TIME DEPENDENT EVOLUTIONARY FUZZY SUPPORT VECTOR MACHINE INFERENCE MODEL FOR PREDICTING DIAPHRAGM WALL DEFLECTION

Andreas F. V. Roy[*], *Min-Yuan Cheng*[†], *Yu-Wei Wu*[†]

Abstract: Brace diaphragm walls are commonly used in underground structures in metropolitan areas, where avoiding costly damage to adjacent infrastructure / buildings is critical to project success. It is necessary to make accurate diaphragm wall deflection predictions to ensure actual deflection falls within allowable limits, and thus ensure the safety of both the project and adjacent structures. Numerous studies and approaches, such as empirical, semi-empirical as well as numerical approaches, have addressed excavation-induced deflection in diaphragm walls. Artificial intelligence (AI) has been used recently by several researchers to improve diaphragm wall deflection prediction capabilities. This paper proposes a hybrid artificial intelligence system, namely the evolutionary fuzzy support vector machine inference model for time series data (EFSIM$_T$), to predict diaphragm wall deflection in deep excavation through the application of historical project data. Simulations were performed on 1,083 instances, segregated into a total of 988 training data sets and 95 test data sets. Validation results show that the EFSIM$_T$ achieves higher performance in comparison with Artificial Neural Networks and the Evolutionary Support Vector Machine Inference Model (ESIM). Therefore, EFSIM$_T$ has great potential as a predictive tool for diaphragm wall deflection problems and assisting project managers/engineers to ensure safety during the construction process.

Key words: *Diaphragm wall deflection, deep excavation, fuzzy logic, time series data, weighted support vector machines, fast messy genetic algorithms*

Received: June 9, 2013 **DOI:** 10.14311/NNW.2014.24.012
Revised and accepted: April 14, 2014

[*]A. F. V. Roy – Corresponding Author, Dept. of Civil Engineering, Parahyangan Catholic University, Jl. Ciumbuleuit 94 Bandung, 40141, West Java, Indonesia, E-mail: `andrevan@unpar.ac.id, afvroy21@gmail.com`

[†]Min-Yuan Cheng, Yu-Wei Wu, Dept. of Construction Engineering, National Taiwan University of Science and Technology, #43, Sec. 4, Keelung Rd., Taipei, 106, Taiwan, R.O.C., E-mail: `myc@mail.ntust.edu.tw, d9305503@mail.ntust.edu.tw`

1. Introduction

Brace diaphragm walls are underground structures commonly used in dense urban areas with existing infrastructure / buildings. Brace diaphragm technology is employed primarily in retaining structures designed to support deep excavations and protect adjacent structures and buildings by minimizing ground deformation. In order to prevent costly damage, builders try to avoid excessive diaphragm wall deflection and excavation-induced ground settlement. It is necessary to make accurate diaphragm wall deflection predictions to ensure actual deflection falls within allowable limits, and this ensure the safety of both the project and adjacent structures.

Many studies on diaphragm wall deflection prediction have been conducted, adopting empirical, semi-empirical as well as numerical approaches [1-4]. While numerical models (e.g., the finite element method [FEM]) have been most commonly employed, their predictions can differ significantly from actual field measurements and case histories [5]. Moreover [5] reported that adjusting the properties of certain models could enhance predictive accuracy. However, the proposed approach was not systematic and most often could not be applied to similar problems. Furthermore, [6,7] found that FEM analysis depends heavily on the constitutive behavior of soil, soil parameters generally obtained from laboratory tests, which are inadequate to represent actual soil conditions and diverse construction effects.

As empirical and numerical approaches have inherent drawbacks, several researchers tried to seek more reliable alternatives. Recently, there has been growing interest in using artificial neural networks (ANNs) as an artificial intelligence (AI) technique due to their excellent performance modeling nonlinear relationships and the dependent relationships shown between variables. Goh et al. [8] demonstrated that ANNs are able to synthesize data derived from FEM on braced excavations in order to capture nonlinear relationships between variables and predict wall displacement to a reasonable level of accuracy. Jan et al. [6] employed ANNs to predict diaphragm wall deflection in 18 historical cases from Taipei, each of which covered between four and seven excavation stages and contained seven distinct input variables. Moreover, Chua and Goh [9] fused ANNs, a Bayesian framework and genetic algorithm (GA) to form evolutionary Bayesian back-propagation (EBBP) as an expanded approach proposed in [8]. This hybrid approach was taken in order to handle "overfitting" problems inherent in ANNs as data and input variables numbers increased and contained "noise".

Beside thes "overfitting" problem ANNs present several additional drawbacks [10, 11], therefore numerous studies have been conducted to overcome such drawbacks. For example, the Structural Genetic Trained Neural Network (SGTNN) proposed in [10], combines ANNs and GA. Moreover, continued development of new AI techniques has expanded research into their adaptation and utilization to solve geotechnical engineering problems. This paper focuses on predicting deep excavation diaphragm wall deflection using the Evolutionary Fuzzy Support Vector Machine Inference Model for Time Series Data (EFSIM$_T$), a fusion of Fuzzy Logic (FL), weighted Support Vector Machines (weighted SVMs) and a fast messy genetic algorithm (FMGA). In EFSIM$_T$, FL handles vagueness and uncertainty. Moreover, FL is also used as a fuzzy inference mechanism. Weighted SVMs in the

$EFSIM_T$handle fuzzy input-output mapping and focus on time series data characteristics inherent to the diaphragm wall deflection database and FMGA is deployed as an optimization tool to handle FL and weighted SVMs search parameters. This study applied diaphragm wall data compiled previously from 18 historical deep excavation project cases located in metropolitan Taipei with the intent of obtaining a level of prediction result accuracy higher than both [6], which employed only ANNs, and [12], which applied ESIM. Therefore, the performance of the proposed system was compared with [6] and [12] results.

2. Overview of Fuzzy Logic, Weighted Support Vector Machines and Fast Messy Genetic Algorithm

2.1 Fuzzy Logic

Fuzzy Logic (FL) a popular AI technique invented by Zadeh in the 1960s, has been used in forecasting, decision making and action control in environments characterized by uncertainty, vagueness, presumptions and subjectivity [13]. FL simulates the human decision-making process by employing approximate reasoning logic [14]. Heshmaty and Kandel [15] expressed that FL provides a more realistic approach than that used by traditional mathematical models to address phenomena in nature characterized by vagueness and uncertainty.

FL consists of a set of rules that relates a set of inputs to a set of outputs. Quantitative relationships are established through a membership function (MFs) between actual variable values and the qualitative and linguistic variables used in 'if-then' rules. Therefore, linguistic variables described by MFs and fuzzy if-then rules play an essential role in fuzzy logic applications [16].

FL consists of four major components: fuzzification, rule base, inference engine and defuzzification. Fuzzification is a process that uses MFs to convert the value of input variables into corresponding linguistic variables. The result, which is used by the inference engine, stimulates the human decision-making process based on fuzzy implications and available rules. In the final step, the fuzzy set, as the output of the inference process, is converted into crisp output. This process, which reverses fuzzification, is called defuzzification [17].

Despite the advantages of FL, the approach presents a number of problems; including identifying appropriate MFs and number of rules for application. This process is subjective in nature and reflects the context in which a problem is viewed. Ko [18] found that the more complex the problem, the more difficult the construction of MFs and the fuzzy rules. Such shortcomings are seen by some researchers as optimization problems, as determining MFs configurations and fuzzy rules is complicated and problem oriented. To overcome remaining difficulties, some researchers have tried to fuse FL with AI optimization techniques such as simple genetic algorithms (SGAs) and ant colony [19, 20]. These optimization methods have demonstrated their ability to minimize time-consuming operations and the level of human intervention necessary to optimize MFs and fuzzy rules.

2.2 Weighted Support Vector Machines

Weighted Support Vector Machines are also known as Fuzzy Support Vector Machines (FSVMs); a name proposed in [21] as weight is effectively the fuzzy membership addressed for each training point. In this paper, to avoid confusion with the FL technique, the term "WSVMs" is used. FSVMs were developed in [22] to enhance support vector machines (SVMs) abilities to reduce the effect of outliers and noise in data points. While SVMs theory has been demonstrated to be very powerful in solving classification problems [23], it has certain drawbacks. For example, SVMs treat all training points of a given class uniformly, however in many real world applications, not all training data points are equally important for classification purposes. To solve this problem, [22] applied a fuzzy member to each input point in SVMs, thus allowing different input points to contribute differently to the learning decision surface. In such time series prediction problems, the older training points are associated with lower weights so that the effect of older training points can be reduced when the regression function is optimized.

Given a set S of labeled training data points associated with weights

$$(y_1, x_1, s_1), \ldots, (y_m, x_m, s_m) \tag{1}$$

where $x_i \in R^n$ is the input vector, $y_i \in R$ is the desired value and $\sigma \leq s_i \leq 1$ is a weight for $(x_i, y_i)(i = 1, \ldots, m)$ and a sufficiently small $\sigma > 0$ represents the lower bound of weighted data. The WSVMs for regression solves and optimizes:

$$\min \qquad \frac{1}{2} w.w + C \sum_{i=1}^{l} s_i(\xi_i + \xi_i*) \tag{2}$$

$$\text{subject to} \qquad \begin{cases} y_i - (w.\varphi(x_i) + b) \leq \varepsilon + \xi_i, \\ (w.\varphi(x_i) + b) - y_i \leq \varepsilon + \xi_i*, \\ \xi_i, \xi_i* \leq 0 \end{cases}$$

where C is a constant and $\varphi(x)$ is the high dimensional feature space, which is non-linearly mapped from input space x. ξ_i and ξ_i* represent upper and lower training errors, respectively. It should be noted that a smaller s_i reduces the effect of the parameter ξ_i in Eq. (2), so that the corresponding point $\varphi(x_i)$ is treated as less important.

The above optimization problem can be transformed into

$$\begin{aligned} \max W(\alpha) \quad = \quad & -\frac{1}{2} \sum_{i,j=1}^{l} (\alpha_i - \alpha_i*)(\alpha_j - \alpha_j*) K(x_i, x_j) - \varepsilon \sum_{i=1}^{l} (\alpha_i + \alpha_i*) + \\ & + \sum_{i=1}^{l} y_i(\alpha_i - \alpha_i*) \end{aligned} \tag{3}$$

subject to $\sum\limits_{i=1}^{l} y_i \alpha_i = 0$, $0 \leq \alpha_i \leq s_i C, i = 1, \ldots, l$
and the Kuhn-Tucker condition is defined as

$$\overline{\alpha_i}(\varepsilon + \overline{\xi_i} - y_i + \overline{w}.x_i + \overline{b}) = 0, i = 1, \ldots, l, \tag{4}$$

$$\overline{\alpha_i^*}(\varepsilon + \overline{\xi_i^*} + y_i - \overline{w}.x_i - \overline{b}) = 0, i = 1, \dots, l, \tag{5}$$

$$(s_i C - \overline{\alpha_i})\overline{\xi_i} = 0, i = 1, \dots, l, \tag{6}$$

$$(s_i C - \overline{\alpha_i^*})\overline{\xi_i^*} = 0, i = 1, \dots, l \tag{7}$$

Point x_i with the corresponding $\overline{\alpha_i^*} > 0$ is a support vector. The other type of support vector, with corresponding $0 \leq \overline{\alpha_i}^{(*)} \leq s_i C$, lies on the ε-insensitive tube around the decision function. The one with corresponding $\overline{\alpha_i^*} = s_i C$ is outside the tube. An important difference between SVMs and WSVMs is that the points with the same value of $\overline{\alpha_i}^{(*)}$ may indicate a different type of support vector in WSVMs due to the factor s_i [24].

$K(x_i, x_j)$ in Eq. (3) is defined as the kernel function. The value of the kernel is equal to the inner product of two vectors X_i and X_j in the feature space $\varphi(x_i)$ and $\varphi(x_j)$, that is, $K(x_i, x_j) = \varphi(x_i) * \varphi(x_j)$. The chosen kernel function must fulfill Mercer's condition, which determines whether a prospective kernel is actually an inner product in some space and guarantees that unique global optimal solutions are achieved [23]. Several admissible kernel functions used today include the polynomial kernel, radial basis function (RBF) and sigmoid kernel. However, the RBF kernel has been recommended for general users as a first choice due to its ability to analyze higher-dimension data, the use of only one hyperparameter to search, and fewer numerical difficulties [25].

In sequential learning and inference methods such as time series problems, where a point from the recent past may be given greater weight than a point from further in the past, function of time t_i can be selected as the weighted SVMs s_i scheme.

$$s_i = f(t_i) \tag{8}$$

with this scheme assuming the last point x_m as the most important and $s_m = f(t_m) = 1$ and the first point x_1 as the least important, and choosing $s_1 = f(t_1) = \sigma$ [21]. Lin and Wang [21] proposed two time functions, linear and quadratic, as shown in equations (9) and (10). Both have been used in [26] on financial time series forecasting problems, whose authors demonstrated their ability to deliver better results than SVMs.

$$s_i = f_l(t_i) = at_i + b = \frac{1-\sigma}{t_m - t_1} t_i + \frac{t_m \sigma - t_1}{t_m - t_1} \tag{9}$$

$$s_i = f_q(t_i) = a(t_i - b)^2 + c = (1-\sigma)\left(\frac{t_i - t_1}{t_m - t_1}\right)^2 + \sigma \tag{10}$$

Like SVM, using WSVMs presents the user with the problem of how to set optimal parameters, as parameter selection affects WSVMs prediction accuracy. The three parameters that must be optimized when using RBF kernels include the penalty parameter (C), kernel parameter (γ) and lower bound of weighting data parameter (σ). To overcome this drawback, an optimization technique (e.g., FMGA) may be used to identify best parameters simultaneously [12].

2.3 Fast Messy Genetic Algorithm

Fast messy genetic algorithms are a recently developed machine learning and optimization tool based on a genetic algorithm approach that can efficiently find optimal solutions for large-scale permutation problems. The latter differ from SGAs, which describe possible solutions using fixed length strings. FMGA applies messy chromosomes to form strings of various lengths [27].

The fast messy genetic algorithm was developed in [28] in 1993 as an improvement on the messy genetic algorithm (MGAs). MGAs were initially developed to overcome the SGAs linkage problem, which resulted from a parameter coding problem that could generate suboptimal solutions [29]. However, MGAs faced a problem as well. Goldberg et al. [28] proposed three modifications in order to reduce the size of the initial population as well as the MGAs execution time initialization and primordial phase. Those modifications utilize probabilistically complete initialization (PCI) instead of partially enumerative initialization (PEI), use building block filtering, and take a more conservative approach to thresholding in tournament selection.

The FMGA contains two loop types, namely the inner and outer. The process starts with the outer loop. Firstly, a competitive template, represented by a problem-specific, fixed-bit string, is generated randomly or found during the search process. Each outer loop cycle is one "era", which iterates over the order k of processed building blocks (BBs). A building block is a set of genes, a subset of strings that are short, low-order and high-performance.

With the start of each new era, the three phase operations of the inner loop, including the initialization phase, the building block filtering (BBF) or primordial phase, and the juxtapositional phase, are invoked. In the initialization phase, an adequately large population contains all possible BBs of order k. FMGA performs the PCI process at this stage, which randomly generates n chromosomes and calculates their fitness values. There are two operations in the primordial phase, namely building-block filtering and threshold selection. In the primordial phase, 'bad' genes that do not belong to BBs are filtered out, so that, in the end, the resultant population encloses a high proportion of 'good' genes belonging to BBs. In the juxtaposition phase, operations are more similar to those of SGAs. The selection procedure for good genes (BBs) is used together with a cut-and-splice operator to form a high quality generation, which may contain the optimal solution.

Once the inner loop is finished, the next outer loop begins. The competitive template is replaced by the best solution found so far, which becomes the new competitive template for the next era. The whole process is repeated until the maximum number era (k_{max}) is reached. The FMGA can also perform over "epochs". This term is used to describe a complete process that starts from a first era and continues until k_{max}. The best solution found in one complete process is passed to succeeding epochs though the competitive template. Epochs can be performed as many times as desired. The algorithm is terminated once a sufficient solution is obtained or no further improvement is made.

3. Evolutionary Fuzzy Support Vector Machine Inference Model for Time Series Data for Predicting Diaphragm Wall Deflection

3.1 The Proposed Model

The proposed EFSIM$_T$ is an alternative approach to retaining and utilizing experiential knowledge that fused three different AI techniques namely FL, WSVMs and FMGA [30]. FL deals with vagueness and approximate reasoning; WSVMs act as a supervised learning tool to handle fuzzy input-output mapping and focus on time series data characteristics; and FMGA works simultaneously for the fittest MFs, the defuzzification parameter (*dfp*) as FL parameters, and the SVM hyper-parameter (here in C and γ), the lower bound of the weighted data parameter (σ) as WSVMs parameters. Fig. 1 illustrates the EFSIM$_T$ architecture.

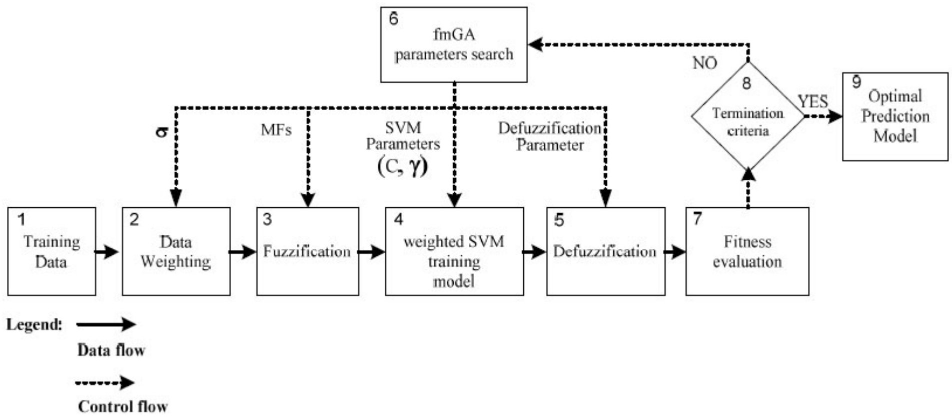


Fig. 1 *EFSIM$_T$ architecture [30].*

Below are the major steps of the proposed EFSIM$_T$ model:

1. Training Data. EFSIM$_T$ uses sequential data as training data. Sequential data reflect identified attributes, and training data are normalized into a (0, 1) range, which helps avoid attributes with greater numeric ranges dominating those with smaller numeric ranges, and also helps avoid numerical difficulties [25].

2. Data Weighting. Each training data point was weighted to the time function using either a linear or quadratic function, as shown in Equation (9) and (10). The last data point x_m as the recent data, treated as most important, and such, has a weighting value s_m of 1. The first data point x_1 as the most distant past, treated as the least important assigned a weighting values_m equal to σ. In this step, the lower bound of weighted data parameters (σ) are generated randomly in the range of $0.1 - 1$ and encoded by FMGA.

3. Fuzzification. This process converts each normalized input variable value from the first step into corresponding membership grades. The MFs are used to map normalized input variables to corresponding membership grades. The model used trapezoidal MFs and triangular MFs shapes (see Fig. 2) that, in general, can be developed by referencing summit points and widths [19, 31]. The summit and width representation method (SWRM) was applied to encode complete MF sets (see Fig. 2 (c)) [32]. Each normalized input pattern was converted to membership grades corresponding to the specific MF set generated and encoded by FMGA.

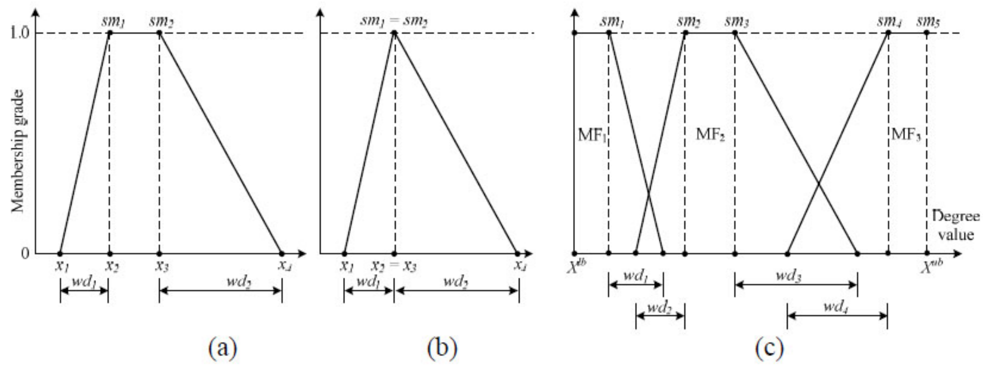


Fig. 2 *Membership function: (a) trapezoidal; (b) triangular; (c) complete MF set [18].*

4. Weighted SVM Training Model. The fuzzification process output, in the form of membership grades, is a fuzzy input for WSVMs. WSVMs train this dataset to obtain the prediction model. WSVMs use penalty (C) and kernel parameters (γ), which are generated randomly and encoded by FMGA. This study used the RBF kernel as a reasonable first choice [25]. In this research, the range value search by FMGA for C and γg is followed [12]. Moreover, to train the dataset, LIBSVM developed in [33] was embedded into the EFSIM$_T$ model. LIBSVM is currently one of the most widely used Support Vector Machine software. LIBSVM is also able to process a weight to each data instance.

5. Defuzzification. Once WSVMs have finished the training process, output numbers are expressed in terms of the fuzzy set, and must beconverted into a single real number. Employing FMGA, the EFSIM$_T$ generates a random defuzzification parameter (*dfp*) substring and encodes it to convert WSVMs output. This evolutionary approach is simple and straightforward, as it uses *dfp* as a common denominator for WSVMs output.

6. FMGA Parameter Search. The FMGA is utilized to search simultaneously for the fittest shapes of MFs, *dfp*, penalty parameter C, RBF kernel parameter γ and the lower boundary of weighted data parameter σ. FMGA works

based on the concept of genetic operations. Thus, chromosome design plays a central role in achieving objectives. The chromosome representing a possible solution for searched parameters consists of five parts: the MF substring, *dfp* substring, penalty parameter substring, kernel parameter substring and lower bound of weighting data substring. Each substring has a specific length that should fit within certain requirements corresponding to the searched parameter, including the length of decimal point string and upper and lower parameter bounds.

The chromosome, as the model variable in EFSIM_T, is encoded into a binary string. The chromosome consists of two segments, including FL and WSVMs. The FL segment contains MF and *dfp* substrings. The WSVM segment contains penalty parameters C, kernel parameter γ from the RBF function and the lower boundary of weighted data parameter σ. Fig. 3 illustrates the chromosome structure.

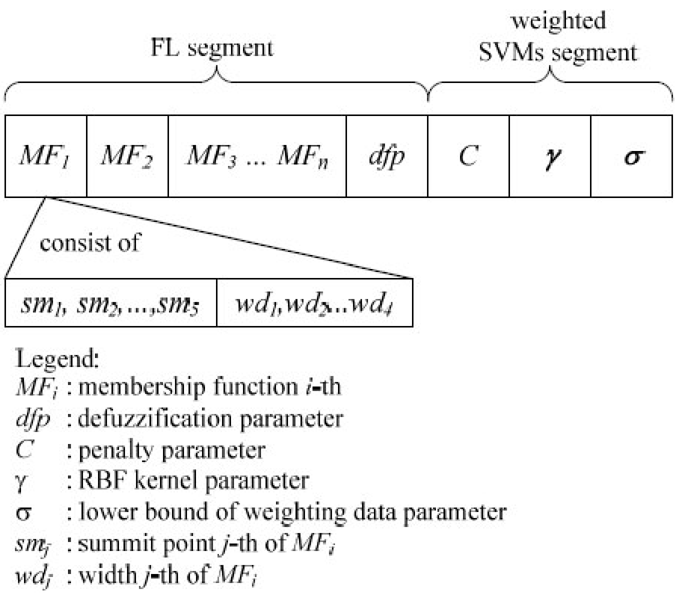


Fig. 3 *EFSIM_T chromosome structure [30].*

The search domains of every parameter are large, and it would be time-consuming and inefficient to conduct a comprehensive search. This research study adopts a search domain suggested by previous studies and systematically explores model parameters within reasonable range. The search domain for MFs is adopted from [18]. The search domains for C and γ are adopted from [12]. The search domain of σ are followed [22] suggestion, while *dfp* search domains determine by systematically exploring the parameters using several experiments within 0 to 1. Tab. I summarizes parameter settings and numbers of bits required for chromosome design.

Parameter	Upper bound (Ub)	Lower bound (Lb)	Number of bits	Remarks
MF set	$X^{ub} = 0.0$	$X^{lb} = 1.0$	27*)	Ub and Lb followed [18]
C	200	0	5	Ub and Lb followed [12]
γ	1	0.0001	10	
σ	1	0.1	10	Ub and Lb followed [22]
dfp	1	0.5	9	Ub and Lb determined by systematically exploring the range of parameters within 0 to1

Note: *) Number of bits required for one complete MF set

Tab. I *Summary of EFSIM$_T$ parameter settings.*

3.2 Knowledge Representation of Diaphragm Wall Deflection Problem

The EFSIM$_T$ proposed herein solved this problem using information and measurements from 18 historical deep excavation project cases located in metropolitan Taipei. Factors affecting this problem were referred to in a study by [6]. Similar to the latter [6] approach, the EFSIM$_T$ model is applied with the notion of making an accurate prediction of the succeeding stage that can be derived from the information of two or more previous stages as input. Therefore, diaphragm wall deflection data from prior stages are important inputs to help predict the values of deflection variables in succeeding stages of an excavation project [12].

In total, eight important factors were identified in [6], of which seven were selected as input parameters and one as the output parameter. All of the factors related to the diaphragm wall structure system are illustrated in Fig. 4.

In Fig. 4, H represents final excavation depth. According to [6], embedment depth is typically set to $0.8H$, therefore the total diaphragm wall length measured $1.8H$. W represents diaphragm wall thickness and Ri represents the observation point factor for each of the 18 segments. This segmented approach made diaphragm wall deflection prediction a time series data problem, and gave EFSIM$_T$ the potential to solve such. If, as in certain cases, embedment depth is less than $0.8H$, deflection between the bottom of diaphragm wall and the 19th observation point is assigned as linear and converges to zero. Each observation point can be regarded as an instance that consists of seven inputs and one output, illustrated as follows:

Inputs:

1. Diaphragm wall thickness: W
2. Depth of excavation surface: D
3. Equivalent SPT-N value between D+$0.25H$ and D-$0.25H$: $\overline{N}$
4. The factor of an observation point: R

5. Wall deflection of the observation point in (i-1)-th stage: S_{i-1}
6. Wall deflection of the observation point in (i-2)-th stage: S_{i-2}
7. Wall deflection of the observation point in (i-3)-th stage: S_{i-3}

Output:

1. The wall deflection of the observation point in i-th stage: S_i

Notably, i must be greater than or equal to three. This setting prevents the absence of fifth to seventh inputs. Therefore, when $i = 3$ the S_{i-3} is set to zero.

Tab. II shows data from 18 historical deep excavation project cases that include the number of excavation stages, excavation depth and construction method used. Among the 18 historical deep excavation project cases, 3 cases were performed by the bottom-up method and 15 cases performed by the top-down method. Bottom-up methods and top-down methods are two main basement construction methods [34]. In the bottom-up method, after the construction of basement piles and the diaphragm wall, the construction agency excavates the enclosed area to the desired depth and then proceeds for installation of the strutting/ bracing system to support the excavation walls as the excavation proceeds, followed by construction of

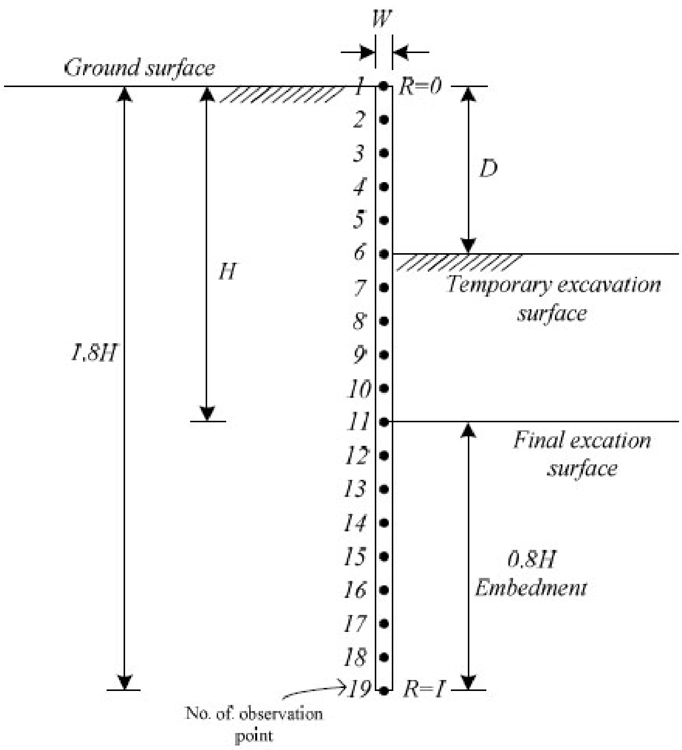


Fig. 4 *Representation of the diaphragm wall structure system [12].*

No.	Stages	Depth (m)	Construction method	No.	Stages	Depth (m)	Construction method
1	5	12.30	Top-down	10	6	14.05	Top-down
2	4	13.90	Bottom-up	11	4	13.60	Top-down
3	6	16.00	Top-down	12	5	17.35	Bottom-up
4	5	12.60	Top-down	13	5	13.15	Top-down
5	5	12.30	Top-down	14	5	23.85	Top-down
6	5	12.25	Top-down	15	6	19.40	Top-down
7	4	10.00	Top-down	16	6	19.40	Top-down
8	6	18.95	Top-down	17	5	13.70	Top-down
9	4	9.30	Top-down	18	7	19.70	Bottom-up

Tab. II *18 Historical cases of deep excavation projects in metropolitan Taipei [12].*

the basement [35]. As bottom-up methods employ temporary steel struts to balance the lateral pressure on the excavation, the top-down methods apply concrete floor slabs to resist the lateral earth pressure. Therefore, the top-down method is a construction method that builds the permanent structure members of the basement along with the excavation from the top to the bottom. Moreover, as the name implies, the top-down method allowed for simultaneous construction of the basement and superstructure erection.

In these 18 historical deep excavation project cases, the number of deep excavation stages varied from four to seven. As each stage was treated individually, these cases comprised 93 stages in total. Referencing observation [6] that engineering failures rarely occur during the first and second excavation stages, this study excludes these two stages from consideration. Therefore, similar to [6] and [12], valid data on a total of 57 stages were collected and used here.

Following the [6] and [12] data setting, the first seventeen construction cases (encompassing a total of 52 stages) were used for training. Data from the 18th case (5 stages) were employed in testing. Nineteen observation points were set for every stage, although excavation depths were not uniform. Therefore, 19 sets of data were collected for each stage. Based on the above, $52\times19 = 988$ training data sets and $5\times19 = 95$ testing data sets were collected. Tab. III shows example data of one complete excavation stage.

3.3 Results and Comparison

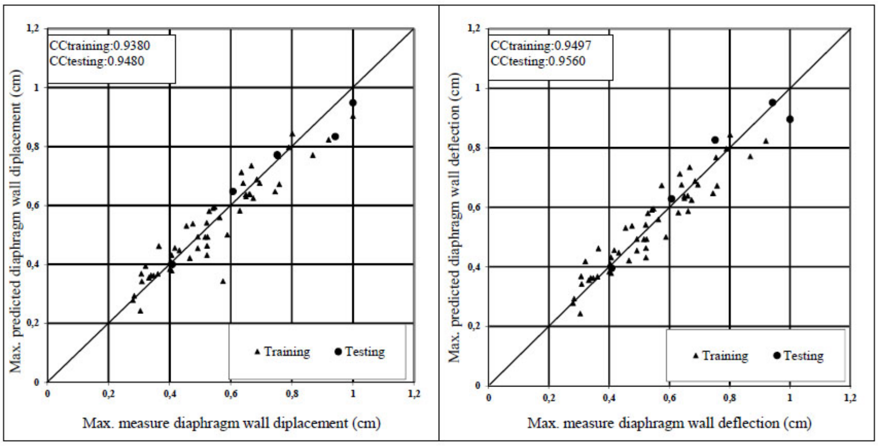
This section verifies and validates the performance of the hybrid system EFSIM$_T$ in predicting diaphragm wall deflection problems. As mentioned before, the model proposed herein predicts diaphragm wall deflection by adopting a database used by [6] and [12]. The database includes a total of 1,083 instances from the 57 excavation stages. Jan et al. [6] and Cheng and Wu [12] split the data into two groups, a total of 988 instances in 52 excavation stages were classified as training data, while 95 instances in five excavation stages were classified as the testing data.

Excavation Stage	Segment	Output Factor	Input Factors						
		Deflection (mm)	W (cm)	D (m)	R	$\overline{N}$	S_{i-1} (mm)	S_{i-2} (mm)	S_{i-3} (mm)
5	1	13.000	70	13.700	0.000	7.930	13.200	15.050	17.400
	2	17.680	70	13.700	0.056	7.930	18.150	16.090	15.420
	3	20.690	70	13.700	0.112	7.930	20.430	17.120	13.340
	4	24.320	70	13.700	0.168	7.930	22.970	17.540	12.440
	5	28.110	70	13.700	0.224	7.930	25.720	17.900	11.920
	6	31.960	70	13.700	0.280	7.930	27.520	17.900	11.350
	7	35.940	70	13.700	0.336	7.930	29.440	17.800	10.010
	8	40.780	70	13.700	0.392	7.930	31.460	17.010	8.670
	9	44.340	70	13.700	0.448	7.930	32.650	16.180	7.630
	10	47.050	70	13.700	0.504	7.930	30.990	14.920	6.720
	11	46.170	70	13.700	0.560	7.930	27.920	13.660	5.850
	12	43.160	70	13.700	0.616	7.930	24.770	12.350	5.190
	13	38.160	70	13.700	0.672	7.930	20.980	10.980	4.440
	14	32.180	70	13.700	0.728	7.930	17.200	9.550	3.210
	15	26.880	70	13.700	0.784	7.930	14.010	7.680	2.860
	16	21.890	70	13.700	0.840	7.930	10.900	5.800	1.820
	17	17.680	70	13.700	0.896	7.930	8.540	4.810	1.550
	18	13.850	70	13.700	0.952	7.930	6.260	3.870	1.370
	19	10.680	70	13.700	1.008	7.930	4.590	3.400	1.270

Tab. III *Example of data excavation stages.*

EFSIM$_T$ employs FL to manage environments characterized by uncertainty, vagueness, presumptions and subjectivity. This capability is suited to diaphragm wall deflection database characteristics, as causes and effects of factors that determine the behavior of the modeled problems do not need to be fully understood due to change of in situ environment and effects of construction [6]. Such conditions make prediction of diaphragm wall deflection problems a highly uncertain task. Moreover, EFSIM$_T$ is also able to deal with time series data characteristics inherent to the diaphragm wall deflection database, as the information and measurements are presented in 19 sequential diaphragm wall deflection data (each diaphgram wall is discretized into 18 uniform subintervals).

The accuracy of the proposed system is demonstrated by comparing the maximum predicted wall displacement and the maximum measured wall displacement for both time functions (linear and quadratic). Figs. 5a and 5b present comparison results between measured and predicted maximum diaphragm wall displacement. Fig. 5a presents prediction results using a linear time function. Fig. 5b presents prediction results using a quadratic time function. Figs. 5a and 5b also show the correlation coefficient between measured and predicted maximum diaphragm wall displacement using two time functions (i.e., linear and quadratic) for training and testing data.



(a) (b)

Fig. 5 *Measured vs. predicted maximum diaphragm wall deflection: (a) with linear time function; (b) with quadratic time function.*

As this research intends to compare the level of prediction result accuracy of the approach work, which employed only ANNs [6], and the other approach, which applied ESIM [12], the error qualification criteria are adopted from both researchers. The error qualification criteria are defined into three different ranges: less than 10%, between 10% and 20% and greater than 20%. Results of EFSIM$_T$ obtained using linear time function are 39 cases with relative error less than 10%, 14 cases with relative error between 10% and 20%, and four cases with relative error greater than 20%, respectively. The EFSIM$_T$results obtained using the quadratic time function are 38 cases with relative error less than 10%, 16 cases with relative error between 10% and 20%, and three cases with relative error greater than 20%. Using a relative percentage error of 20% as the threshold for prediction failure [6], the linear time function EFSIM$_T$ attained 92.98% accuracy and the quadratic time function EFSIM$_T$ reached 94.74% accuracy, while the prediction results of ANNs and ESIM only reached 77.19% and 78.95%, respectively.

Such results represent a significant improvement over results reported in [6], which used NNs, and [12], which employed ESIM, as shown in Table IV. Improvements reflect the superior abilities of EFSIM$_T$ to 1) cope with time series data characteristics inherent to segmental diaphragm deflection data and 2) handle the complex relationships between input and output variables and the many uncertainties related to geological conditions. Moreover, results obtained using the quadratic time function proved more desirable, as the level of accuracy is higher than using the linear time function.

4. Discussion

Based on the results and comparison mentioned above, EFSIM$_T$ can provide better estimation for providing estimates of lateral wall deflections or ground deforma-

Relative error (pct. range)	Number of cases			
	$EFSIM_T$			
	Linear time function	Quadratic time function	ESIM	ANNs
$[0\%, 10\%]$	39	38	32	28
$[10\%, 20\%]$	14	16	13	16
$> 20\%$	4	3	12	13
Model accuracy *)	92.98%	94.74%	78.95%	77.19%

Note: *) Prediction failure threshold: relative error >20%.

Tab. IV *Number of cases showing comparative accuracies of $EFSIM_T$, ESIM and ANNs.*

tions not only for preliminary design purposes, but also to evaluate the construction performance to avoid the braced excavations system failure. The $EFSIM_T$ as prediction model is not only able to capture the nonlinear relationship between input and output factors, but to handle inherent time series data characteristics. Such merits are beneficial as the deep excavation project has many stages. The deflection parameters in previous stages are important inputs to help predict the values of deflection variables in succeeding stages of an excavation project. Therefore, the deflection observed in any given stage is highly correlated to deflection parameters in previous stages. The data from previous stages can be employed to predict deflection in the following stage with improved accuracy. Moreover, $EFSIM_T$ developed by assuming the deep excavation data has inherent time series data characteristics which followed certain weighting patterns, whether they involve linear or quadratic functions. In reality, it is possible that the patterns might not match any function. Therefore, the $EFSIM_T$ performance can still be improved to reach the better prediction if $EFSIM_T$ able to generate the weighting pattern arbitrary. However, $EFSIM_T$ has great potential as a simple predictive tool for estimating the maximum diaphragm wall deflection.

This research only used seven input factors based on study [6] that considered only the behavior of diaphragm walls obtained from 18 construction projects in the Taipei Basin. In their research work, [6] neither include the sub-soil conditions nor as the support system data. The factors used are related to the diaphragm wall structure system and information for two or more previous stages as the monitored data. As inherent limitation of [6] historical data, logically the level of accuracy as well as the level of confidence can be different when the prediction includes more factors. Such approaches are used in [9] by expanding the research work conducted in [8] who employed only ANNs and seven input factors. Chua and Goh [9], used 35 input factors in total, which include the support system properties as well as the variation of soil properties with depth. However, as a large database is used in this research work, to avoid the "overfitting" problem, [9] is modified the ANNs into the evolutionary Bayesian back-propagation (EBBP) that combines ANNs with genetic algorithms and the Bayesian framework. For that reason, the expanding research work can be generated by including more input factors such as soil parameters and the safety factor. As the historical data used were taken exclusively from the Taipei Basin, the expanding research work also needs to consider extending to the other data sets taken from other areas that have different geological conditions as well as the scale and method of construction.

5. Conclusion

In order to prevent costly damage, builders try to avoid excessive diaphragm wall deflection and excavation-induced ground settlement. Therefore, it is necessary to make accurate diaphragm wall deflection predictions to ensure actual deflection falls within allowable limits, and this ensure the safety of both the project and adjacent structures. This study proposed EFSIM$_T$ as an alternative hybrid AI approach to predict diaphragm wall deflection using historical cases.

EFSIM$_T$ was developed by fusing together FL, weighted SVMs and FMGA. FL was used to address uncertainties inherent in geotechnical problems (e.g., soil parameters); weighted SVMs addressed complex relationships related to fuzzy input-output mapping and focused on segmented diaphragm wall deflection observations that exhibit time series data characteristics; and FMGA was deployed as an optimization tool to handle FL and weighted SVM search parameters.

The accuracy of the proposed EFSIM$_T$ was higher (for both linear and quadratic functions) than either ANNs-only proposed in [6] or ESIM approaches proposed in [12]. This is attributable to the superior ability of EFSIM$_T$ to cope with 1) time series data characteristics inherent in diaphragm wall deflection data, 2) complex relationships between input and output variables, and 3) uncertainties inherent in diaphragm wall deflection problems. Unlike other approaches, whether empirical or semi-empirical, EFSIM$_T$ as the hybrid artificial intelligence system is able to capture the nonlinear relationship between input and output factors and also handles inherent time series data characteristics, without requiring to derive mathematical models in the form of some analytical nonlinear functions. Hence, the EFSIM$_T$ has great potential as a predictive tool for diaphragm wall deflection problems, assisting project managers/ engineers to ensure safety during the construction process and preventing costly damage to adjacent infrastructure and buildings. Moreover, as the EFSIM$_T$ here has been applied to a limited number of input factors related to the diaphragm wall structure system and to information for two or more previous stages, the new research studies can be generated to compare how well the prediction level can be improved if the new research studies include the sub-soil conditions, the support system factors and the safety factors.

References

[1] Peck, R. B.: Deep excavation and tunneling in soft ground, In: Proceeding 7th International Conference on Soil Mechanics and Foundation Engineering, Mexico City, International Society for Soil Mechanics and Foundation Engineering, pp. 225–290, 1969.

[2] Clough, G. W., O'Rourke T. D.: Construction induced movement of in situ walls, In: Proceedings Design and performance of earth retaining structures, ASCE, pp. 439-470, 1990.

[3] Ou, C. Y., Hsieh, P. G., Chiou, D. C.: Characteristics of ground surface settlement during excavation. Can. Geotech. J., 30, (5), pp. 758–767, 1993.

[4] Hashash, Y. M. A., Whittle, A. J.: Ground movement prediction for deep excavations in soft clay. Journal of Geotechnical Engineering, 122, (6), pp. 474-486, 1996.

[5] Hashash, Y. M. A., Marulanda, C., Ghaboussi, J., Jung, S.: Systematic update of deep excavation model using field performance data. Computer and Geotechnics, 30, pp. 477–488, 2003.

[6] Jan, J. C., Hung, S. L, Chi, S. Y., Chern, J. C.: Neural network forecast model in deep excavation. Journal of Computing in Civil Engineering, 16, (1), pp. 59–65, 2002.

[7] Osouli, A., Hashash, Y. M. A., Song, H.: Interplay between field measurements and soil behavior for capturing supported excavation response. Journal of Geotechnique and Geoenvironmental Engineering, 136, (1), pp. 69-84, 2010.

[8] Goh, A. T. C., Wong, K. S., Broms, B. B.: Estimation of lateral wall movements in braced excavations using neural networks. Canadian Geotechnical Journal, 32, (6), pp. 1059–1064, 1995.

[9] Chua, C. G., Goh, A. T. C.: Estimating wall deflection in deep excavations using Bayesian neural networks. Tunnelling and Underground Space Technology, 20, pp. 400–409, 2005.

[10] Tang, K. S., Chan, C. Y., Man, K. F., Kwong, S.: Genetic structure for NN topology and weights optimization, In: Proceeding of the 1st International Conference on Genetic Algorithms in Engineering Systems: Innovations and Applications, IEE, Stevenage, England, pp. 250–255, 1995.

[11] Rubanov, N. S.: The layer-wise method and the backpropagation hybrid approach to learning a feedforward neural network. IEEE Transactions on Neural Network, 11, (2), pp. 295–305, 2000.

[12] Cheng, M. Y., Wu, Y. W.: Prediction of Diaphragm Wall Deflection in Deep Excavations Using Evolutionary Support Vector Machine Inference Model (ESIM). In: Proceeding 26th International Symposium on Automation and Robotics in Construction (ISARC 2009), pp. 176–182, 2009.

[13] Bojadziev, G., Bojadziev, M.: Fuzzy logic for business, finance, and management, 2nd ed., World Scientific, Singapore, 2007.

[14] Zadeh, L. A.: Fuzzy sets. Information and Control, 8, (3), pp. 338–353, 1965.

[15] Heshmaty, B., Kandel A.: Fuzzy Linear Regression And Its Applications To Forecasting In Uncertain Environment. Fuzzy Sets And Systems, 15, pp. 159–191, 1985.

[16] Zadeh, L. A.: Outline of a new approach to the analysis of complex systems and decision processes. IEEE Transactions on Systems, Man, and Cybernetics, 3, (1), pp. 28–44, 1973.

[17] Klir, G. J., Yuan, B.: Fuzzy sets and fuzzy logic: Theory and applications. Prentice Hall PTR, Upper Saddle River, New Jersey, 1995.

[18] Ko, C. H.: Evolutionary Fuzzy Neural Inference Model (EFNIM) for Decision-Making in Construction Management, Ph.D. Thesis, Department of Construction Engineering, National Taiwan University of Science and Technology, Taipei, Taiwan, 2002.

[19] Ishigami, H., Fukuda, T., Shibata, T., Arai, F.: Structure optimization of fuzzy neural network by genetic algorithm. Fuzzy Sets and Systems, 71, pp. 257–264, 1995.

[20] Martinez, C., Castillo, O., Montiel, O.: Comparison between Ant Colony and Genetic Algorithms for Fuzzy System Optimization, in: Castillo et al. (Eds.). Soft Computing for Hybrid Intel. Systems. Springer-Verlag, Berlin Heidelberg. pp. 71–86, 2008.

[21] Fan, H., Ramamohanarao, K.: A weighting scheme based on emerging patterns for weighted support vector machines. In: Proceeding IEEE International Conference on Granular Computing, 2, (2), pp. 435–440, 2005.

[22] Lin, C. F., Wang, S. D.: Fuzzy Support Vector Machines. IEEE Transactions on Neural Networks, 13, (2), pp. 464–471, 2002.

[23] Burges, C.: A tutorial on support vector machines for pattern recognition. Data Mining and Knowledge Discovery, 2, (2), pp. 121–167, 1998.

[24] Lin, C. F.: Fuzzy Support Vector Machines. Ph.D. Thesis, Dept. of Electrical Engineering, National Taiwan University, Taipei, Taiwan, 2004.

[25] Hsu, C. W., Chang, C. C., Lin, C. J.: A practical guide to support vector classi?cation, Technical report, Department of Computer Science, National Taiwan University, Taipei, Taiwan, 2003.

[26] Khemchandani R., Jayadeva, Chandra S.: Regularized least square fuzzy support vector regression for financial time series forecasting. Expert System with Applications, 36, (1), pp. 132–138, 2009.

[27] Feng, C. W., Wu, H. T.: Integrated fmGA and CYCLONE to Optimize Schedule of Dispatching RMC Trucks. Automation in Construction, 15, pp. 186–199, 2006.

[28] Goldberg, D. E., Deb, K., Kargupta, H., Harik, G.: Rapid, accurate optimization of difficult problems using fast messy genetic algorithms. In: Proceedings of the Fifth International Conference on Genetic Algorithms, pp. 56–64, 1993.

[29] Goldberg, D. E., Deb, K.: mGA in C: A Messy Genetic Algorithm in C. IlliGAL Technical Report 91008, Department of General Engineering, University of Illinois at Urbana-Champaign, Urbana, Illinois, 1991.

[30] Cheng, M. Y., Roy, A. F. V.: Evolutionary fuzzy decision model for cash flow prediction using time-dependent support vector machines. International Journal of Project Management. 29, pp. 56–65. 2011.

[31] Hayashi, I., Umano, M., Maeda, T., Bastian, A., Jain, L. C.: Acquisition of fuzzy knowledge by NN and GA—A survey of the fusion and union methods proposed in Japan. In: Proceeding 2nd international conference on knowledge-based intelligent electronic systems, Piscataway, pp. 69–78, 1998.

[32] Cheng, M. Y., Ko, C. H.: A genetic-fuzzy-neuro model encodes FNNs usingSWRM and BRM. International Scientific Journal Engineering Applications of Artificial Intelligence, 19, pp. 891–903, 2006.

[33] Chang, C. C., Lin., C. J.: LIBSVM: a library for support vector machines, 2001, Software available at http://www.csie.ntu.edu.tw/∼cjlin/libsvm

[34] Wu, C. H., Ou, C. Y., Tung, N.: Corner Effects In Deep Excavations – Establishment of a Forecast Model for Taipei Basin T2 Zone, Journal of Marine Science and Technology, 18, (1), pp. 1–11, 2010.

[35] Basarkar, S. S., Kumar, M., Mohapatro, B. G., Mutgi, P. R.: Emerging Trend in Deep Basement Construction: Top -Down Technique, IOSR Journal of Mechanical and Civil Engineering (IOSR-JMCE), In: Proceeding Second International Conference on Emerging Trends in Engineering, 2013.