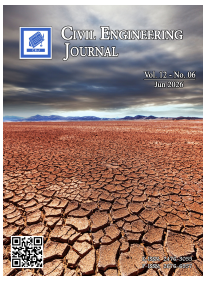
Available online at www.CivileJournal.org

Civil Engineering Journal

(E-ISSN: 2476-3055; ISSN: 2676-6957)

Vol. 12, No. 06, June, 2026



Bidirectional Estimation of Self-Compacting Concrete Parameters Using Simplified Neuro-Fuzzy Networks

Dinh V. Do [1], Robert Salat [2], Cuong H. Nguyen [3], Linh H. Tran [4*]

[1] *Faculty of Electrical Engineering, Sao Do University, Hai Duong 170000, Vietnam.*

[2] *Faculty of Electrical and Computer Engineering, Cracow University of Technology, Krakow 31-155, Poland.*

[3] *Faculty of Building and Industrial Construction, Hanoi University of Civil Engineering, Hanoi 10000, Viet Nam.*

[4] *Faculty of Electrical and Electronic Engineering, Hanoi University of Science and Technology, Hanoi 10000, Viet Nam.*

Received 11 March 2026; Revised 18 May 2026; Accepted 23 May 2026; Published 01 June 2026

Abstract

The objective of this paper is to propose a novel bidirectional modeling framework using simplified neuro-fuzzy networks to approximate the complex dependencies between the workability parameters of self-compacting concrete (SCC) and the quantities of its five principal ingredients, namely cement, fly ash, water, additives, and admixtures. The forward model predicts seven key workability parameters based on the input ingredients and is essential for assessing the quality and performance of fresh SCC before production begins. In contrast, the backward model is trained to recommend the specific quantities of input ingredients required to achieve the expected SCC parameters. A comprehensive dataset consisting of 480 field-tested SCC mixtures was developed and used to train simplified Takagi–Sugeno–Kang (sTSK) networks. The numerical experiments demonstrated high accuracy, with the forward model achieving average relative errors of less than 2.46% and the backward model achieving an average relative error of 1.62%. These findings highlight the effectiveness of the proposed solutions in developing highly accurate approximation models. This study introduces a robust backward estimation approach, addressing a gap left by conventional forward-only models. Moreover, by incorporating Mahalanobis distance into the sTSK architecture, the proposed models use three times fewer nonlinear parameters than classical versions. This reduction enables their potential practical application in real-field concrete mixing and production systems for on-site quality control in production environments.

Keywords: Bidirectional Approximation; Neuro-Fuzzy Network; Self-Compacting Concrete; Workability Parameters.

1. Introduction

Self-compacting concrete (SCC) is a relatively new type of concrete, which has exceptional flexibility and ability to effortlessly flow under its own weight, efficiently filling every form's corners without the need for external vibrations. This unique property enables SCC to flow through to easily fill all corners of structure's geometries. However, the performance of SCC is very susceptible to the precise quantities of primary components, including cement, fly ash, and various chemical admixtures, and the influence of climatic conditions, particularly variations in ambient temperature and humidity [1-3]. Studies continue to focus on refining the input ingredient amounts to improve SCC workability parameters. Most of those studies require manual adjustment of mixed input scenarios to achieve target performance through costly traditional, iterative methods [4-6]. On the other hand, over time, the workability of SCC gradually

* Corresponding author: linh.tranhoai@hust.edu.vn

https://doi.org/10.28991/CEJ-2026-012-06-06

diminishes, a phenomenon exacerbated by high-temperature environments and low humidity levels. As a consequence, the quality and strength of SCC would be affected, complicating construction processes and undermining the overall quality and strength of the whole concrete structure [2, 7].

In the reality of Vietnam's tropical monsoon climate, the adverse weather cycles, including periods of high heat and dryness, pose significant challenges to the workability of SCC. Consequently, there is a compelling need to undertake research and development efforts aimed at devising a predictive model specifically tailored to the Vietnamese climate and the ingredients produced locally. In practical terms, the development of computational models capable of estimating SCC workability parameters based on input components represents a highly anticipated advancement. Presently, SCC formulations rely on predetermined combinations of input components, necessitating extensive retesting of various mixed input scenarios to adjust SCC parameters. This iterative process is both time-consuming and costly. Therefore, accurate computational models offer a promising solution, enabling fast and easy estimation of anticipated outcomes in terms of workability parameters of the formulated SCC.

To bridge the above mentioned gaps, this paper presents a bidirectional application of simplified neuro-fuzzy networks, utilizing a forward network and a backward network to map the complex relationships between self-compacting concrete (SCC) workability parameters and their mixing ingredients. The forward network is designed to estimate parameters of self-compacting concrete (SCC) based on the quantities of primary input ingredients of the mixture, such as cement, fly ash, super-ductile additive (SD), viscosity-modifying admixtures (VMA), and water [8, 9]. Much research has focused on a single selected parameter, such as compressive strength [10-12], plastic viscosity [13], or compressive creep behavior [14, 15]. Other studies have investigated the behavior of SCC mixtures under varying environmental conditions [16]. This paper, as the selected SCC parameters, focuses on seven most popular ones, including slump flow (SF), t_{500}, V_{funnel}, L_{box}, J_{ring}, segregation ratio (SR), and R_{28} [3, 17, 18]. In contrast, the backward network functions as an inverse estimator to derive the quantities of the ingredients when the workability parameters are provided as inputs. The schemes of the forward and backward networks are shown in Figure 1. A key improvement of this proposal is the use of the simplified TSK network based on the use of Mahalanobis distance, which allows the system to operate with three times fewer nonlinear parameters than classical versions. This refinement ensures a faster training process and robust generalization even with smaller industrial data samples, providing a practical tool for real-time, on-site quality control in production environments.

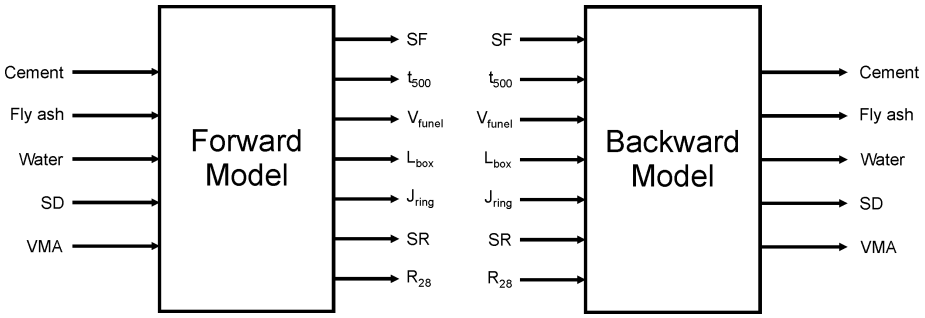


Figure 1. The proposed forward and backward models to approximate the dependencies between input ingredients and the workability parameters of SCC

Various studies have investigated models for estimating specific SCC parameters, including both traditional mathematical frameworks [19] and modern AI-based solutions, among which neural networks are the most widely adopted approach. The popularity of AI-based, data-driven models is attributed to their ability to learn directly from complex data, especially when the physical dependencies between variables are only partially known. Among studies on SCC performance, most have focused only on forward models. Many models have been proposed solely to predict one of the most important parameters, R_{28}, based on input ingredients using machine learning algorithms [20–22]. Another study by Serraye et al. [23] used the W/B ratio and six additional input variables to predict R_{28}. Moreover, a distinct approach proposed by el Asri et al. [24] involved predicting R_{28} based on selected workability parameters, such as SF, V_{funnel}, and L_{box}. The model proposed in that study achieved a notable correlation coefficient of 0.976. Other studies have focused on predicting additional workability parameters, as reported in [16, 18, 25]. Recent artificial intelligence solutions based on deep learning have gained significant momentum in many studies, such as [26, 27]. However, the main limitation of these techniques is that training deep networks with millions of parameters requires large datasets to avoid overfitting. For industrial problems, this requirement is very difficult to satisfy.

In the study of this article, the computational models were trained and tested based on the data from experimental mixing of 480 SCC samples, which were created, adhering to prescribed standard procedures and utilizing appropriate measuring instruments. The conceptual workflow of the frameworks to train and to select the backward and the forward models based on the generated mixture samples is presented in Figure 2.

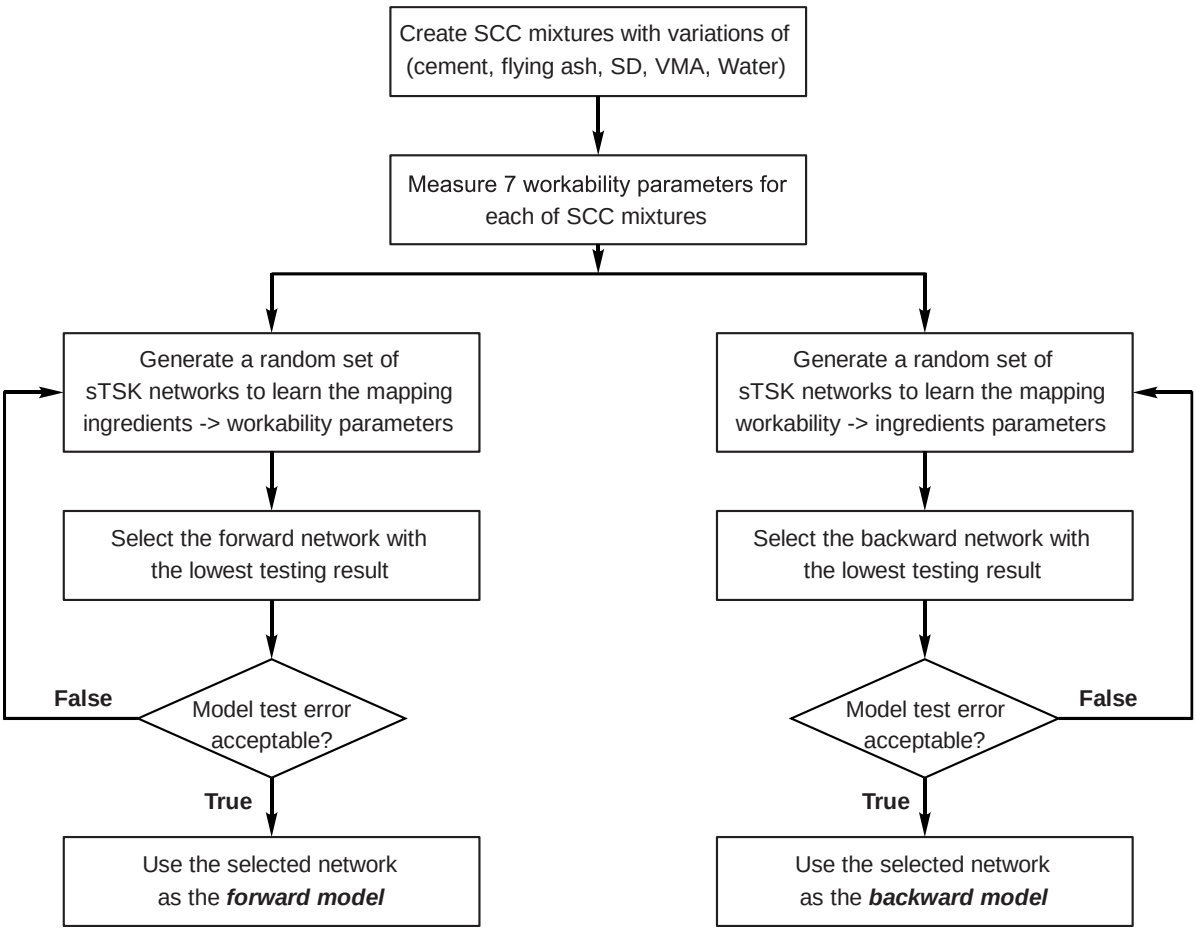


Figure 2. The flowchart of the proposed methodology to train and to select the forward and the backward models

Leveraging neuro-fuzzy networks as nonlinear estimators for bidirectional estimations, the networks were optimized to predict seven selected workability parameters based on the specified input quantities of the aforementioned components of the SCC mixtures. These models, revered for their efficiency in tackling various technical challenges [8, 28, 29], possess adaptability, enabling their parameters to be trained and updated to fit specific data samples. The numerical findings attest that neuro-fuzzy networks serve as commendable nonlinear estimators for SCC workability parameters.

2. Short Descriptions of the SCC Workability Parameters and Their Measurements

The computational models employed in this study focus on estimating specific parameters of SCC, as delineated in the standards set forth by Vietnamese Standard No. 12209 [17]. Within this framework, particular attention is given to the seven parameters of primary interest, which have been briefly discussed in the Introduction section of this paper and are further elaborated upon below.

Among the parameters subject to estimation, the slump flow (*SF*) and t_{500} time stand out as indicators of SCC's fillability in structures devoid of obstructions. The methodology outlined in Vietnamese Standard No. 12209 [17] provides a structured approach for assessing these parameters. To illustrate, a simplified schematic (as in Figure 3) captures the essence of the testing procedure, wherein SCC is poured out from a collar to spread freely, expanding as it moves. The t_{500} time signifies the duration (rounded to 0.1 seconds) taken for the spreading SCC to reach any point at the distance of 500 mm. In practice it's done by marking a circle with a diameter of 500 mm and checking when the spreading SCC reach to perimeter of that circle (e.g., Figure 3 shows that the SCC reached point A first). Notably, this time interval not only serves as a measure of flowability but also offers insights into the relative viscosity of the concrete samples. Subsequently, the SCC's propagation persists until it comes to a standstill, at which juncture the diameters of the SCC are accurately recorded and evaluated.

By delving into the intricacies of these parameters and their underlying testing methodologies, this study seeks to provide a comprehensive understanding of SCC's behavior and characteristics, thereby laying the groundwork for the development of robust computational models capable of accurately predicting these essential properties.

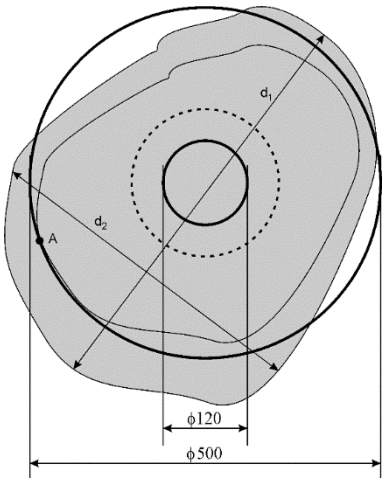


Figure 3. The depiction of the d_1 and d_2 used in the SF test [17, 18]

The slump flow (SF) parameter of a concrete sample is defined as:

$$SF = \frac{d_1+d_2}{2} \quad (1)$$

where, d_1 is the longest distance of the spread area of SCC (also called as the diameter of the area) and d_2 is the distance forming a right angle to d_1 as depicted in Figure 3.

The V_{funnel} time is defined as the duration required for a concrete sample from a fully filled V-shaped funnel to flow out completely. The proposed design of the V-shaped funnel is presented in Figure 4. The measured V_{funnel} time should be rounded to 0.1 s.

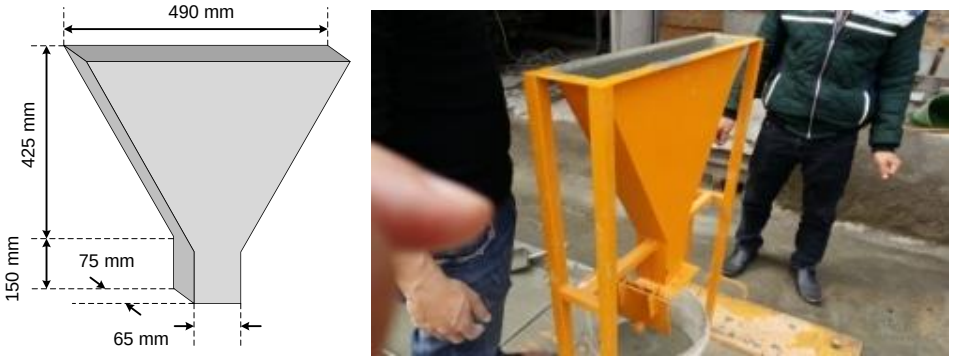


Figure 4. The V-shaped funnel and a demonstration of measuring V_{funnel} parameter [17, 18]

The L_{box} parameter provides insight into the behavior of SCC by measuring the ratio of heights observed at both ends of the concrete specimen once it has spread within an L-shaped enclosure similar to the device presented in Figure 5. This L_{box} parameter serves as a critical indicator of SCC's ability to effectively navigate through confined spaces, particularly between reinforcement bars within complex building structures.

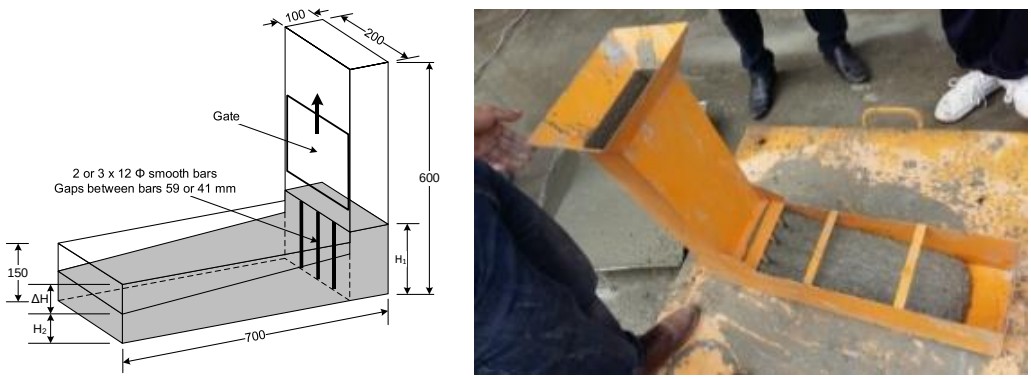


Figure 5. The L-shaped box and a demonstration of measuring L_{box} parameter [17, 18]

In order to assess segregation resistance of the SCC, the sieve segregation test is proposed. This method involves allowing freshly prepared SCC to settle for an approximate duration of 15 minutes. After that, the upper portion of the concrete sample is extracted and poured into a sieve with a mesh size of 5 mm. By comparing the material that passes through the sieve with the total mass poured, the segregation ratio is determined as:

$$SR = \frac{(m_{ps} - m_p) \cdot 100}{m_c} \quad (2)$$

where, m_{ps} is the mass of the part of SCC that passed through the sieve (including the sieve receiver), m_p is the mass of the sieve receiver, and m_c is the total input mass of SCC.

The J_{ring} test evaluates the passing ability of SCC samples in the presence of obstructions. In this test, fresh SCC samples are poured into a slump cone positioned upright on a rigid, square base plate. As presented on Figure 6, during the J_{ring} test, the obstructions to the concrete are simulated with the help of a ring-shape device. The ring-shape device is placed in the center of the square base plate with the cone inside. With the SCC sample poured inside, the cone is pulled up to allow the concrete sample to spread out and part of them passes through the ring.

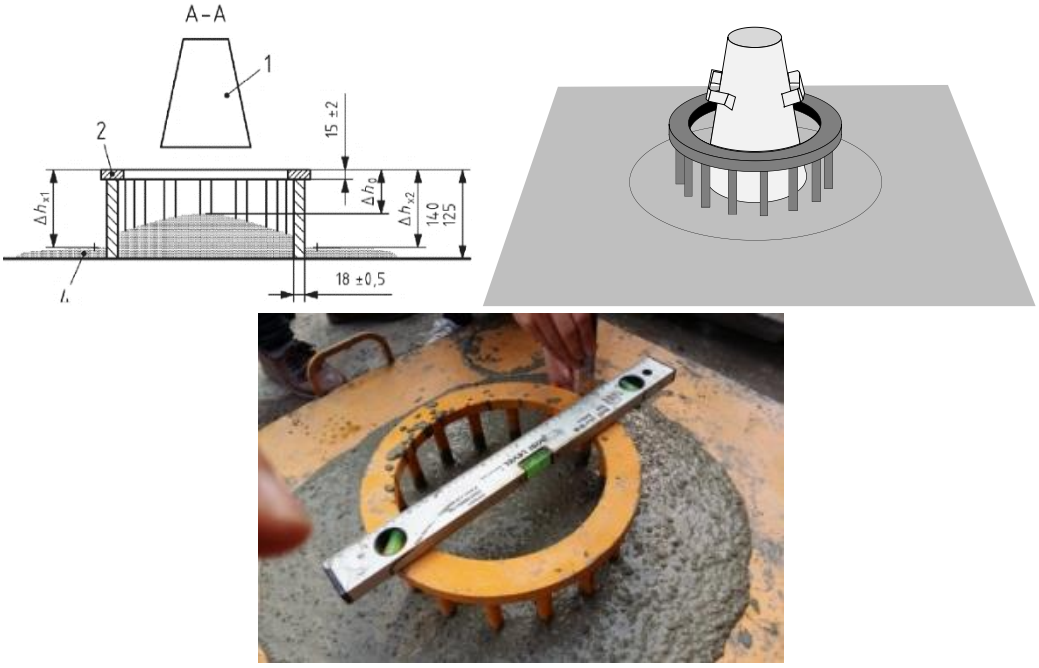


Figure 6. The design of a J_{ring} with characteristic points and values and its application in practical test

After the sample is settled, 5 characteristic height differences were measured as:

- The height difference measured at the center of the ring: Δh_0
- The two height differences at two farthest points of the ring: Δh_{x1} and Δh_{x2}.
- The two height differences at two farthest points of the ring device at the perpendicular direction to the previous two points: Δh_{y1} and Δh_{y2}.

With these 5 height differences, the J_{ring} parameter is determined using the following formula:

$$J_{ring} = \frac{\Delta h_{x1} + \Delta h_{x2} + \Delta h_{y1} + \Delta h_{y2}}{4} - \Delta h_0 \quad (3)$$

and for the SCC sample to meet acceptance criteria, this value ought to be under 10 mm.

The final, 7th parameter R_{28}, assesses the compressive strength of the concrete mixture. Concrete samples were taken upon completion of the mixture and stored under standardized conditions. After 28 days, these samples were subjected to compressive strength testing using a spindle compressor.

The operational characteristics of SCC mixtures are dictated by the quantities of input components utilized. In this study, five key components were varied for investigation: cement powder, fly ash, SD, VMA, and water. Each test scenario maintained a same aggregate composition of 770 kg of rocks and 808 kg of sands. 30 main mixtures of SCC were investigated, with representative ingredient compositions detailed in Nguyen and Tran [18]. To comprehensively capture the effects of ingredient variations, for each SCC mixtures, multiple tests were conducted with small variations of input ingredients, resulting in a total of 480 data samples.

Analysis of the measurement results underscores very clearly the nonlinear dependencies of SCC parameters on the input components [11]. Given this nonlinear relationship, there arises a high demand for mathematical models capable of accurately predicting SCC parameters based on the quantities of input ingredient. Such models offer invaluable generalization capabilities, enabling predictions for input conditions not explicitly represented in the training data samples [30]. In this study, we propose to use the simplified Takagi–Sugeno–Kang (TSK) neuro-fuzzy network. These models are well-known for their effective capability of modeling complex nonlinear mapping.

3. The Simplified Neuro-Fuzzy Network as Nonlinear Estimators

In this study a simplified model of TSK neuro-fuzzy network is proposed as nonlinear estimators for the nonlinear dependencies between the input ingredients and the workability parameters. The structure of a network with *N* inputs, *M* fuzzy rules is shown on Figure 7. This simplified model has two significant deviations from the original classical TSK model [31]:

1. The fuzzifier blocks use the Mahalanobis distance [32] in lieu of the Euclidean distance to compute the similarity between the input vector and the center of the corresponding fuzzy rule, and
2. It refrains from calculating the weighted responses in the final output.

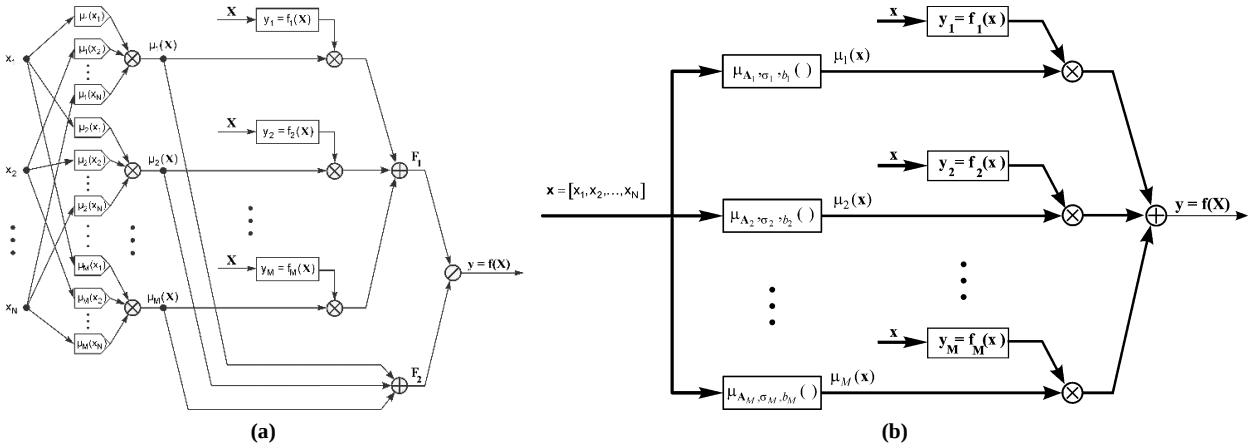


Figure 7. The structure of (a) classical TSK network with one output and (b) the corresponding network after simplification

For an input vector $\boldsymbol{x} = [x_1, x_2, \dots, x_N] \in \mathbb{R}^N$, and a center point $c_i = [c_{i1}, c_{i2}, \dots, c_{iN}] \in \mathbb{R}^N$ in the classical TSK network, the membership function estimates "how close the input vector x is to the given center point c_i" using product operator as follow:

$$\mu(\boldsymbol{x} \approx \boldsymbol{c}_i) = \mu_i(\boldsymbol{x}) = \prod_{j=1}^{N} \mu_i(x_j) = \prod_{j=1}^{N} \frac{1}{1+\left(\frac{x_j - c_{ij}}{\sigma_{ij}}\right)^{2b_{ij}}} \tag{4}$$

where it's evident that each dimension of x and c_i contributes equally to the membership, or in other words, the locus of equidistant points is a hypersphere. Consequently, from Equation 4 it can be seen that the network has $N \times M$ width parameters σ_{ij} and also $N \times M$ shape parameters b_{ij}.

In the modified network, the membership function is defines using the formula with the Mahalanobis distance [32]:

$$\mu_i(\boldsymbol{x}) = \frac{1}{1+\left(\frac{\|x-c_i\|_M^2}{\sigma_i^2}\right)^{b_i}} = \frac{1}{1+\left(\frac{\|x-c_i\|^T \cdot A_i \cdot \|x-c_i\|}{\sigma_i^2}\right)^{b_i}} \tag{5}$$

where the so called *scaling matrices* Ai would cause the locus of equidistant points to be a hyperellipsoid. The shapes of hyperellipsoids enhance the space division into activation areas of the rules to support better data modelling. This effect causes that a simple sum of responses can be used instead of the weighted sum [31].

The incorporation of scaling matrices A*i* also allow to reduce the number of width and shape parameters both to *M* only (i.e. each reasoning rule has only one parameter each, not *N*) and the number of shape parameter b_i also to *M* only.

For the classic network, as shown in Figure 7-a, the final output is a *weighted* response from the individual reasoning rules:

$$y = \frac{\sum_{i=1}^{M} \mu_i(\boldsymbol{x}) \cdot f_i(\boldsymbol{x})}{\sum_{i=1}^{M} \mu_i(\boldsymbol{x})} \tag{6}$$

where the simplified TSK, shown in Figure 7-b, simply compute the sum of the responses:

$$y = \sum_{i=1}^{M} \mu_i(\boldsymbol{x}) \cdot f_i(\boldsymbol{x}) \tag{7}$$

where the TSK function $f_i(\boldsymbol{x})$ are linear functions of input signals.

The significant reduction of nonlinear parameters σ_i and b_i and the removal of the weighting sum have a substantial impact on the complexity and the network's training processes. However, the Equation 5 introduces additional matrices A_i as nonlinear parameters. But as demonstrated in Tran [31], these matrices can be estimated once following the initialization of the data centers using clustering methods and they do not require updating during the costly iterative training of other parameters. With a dataset containing p input samples x_j and a set of M data centers c_i determined from any clustering method, the matrices A_i can be estimated as follow (for $j = 1,2,\dots,p;\ i = 1,2,\dots,M$):

Determine membership values u_{ji} indicating how strong vector x_j relates to centers c_i:

$$u_{ji} = \frac{\frac{1}{\|x_j - c_i\|^2}}{\sum_{k=1}^{M} \frac{1}{\|x_j - c_i\|^2}} \quad \text{if no distance is 0} \tag{8-a}$$

If any of the distance is 0, then the membership is calculated as:

$$u_{ji} = \begin{cases} 1 & if \quad \|\boldsymbol{x}_j - \boldsymbol{c}_i\| = 0 \\ 0 & if \quad \exists k \neq i: \|\boldsymbol{x}_j - \boldsymbol{c}_k\| = 0 \end{cases} \tag{8-b}$$

- Determine the covariance matrices $\mathbf{F}_i$:

$$\boldsymbol{F}_i = \sum_{j=1}^{p} u_{ji}^2 (\boldsymbol{x}_j - \boldsymbol{c}_i) \cdot (\boldsymbol{x}_j - \boldsymbol{c}_i)^T \tag{9}$$

- Determine the scaling matrices $\mathbf{A}_i$:

$$\boldsymbol{A}_i = \sqrt[N]{det(\boldsymbol{F}_i)} \cdot \boldsymbol{F}_i^{-1} \tag{10}$$

For the TSK network, akin to other feedforward networks, can be trained in a conventional approach of employing machine learning algorithms in supervised mode. The training process utilizes the learning and the testing datasets to determine the sub-optimal values for the parameters. If the learning data samples are pairs of input-output $\{x_i, d_i\}$ where $i = 1,\dots,p;\ x_i \in \mathbb{R}^N;\ d_i \in \mathbb{R}^K$ then the parameters of the TSK network are adapted to generate the output signals following the expected destination value, i.e. minimizing the error function:

$$E_{learn} = \frac{1}{2} \sum_{i=1}^{p} \|TSK(\boldsymbol{x}_i) - \boldsymbol{d}_i\|^2 \rightarrow min \tag{11}$$

After that, the network is tested with other pairs of data $\{\boldsymbol{x}_i^{test}, \boldsymbol{d}_i^{test}\}$ where $i = 1,\dots,q$ (it's recommended that the testing data are new, i.e. are not part of the learning data):

$$E_{test} = \frac{1}{2} \sum_{i=1}^{q} \|TSK(\boldsymbol{x}_i^{test}) - \boldsymbol{d}_i^{test}\|^2 \tag{12}$$

During the training process, for each try of the network, the number of reasoning rules is determined using the classical "trial and test" approach. Starting with very simple networks (for example with only one rule), the number of rules is increased during the trials and for each configuration of the network, a number of networks were randomly initialized, trained and tested with the provided datasets. With these different networks were trained and tested, the winning network is the one with lowest testing error. For training the TSK networks, the hybrid algorithm described in Padavala et al. [25] was used.

4. Numerical Experiments and Results

For the purpose of this study, experimental 480 SCC samples (16 variants of each of 30 types of SCC) were mixed and tested. For each of 30 types of SCC, 12 variants were randomly selected to the training set, which equal to 360 learning samples, the remaining 4 variants (i.e. 120 samples) were assigned to the testing set. As measures of performance, the following error functions were determined for the testing dataset:

- SSE (*Sum of Squared Error*) for training data:

$$SSE = \sum_{i=1}^{p} \|\boldsymbol{y}_i - \boldsymbol{d}_i\|^2 \tag{13}$$

- MAE (*Mean Absolute Error*):

$$MAE = \frac{1}{q}\sum_{i=1}^{q}\|\boldsymbol{y}_i^{test} - \boldsymbol{d}_i^{test}\| \quad (14)$$

- MRE (*Mean Relative Error*):

$$MRE = \frac{1}{q}\sum_{i=1}^{q}\frac{\|\boldsymbol{y}_i^{test} - \boldsymbol{d}_i^{test}\|}{\|\boldsymbol{d}_i^{test}\|} \quad (15)$$

- Max AE (*Max Absolute Error*):

$$Max\ AE = \max_{i=1\to q}\|\boldsymbol{y}_i^{test} - \boldsymbol{d}_i^{test}\| \quad (16)$$

- Max RE (*Max Relative Error*):

$$Max\ RE = \max_{i=1\to q}\frac{\|\boldsymbol{y}_i^{test} - \boldsymbol{d}_i^{test}\|}{\|\boldsymbol{d}_i^{test}\|} \quad (17)$$

For the purpose of estimating 7 parameters of the self-compacting concrete (i.e. the *SF*, t_{500}, V_{funnel}, L_{box}, J_{ring}, *SR* and R_{28}), the forward TSK network would have 7 outputs corresponding to the parameters, 5 inputs corresponding to the input parameters of 5 main components (cement, fly ash, water, SD, VMA). Inversely, the backward network would have corresponding 7 inputs and 5 outputs to estimate the required amounts of input ingredients from the expected workability parameters now acting as the input into the network.

4.1. Results for the Forward TSK Network

With the trial and error method mentioned above, different forward TSK networks with 5 inputs and 7 outputs were randomly initialized with different number of reasoning rules. Based on the datasets available, these TSK networks were trained with the same learning data and then tested with the same testing data. The best testing SSE (Sum of Squared Error) for different number of rules are presented in Figure 7.

As it can be seen from Figure 8, the network with 22 reasoning rules achieved the best testing error during the trials. It showed also that networks with more than 25 rules overfit the learning data, which make the testing performance on new data much worse. The detail results for the forward TSK network with 22 rules are presented below.

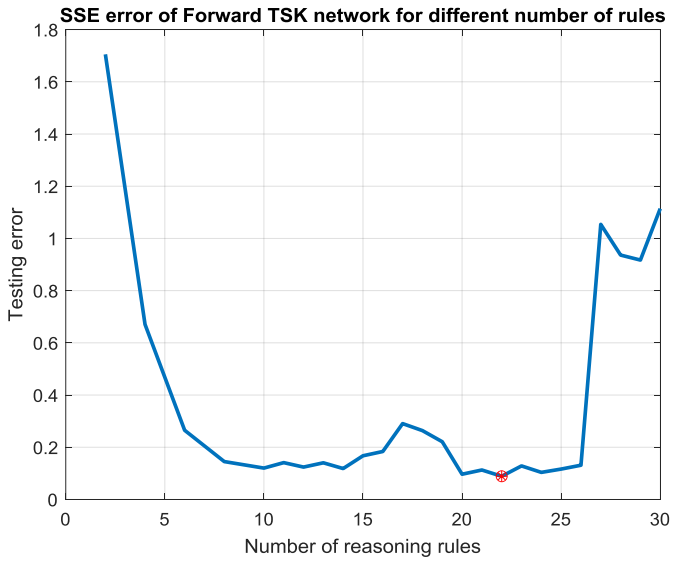


Figure 8. The SSE results of forward TSK networks at different number of reasoning rules

In Figure 9-a and Figure 9-b, the results for the *SF* and t_{500} parameters are shown for 120 test cases. Visually the high correlation between the expected values and the responses from the models can be seen. The detailed numerical errors are listed in Table 1. The model demonstrates high precision for *SF*, with an MRE of only 0.87%. Because *SF* indicates the concrete's ability to spread freely, this high correlation shows the model can accurately forecast fillability

based on ingredient dosages. While visually correlated, t_{500} prediction exhibits a higher MRE of 2.32%, however this corresponding to a maximum of absolute value less than 0.45 s, which is acceptable for Vietnam quick field testing. The result also shows that t_{500} is more sensitive to minor fluctuations in input ingredients.

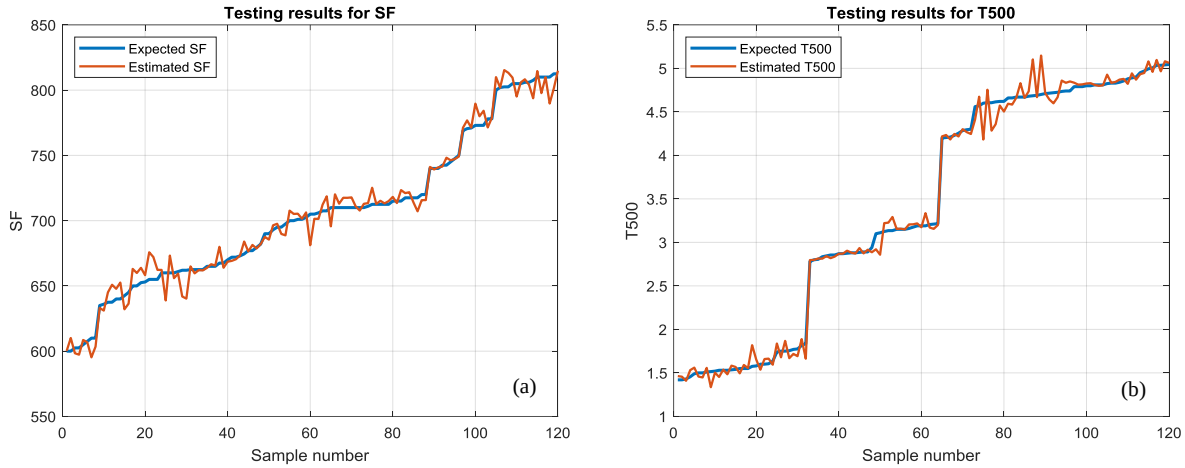


Figure 9. The estimation results of forward network for (a) SF and (b) t500 parameters

In Figures 10-a, 10-b, and 11-a, the results for V_{funnel}, L_{box}, and J_{ring} parameters are shown for 120 test cases. It can also be seen the high correlation between the expected values and the responses from the models. The predictions of both parameters achieved MRE about 0.86%, which validates the network's performance to map the passing ability of SCC to pass through structural reinforcements in complex confined spaces.

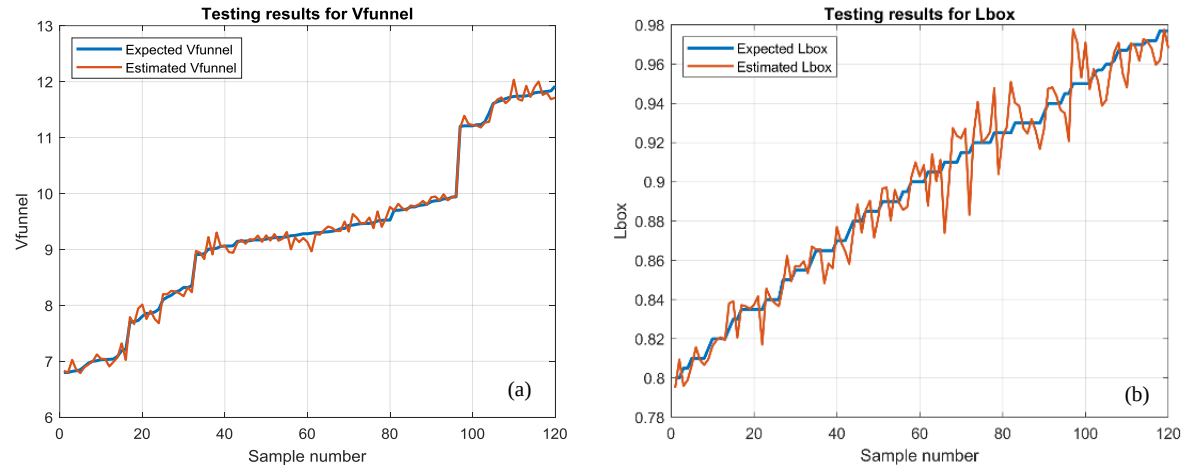


Figure 10. The estimation results of forward network for (a) V_{funnel} and (b) L_{box} parameters

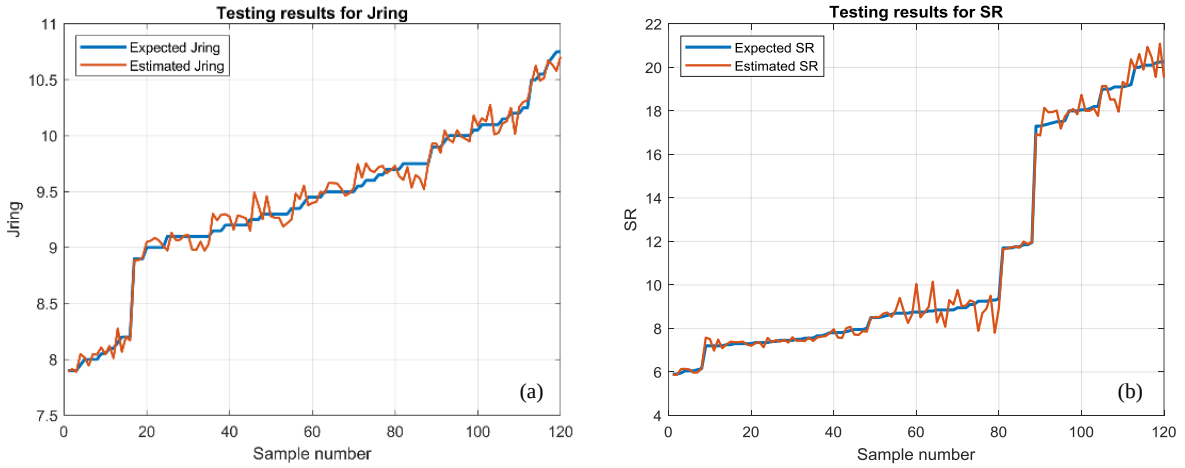


Figure 11. The estimation results of forward network for (a) J_{ring}, (b) SR and (c) R_{28} parameters

In Figures 11-b and 11-c, the results for the parameters *SR* and R_{28} for test cases are shown, where the numerical values are summarized in Table 1. A relative higher error of *SR* shows that this parameter is also sensitive to the variations of the ingredients as the t_{500} parameter. But with a max average error at 1.5, the model can still predict *SR* with very good accuracy for practical application. The last, but also important parameter R_{28} could be predict with the smallest relative error at only 0.7%, confirming that the simplified TSK network is exceptionally robust for predicting the structural integrity of a mixture before it is poured.

Table 1. Approximation errors of forward network for the workability parameters

Parameters	MAE	MRE (%)	Max AE	Max RE (%)
SF	6.0537	0.87%	23.834	3.38%
t_{500}	0.06767	2.32%	0.447	15.45%
V_{funnel}	0.07895	0.86%	0.325	3.50%
L_{box}	0.007833	0.87%	0.036	3.96%
J_{ring}	0.06977	0.74%	0.241	2.61%
SR	0.2688	2.46%	1.503	16.16%
R_{28}	0.3646	0.70%	1.182	2.28%

To evaluate the robustness and efficiency of the proposed sTSK model, several alternative neural networks were benchmarked using the same experimental dataset. These comparative models included a classical TSK fuzzy network with a similar structure of 22 rules, a classical TSK fuzzy network featuring 16 rules with the best testing error, and a Multi-Layer Perceptron (MLP) consisting of a single hidden layer. The best MLP configuration with the lowest testing error has 18 hidden neurons.

As summarized in Table 2, the empirical results indicate that the proposed sTSK model outperforms both the classical TSK and MLP architectures. Both classical TSK variants exhibited higher MAE values, signaling worse generalization capabilities compared to the sTSK network. This performance gap is primarily attributed to the overfitting effect, as the classical TSK models possess a substantially higher number of parameters. In contrast, the sTSK architecture utilizes approximately three times fewer nonlinear parameters, which enhances its ability to model the complex, nonlinear dependencies of SCC ingredients with less tendency to overlearning. Furthermore, the MLP network also demonstrated lower predictive accuracy than the sTSK model, reinforcing the superiority of the proposed simplified neuro-fuzzy framework for industrial applications with limited data samples.

Table 2. Comparison of MAE errors of selected neural networks as the forward model

Parameters	sTSK	Classic TSK with 22 rules	Classic TSK with 16 rules	MLP with 18 hidden neurons
SF	6.0537	6.889	6.750	6.502
t_{500}	0.06767	0.155	0.0964	0.077
V_{funnel}	0.07895	0.177	0.092	0.178
L_{box}	0.007833	0.0173	0.0113	0.0188
J_{ring}	0.06977	0.143	0.0727	0.131
SR	0.2688	0.714	0.474	0.295
R_{28}	0.3646	0.853	0.535	1.015

4.2. Results for the Backward TSK Network

In a similar way to the forward model, the backward TSK network was trained and selected using the same datasets but the roles of input and output vectors were swapped. The backward TSK networks now have 7 inputs and 5 outputs, and the number of reasoning rules were determined using the trial and error method mentioned in Section 3. During this search, the best testing SSE for the backward TSK with different number of rules are presented in Figure 12.

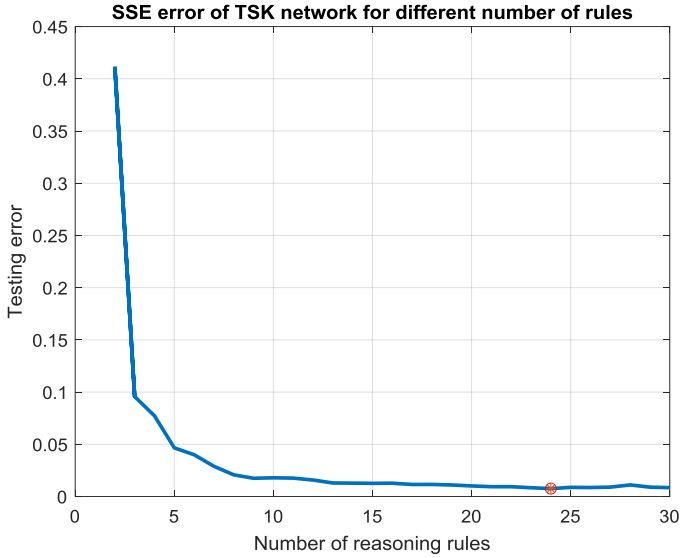


Figure 12. The SSE results of backward TSK networks at different number of rules

Figure 12 illustrates the iterative "trial and error" process used to find the optimal number of rules used in the sTSK network. The results determined that 24 reasoning rules provided the lowest testing SSE. Similar to the forward model, having not enough rules (less than 24) would cause underlearning (underfitting) effect, where the model performs poorly on both learning and training data. Having more than 25 rules, the network becomes more and more complicate, but fails to improve the generalization performance. The detail results for the forward TSK network with 24 rules are presented below (Figure 13).

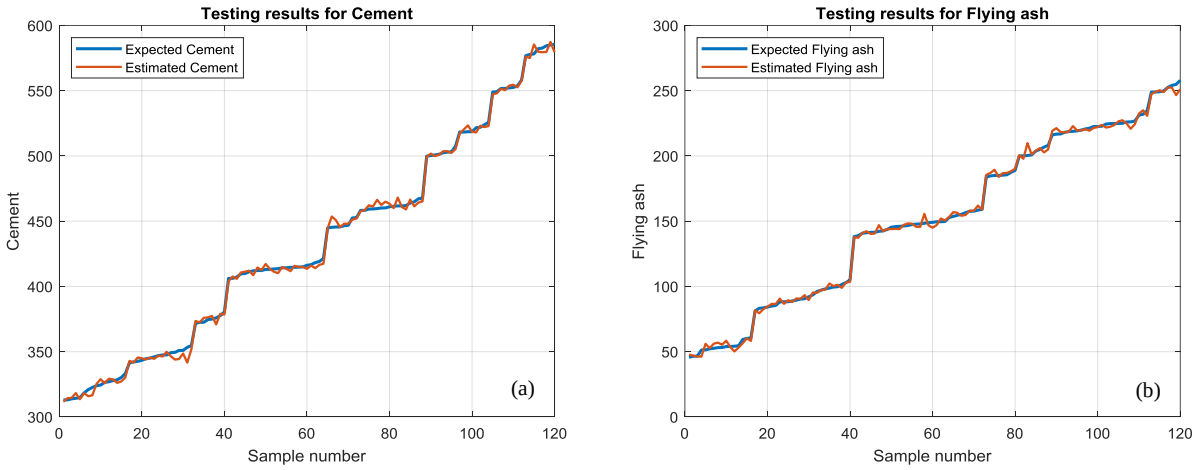


Figure 13. The estimation results of backward network for (a) cement and (b) fly ash parameters

Figures 13-a and 13-b presented the results for the quantity of cement and fly ash needed for the samples. Similar to the results for forward network, a high correlation between the expected values and the responses from the backward models can be seen. The detailed numerical errors are listed in Table 3. The backward model successfully derived the required cement quantities with a MRE only at 0.55%, allowing allows the model to recommend accurately the amount of cement needed for specific performance targets. On the other hand, fly ash estimation had the highest error in the backward model (MRE equal 1.62%), this slight variance may be due to the fact that in our field tests, the storage of the fly ash component was not ideal, which may introduce some changes in the material properties.

Table 3. Approximation errors of backward model for the required ingredients

Parameters	MAE	MRE (%)	Max AE	Max RE (%)
Cement	2.2789	0.55%	11.717	3.32%
Fly ash	1.9204	1.62%	9.443	9.96%
Water	0.7831	0.40%	5.095	2.56%
Super ductile	0.02705	0.46%	0.175	2.91%
VMA	0.001640	0.79%	0.007	3.26%

Figures 14-a to 14-c presented the results for water, super ductile and VMA ingredients of the 120 test cases, with the MRE errors were 0.4%, 0.46% and 0.79% respectively. Given that water content is one of the most influential factors on SCC workability and environmental sensitivity, this precision is vital for effective mixture design. Overall, the backward model can recommend all the selected five input ingredients with very high accuracy to achieve target values of all seven workability parameters.

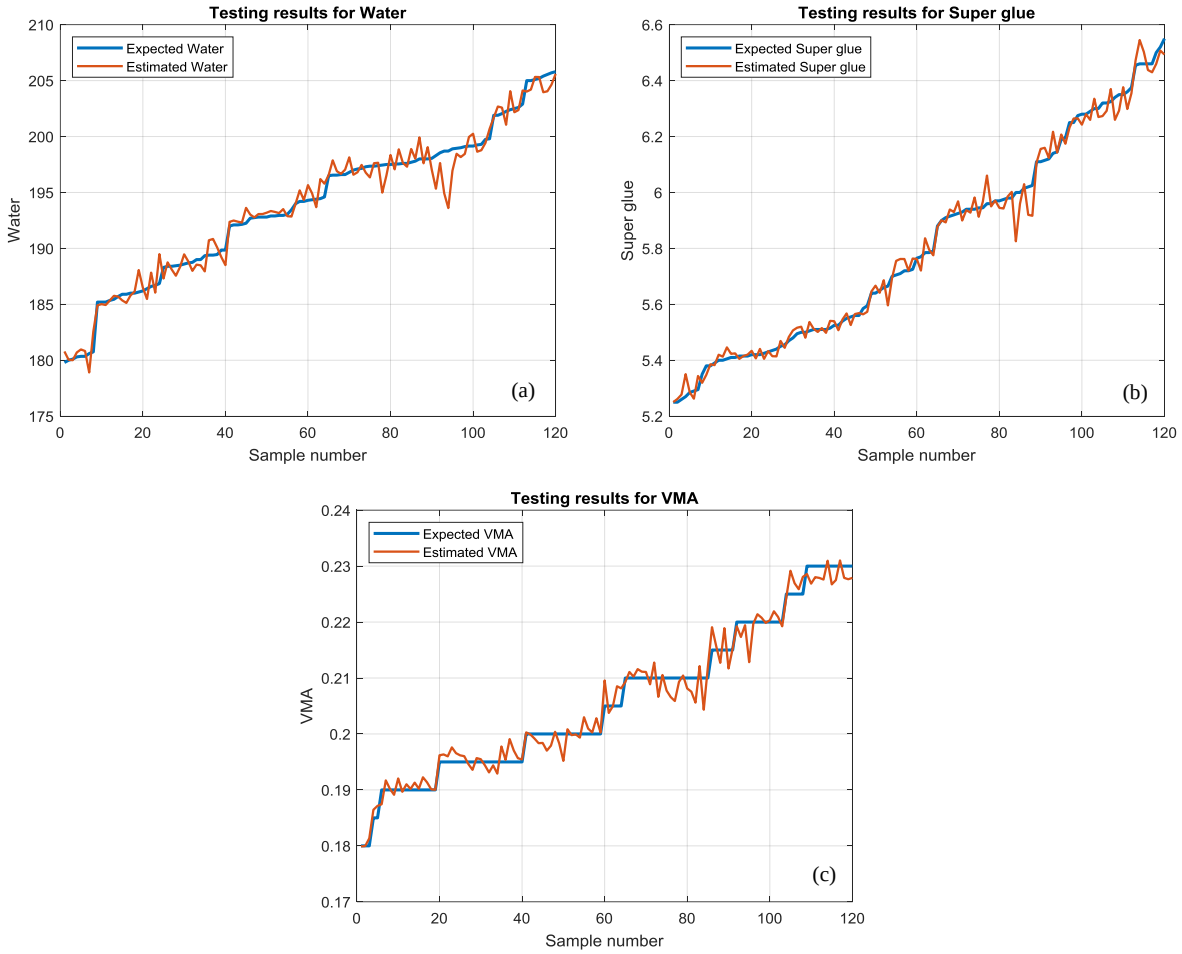


Figure 14. The estimation results of backward network for (a) water, (b) super-ductile additive and (c) VMA parameters

For the backward model, the sTSK network was tested also against a classical TSK network and an MLP network based on the same data set. The classical TSK with lowest testing error has 20 reasoning rules, where the MLP network has one hidden layer with 16 hidden neurons. The values of MAE for each output and each network are presented in Table 4. It can be seen that the sTSK has outperformed both classical TSK and MLP as a backward model as well.

Table 4. Comparison of MAE errors of selected neural networks as the backward model

Parameters	sTSK	Classic TSK with 20 rules	MLP with 16 hidden neurons
Cement	2.2789	2.4916	2.927
Fly ash	1.9204	2.1730	2.394
Water	0.7831	0.8221	0.804
Super ductile	0.02705	0.0293	0.0346
VMA	0.001640	0.001956	0.002164

To evaluate the inverse estimation performance of the proposed framework, the backward sTSK network was also tested against a classical TSK network and a Multi-Layer Perceptron (MLP) architecture using the same experimental dataset. In this comparative analysis, the classical TSK model with the lowest testing error was configured with 20 reasoning rules, while the MLP network featured a single hidden layer with 16 neurons. As detailed in Table 4, which presents the MAE for each output across the three models, the sTSK architecture consistently outperformed both the classical TSK and the MLP networks. This better performance demonstrates that the simplified neuro-fuzzy structure is a more effective backward estimator, providing higher precision in reverse-engineering mixture proportions while maintaining better generalization properties than parameter-heavy classical models.

An analysis of the consolidated data from Tables 1 to 4 indicates that the sTSK neuro-fuzzy networks demonstrate very high accuracy in the assigned prediction tasks. Such good levels of predictive accuracy confirm that the proposed bidirectional sTSK architecture is effectively capable of mapping the intricate nonlinear relationships between the primary SCC input components and their corresponding workability characteristics.

5. Conclusion

This paper proposed and validated a bidirectional modeling framework using simplified TSK networks to approximate the complex, nonlinear dependencies between self-compacting concrete (SCC) ingredients and their performance parameters. Based on a comprehensive dataset of 480 field-tested mixtures, numerical results demonstrated that the forward modeling phase yielded an MRE of less than 2.5% across all seven performance parameters, whereas the backward network exhibited even greater precision, with MRE values remaining below 1.62% for all predicted ingredient quantities. These results confirm the models' ability to accurately reverse-engineer mixture proportions, specifically cement, fly ash, water, and chemical admixtures, to meet targeted performance goals.

A great advancement of this work is the use of the sTSK model, which replaces Euclidean distance with Mahalanobis distance to create hyperellipsoidal activation areas to allow the network to operate with approximately three times fewer nonlinear parameters than classical versions, significantly streamlining the training process and enhancing implementation feasibility in real-world production environments where computational efficiency is essential for on-site quality control.

Overall, the bidirectional sTSK framework provides a promising, cost-effective alternative to time-consuming iterative physical testing, offering a scalable tool for industrial optimization. The framework can be extended for prediction of other concrete components and in both forward and backward problems, which enable the identification of input mixture components that correspond to an expected specific set of workability parameters for SCC mixtures. In practical applications, these models can be extended to allow users to quickly and efficiently predict other parameters of SCC in particular and other industrial products in general for real-time quality control and efficient mixture optimization within industrial production environments. It is important to emphasize that these models, like other supervised learning models, require retraining when the dataset is expanded to include new cases that differ significantly from the original training data.

6. Declarations

6.1. Author Contributions

Conceptualization, D.V.D., R.S., and L.H.T.; methodology, D.V.D. and L.H.T.; software, L.H.T.; validation, D.V.D., R.S., C.H.N., and L.H.T.; formal analysis, D.V.D. and L.H.T.; investigation, C.H.N. and L.H.T.; resources, C.H.N. and L.H.T.; writing—original draft preparation, D.V.D. and L.H.T.; writing—review and editing, D.V.D., R.S., C.H.N., and L.H.T.; visualization, L.H.T.; supervision, D.V.D. and L.H.T.; project administration, D.V.D. All authors have read and agreed to the published version of the manuscript.

6.2. Data Availability Statement

The data presented in this study are available on request from the corresponding author.

6.3. Funding

The authors received no financial support for the research, authorship, and/or publication of this article.

6.4. Conflicts of Interest

The authors declare no conflict of interest.

7. References

[1] Okamura, H., & Ouchi, M. (2003). Self-Compacting Concrete. Journal of Advanced Concrete Technology, 1(1), 5–15. doi:10.3151/jact.1.5.

[2] Brouwers, H. J. H., & Radix, H. J. (2005). Self-compacting concrete: Theoretical and experimental study. Cement and Concrete Research, 35(11), 2116–2136. doi:10.1016/j.cemconres.2005.06.002.

[3] Tattersall, G. H. (1991). Workability and Quality Control of Concrete. CRC Press, London, United Kingdom.doi:10.1201/9781482267006.

[4] Rizal, N. S., Arifi, E., & Mufarida, N. A. (2025). High Initial Concrete Compressive Strength with Variations of Superplasticizer and Silica Fume Additions. Civil Engineering Journal (Iran), 11(1), 107–119. doi:10.28991/CEJ-2025-011-01-07.

[5] Irshidat, M. R., & Kailani, H. Y. (2025). Sustainable Utilization of Recycled Concrete Powder as Sand Replacement in Cement Mortar Production: Impact of Sand-Cement Ratio. Civil Engineering Journal (Iran), 11(8), 3375–3394. doi:10.28991/CEJ-2025-011-08-016.

[6] Hamdouni, S., Benaicha, M., Alaoui, A. H., & Burtschell, Y. (2025). Optimizing self-compacting concrete: A comprehensive analysis of weighting methods and multi-response optimization. Results in Engineering, 26. doi:10.1016/j.rineng.2025.105138.

[7] Cui, T., Kulasegaram, S., & Li, H. (2024). Design automation of sustainable self-compacting concrete containing fly ash via data driven performance prediction. Journal of Building Engineering, 87, 108960. doi:10.1016/j.jobe.2024.108960.

[8] Yeh, I. C., & Chen, J. W. (2005). Modeling workability of concrete using design of experiments and artificial neural networks. Journal of Technology, 20(2), 153-162.

[9] Hu, X., Li, B., Mo, Y., & Alselwi, O. (2021). Progress in artificial intelligence-based prediction of concrete performance. Journal of Advanced Concrete Technology, 19(8), 924–936. doi:10.3151/jact.19.924.

[10] Gou, J., Zaman, A., & Farooq, F. (2025). Machine learning-based prediction of compressive strength in sustainable self-compacting concrete. Engineering Applications of Artificial Intelligence, 161(B). doi:10.1016/j.engappai.2025.112190.

[11] Sobhani, J., Najimi, M., Pourkhorshidi, A. R., & Parhizkar, T. (2010). Prediction of the compressive strength of no-slump concrete: A comparative study of regression, neural network and ANFIS models. Construction and Building Materials, 24(5), 709–718. doi:10.1016/j.conbuildmat.2009.10.037.

[12] Zhang, Y., Yuan, X., Zhang, X., Wang, H., He, P., Luo, L., & Xu, C. (2025). Prediction of high-performance concrete compressive strength using Decision Tree-Guided Artificial Neural Network Pretraining approach. Engineering Applications of Artificial Intelligence, 156(A), 110828. doi:10.1016/j.engappai.2025.110828.

[13] Singh, T., Singh, R. B., Kapoor, K., & Singh, S. P. (2025). Calculating plastic viscosity of self-compacting concrete using V-funnel time: a novel mathematical approach with calibration using coefficient of viscosity. Discover Civil Engineering, 2(1), 201. doi:10.1007/s44290-025-00354-1.

[14] Nguyen, D. B., Wu, C. J., & Liao, W. C. (2023). Experimental Study and Development of Prediction Model of Self-Compacting Concrete Incorporating Fly Ash on Compressive Creep Behaviour. Journal of Advanced Concrete Technology, 21(8), 596–610. doi:10.3151/jact.21.596.

[15] Karthikeyan, J., Upadhyay, A., & Bhandari, N. M. (2008). Artificial neural network for predicting creep and shrinkage of high performance concrete. Journal of Advanced Concrete Technology, 6(1), 135–142. doi:10.3151/jact.6.135.

[16] Naadia, T., Ghernouti, Y., & Gueciouer, D. (2020). Development of a measuring procedure of rheological behavior for self compacting concrete. Journal of Advanced Concrete Technology, 18(7), 328–338. doi:10.3151/jact.18.328.

[17] TCVN 12209:2018. (2018). Self-compacting Concrete – Specification And Test Method. National Standards of Vietnam, Hanoi, Vietnam. (In Vietnamese).

[18] Nguyen, C. H., & Tran, L. H. (2023). Applications of Neural Network and Neuro-Fuzzy Network To Estimate the Parameters of Self-Compacting Concrete. International Journal of GEOMATE, 24(106), 120–129. doi:10.21660/2023.106.3656.

[19] Abood, E. A., Thoeny, Z. A., Azize, N. M., Imran, H., Kamchoom, V., Onyelowe, K. C., & Arunachalam, K. P. (2025). Self compacting concrete with recycled aggregate compressive strength prediction based on gradient boosting regression tree with Bayesian optimization hybrid model. Scientific Reports, 15(1), 28175. doi:10.1038/s41598-025-11161-0.

[20] Shukla, N. K., Garg, A., Bhutto, J., Aggarwal, M., Abbas, M., Hussein, H. S., Verma, R., & Khan, T. M. Y. (2023). Predicting the compressive strength of SCC containing nano silica using surrogate machine learning algorithms. Computers and Concrete, 32(4), 373–381. doi:10.12989/cac.2023.32.4.373.

[21] Golafshani, E. M., & Pazouki, G. (2018). Predicting the compressive strength of self-compacting concrete containing fly ash using a hybrid artificial intelligence method. Computers and Concrete, 22(4), 419–437. doi:10.12989/cac.2018.22.4.419.

[22] Le Nguyen, K., Shakouri, M., & Ho, L. S. (2025). Investigating the effectiveness of hybrid gradient boosting models and optimization algorithms for concrete strength prediction. Engineering Applications of Artificial Intelligence, 149, 110568. doi:10.1016/j.engappai.2025.110568.

[23] Serraye, M., Kenai, S., & Boukhatem, B. (2021). Prediction of compressive strength of self-compacting concrete (SCC) with silica fume using neural networks models. Civil Engineering Journal (Iran), 7(1), 118–139. doi:10.28991/cej-2021-03091642.

[24] el Asri, Y., Benaicha, M., Zaher, M., & Hafidi Alaoui, A. (2022). Prediction of the compressive strength of self-compacting concrete using artificial neural networks based on rheological parameters. Structural Concrete, 23(6), 3864–3876. doi:10.1002/suco.202100796.

[25] Padavala, S. S. A. B., Avudaiappan, S., Paluri, Y., Bharath, C. N., Prathipati, S. R. R. T., & Kumara, A. M. (2025). Self-compacting concrete with fly ash and silica fume: experimental evaluation, microstructural analysis, and machine learning modeling. Scientific Reports, 15(1), 45146. doi:10.1038/s41598-025-33180-7.

[26] De Smedt, J., & De Weerdt, J. (2024). Predictive process model monitoring using long short-term memory networks. Engineering Applications of Artificial Intelligence, 133(D), 108295. doi:10.1016/j.engappai.2024.108295.

[27] Yang, L., & An, X. (2020). Estimating the workability of self-compacting concrete in different mixing conditions based on deep learning. Computers and Concrete, 25(5), 433–445. doi:10.12989/cac.2020.25.5.433.

[28] Saha, P., Prasad, M. L. V., & RathishKumar, P. (2017). Predicting strength of SCC using artificial neural network and multivariable regression analysis. Computers and Concrete, 20(1), 31–38. doi:10.12989/cac.2017.20.1.031.

[29] Seshaiah, B., Rao, P. S., & Rao, P. S. (2021). Effect of mineral admixtures on the properties of steel fibre reinforced SCC proportioned using plastic viscosity and development of regression & ANN model. Computers and Concrete, 27(6), 523–535. doi:10.12989/cac.2021.27.6.523.

[30] Haykin, S. (2009). Neural networks and learning machines (3rd Ed.). Pearson Education India, Chennai, India.

[31] Tran, H. L. (2002). The modification of TSK network in neuro-fuzzy systems, Proceeding of International Conference on Fundamentals of Electrotechnics and Circuit Theory (IC-SPETO), Poland.

[32] Sankhya, A. (2018). Reprint of: Mahalanobis, P.C. (1936). On the Generalised Distance in Statistic, 80(1), 1–7. doi:10.1007/s13171-019-00164-5.