental Engineering* 131(12): 1457–1465. http://dx.doi.org/(ASCE)1090-0241(2005)131:12(1457)

Pendharkar, P. C. 2011. A hybrid radial basis function and data envelopment analysis neural network for classification, *Computers & Operations Research* 38(1): 256–266. http://dx.doi.org/10.1016/j.cor.2010.05.001

Perret, S.; Khayat, K. H.; Gagnon, E.; Rhazi, J. 2002. Repair of 130-year old masonry bridge using high-performance cement grout, *Journal of Bridge Engineering* 7(1): 31–38. http://dx.doi.org/10.1061/(ASCE)1084-0702(2002)7:1(31)

Perret, S.; Palardy, D.; Ballivy, G. 2000. Rheological behavior and setting time of microfine cement-based grouts, *Materials Journal* 97(4): 472–478.

Price, K. V.; Storn, R. M.; Lampinen, J. A. 2005. *Differential evolution: a practical approach to global optimization*. Berlin Heidelberg: Springer-Verlag. 538 p.

Samarasinghe, S. 2006. *Neural networks for applied sciences and engineering*. Taylor & Francis Group, LLC. 570 p.

Storn, R.; Price, K. 1997. Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces, *Journal of Global Optimization* 11(4): 341–359. http://dx.doi.org/10.1023/A:1008202821328

Suykens, J. A. K.; Vandewalle, J. 1999. Least squares support vector machine classifiers, *Neural Processing Letters* 9(3): 293–300. http://dx.doi.org/10.1023/A:1018628609742

Suykens, J.; Gestel, J. V.; Brabanter, J. D.; Moor, B. D.; Vandewalle, J. 2002. *Least square support vector machines*. Singapore: World Scientific Publishing Co. Pte. Ltd. 84 p.

Tekin, E.; Akbas, S. 2011. Artificial neural networks approach for estimating the groutability of granular soils with cement-based grouts, *Bulletin of Engineering Geology and the Environment* 70(1): 153–161. http://dx.doi.org/10.1007/s10064-010-0295-x

Timofeev, R. 2004. *Classification and regression trees (CART)*: *theory and applications*: Master Thesis. Center of Applied Statistics and Economics, Humboldt University, Berlin.

Vega, F. A.; Matías, J. M.; Andrade, M. L.; Reigosa, M. J.; Covelo, E. F. 2009. Classification and regression trees (CARTs) for modelling the sorption and retention of heavy metals by soil, *Journal of Hazardous Materials* 167(1–3): 615–624. http://dx.doi.org/10.1016/j.jhazmat.2009.01.016

Wang, H.; Hu, D. 2005. Comparison of SVM and LS-SVM for Regression, in *Proc. of the International Conference on Neural Networks and Brain (ICNN&B '05)*, 13–15 October 2005, Beijing, China, 9072845. http://dx.doi.org/10.1109/ICNNB.2005.1614615

Wong, B. K.; Bodnovich, T. A.; Selvi, Y. 1997. Neural network applications in business: a review and analysis of the literature (1988–1995), *Decision Support Systems* 19(4): 301–320. http://dx.doi.org/10.1016/s0167-9236(96)00070-x

Wu, J.-D.; Liu, J.-C. 2012. A forecasting system for car fuel consumption using a radial basis function neural network, *Expert Systems with Applications* 39(2): 1883–1888. http://dx.doi.org/10.1016/j.eswa.2011.07.139

Yu, L. 2011. An evolutionary programming based asymmetric weighted least squares support vector machine ensemble learning methodology for software repository mining, *Information Sciences* 191: 31–46. http://dx.doi.org/10.1016/j.ins.2011.09.034

Zebovitz, S.; Krizek, R. J.; Atmatzidis, D. K. 1989. Injection of fine sands with very fine cement grout, *Journal of Geotechnical Engineering* 115(12): 1717–1733. http://dx.doi.org/10.1061/(ASCE)0733-9410(1989)115:12(1717))

Zhou, J.; Shi, J.; Li, G. 2011. Fine tuning support vector machines for short-term wind speed forecasting, *Energy Conversion and Management* 52(4): 1990–1998. http://dx.doi.org/10.1016/j.enconman.2010.11.007

Min-Yuan CHENG. He is currently a Professor at the Department of Civil and Construction Engineering, National Taiwan University of Science and Technology. He holds lectures in Construction Automation and Construction Process Reengineering. He has published many papers in various international journals such as Journal of Civil Engineering and Management, Automation in Construction, Journal of Construction Engineering and Management, and Expert Systems with Applications. His research interests include management information system, applications of artificial intelligence, and construction management process reengineering.

Nhat-Duc HOANG. He is currently a researcher and lecturer at Center of Research and Development, Duy Tan University. He got the MSc. and PhD. degrees at National Taiwan University of Science and Technology. He has published papers in various international journals such as Journal of Civil Engineering and Management, Automation in Construction, and Journal of Computing in Civil Engineering. His research focuses on applications of Artificial Intelligence in construction engineering and management.