Research Article

A Novel Time Series Prediction Approach Based on a Hybridization of Least Squares Support Vector Regression and Swarm Intelligence

Nhat-Duc Hoang,[1] Anh-Duc Pham,[2] and Minh-Tu Cao[3]

[1] *Institute of Research and Development, Faculty of Civil Engineering, Duy Tan University,*
  P809-K7/25 Quang Trung, Danang 59000, Vietnam
[2] *Faculty of Project Management, The University of Danang, University of Science and Technology,*
  54 Nguyen Luong Bang, Danang 59000, Vietnam
[3] *Department of Construction Engineering, National Taiwan University of Science and Technology,*
  No. 43, Section 4, Keelung Road, Daan District, Taipei 8862, Taiwan

Correspondence should be addressed to Nhat-Duc Hoang; hoangnhatduc@dtu.edu.vn

This research aims at establishing a novel hybrid artificial intelligence (AI) approach, named as firefly-tuned least squares support vector regression for time series prediction (FLSVR$_{TSP}$). The proposed model utilizes the least squares support vector regression (LS-SVR) as a supervised learning technique to generalize the mapping function between input and output of time series data. In order to optimize the LS-SVR's tuning parameters, the FLSVR$_{TSP}$ incorporates the firefly algorithm (FA) as the search engine. Consequently, the newly construction model can learn from historical data and carry out prediction autonomously without any prior knowledge in parameter setting. Experimental results and comparison have demonstrated that the FLSVR$_{TSP}$ has achieved a significant improvement in forecasting accuracy when predicting both artificial and real-world time series data. Hence, the proposed hybrid approach is a promising alternative for assisting decision-makers to better cope with time series prediction.

1. Introduction

Generally, time series forecasting involves the prediction of future values of data based on discovering the pattern in the historical data series and extrapolating that pattern into the future. Time series forecasting is a widely discussed issue and its applications appear in various fields of business and engineering [1]. The reason is that prediction of future events is crucial for many kinds of planning and decision-making processes. Applications regarding time series data can be easily found in the literature, such as wind energy forecasting [2], water resource management [3], traffic accident prediction [4], and cash flow forecasting in construction projects [5]. Hence, it is not surprising that time series analyses and predictions are on the rise among researchers.

Notably, constructing a predictive model for time series forecasting is a challenging task. It is because real-world time series data are often characterized by nonlinearity, being nonstationary, and irregularity [6]. Random noise and effect of unidentified factors are the main causes that degrade the prediction accuracy. Moreover, in most cases, the underlying model that generates the series is unknown and the process of discovering such model is oftentimes hindered by the stochastic nature of the time-dependent data [7]. Particularly, for each time series, determination of a suitable embedding dimension is also of major concern [8, 9]. Therefore, these challenges necessitate the development of advanced approaches.

Over the recent years, there has been increasing efforts dedicated in establishing AI based models to predict real-world time-dependent data. Various AI approaches such as the artificial neural network (ANN), adaptive network based fuzzy inference system (ANFIS), support vector machine

(SVM), and least squares support vector machine (SVM) approaches have been applied to cope with time series prediction in various domains [2, 4, 10]. These previous works have illustrated that application of these techniques, as a solution to the challenges of time series problems, is not only feasible but also very effective.

Among the AI methods, the least squares support vector regression (LS-SVR) is an advanced machine learning technique for solving regression analysis [11]. This method has been proved to possess many advanced features [12, 13]. In the LS-SVR's training process, a least squares cost function is proposed to obtain a linear set of equations in the dual space. Consequently, to derive the solution, it is required to solve a set of linear equations, instead of the quadratic programming as in the standard SVM. Moreover, this linear system can be efficiently solved by iterative methods such as conjugate gradient.

Studies have been carried out to demonstrate the excellent generalization, prediction accuracy, and fast computation of the LS-SVR [13–15]. Since time series forecasting can be formulated as a regression analysis problem, it is very potential to apply the LS-SVR to tackle the problem at hand. Nevertheless, the implementation of the LS-SVR requires an appropriate setting of its tuning parameters, namely, the regularization parameter and the kernel function parameter. Improper specification of these tuning parameters can significantly degrade the performance of the machine learning technique.

In the field of AI, the task of parameter setting is well-known as the model selection process [16]. This problem is critical and it has increasingly drawn attentions of many scholars in a variety of disciplines [2, 14, 17]. In practice, identifying the most suitable set of model's parameters often requires either prior knowledge of the problem domain or tedious trial-and-error processes. To overcome this issue, hybridizing the machine learning techniques with a swarm-based optimization algorithm is a feasible resolution for the problem at hand [16, 18].

Swarm intelligence is a design framework based on social insect behavior [19]. Social insects such as ants, bees, firefly, and wasps are unique in the way these simple individuals cooperate to accomplish complex, difficult tasks. This cooperation is distributed among the entire population, without any centralized control. Each individual simply follows a small set of rules influenced by locally available information. This emergent behavior results in great achievements that no single member could complete by themselves [20]. Firefly algorithm (FA) is one of the recent swarm intelligence methods which was based on the flashing patterns and behavior of tropical fireflies [21]. According to previous works, the algorithm is very efficient and can outperform conventional algorithms in solving many optimization problems [22, 23].

Therefore, the purpose of this study is to fuse the LS-SVR and FA techniques to establish a new hybrid AI model for prediction of time series. Our research goal is to build a model that possesses the capability of delivering accurate as well as operating autonomously without human interference. The second section of this paper reviews the methods needed to accomplish the research objective. In the third section, the framework of the proposed $\text{FLSVR}_{\text{TSP}}$ is described in detail. The fourth section demonstrates the experimental results. Conclusion on our study is mentioned in the final section.

2. Literature Review

2.1. Time Series Prediction. Time series forecasting is an important subject in which past observations of an interested variable are recorded and analyzed to establish a prediction model [24]. The developed model is built with the expectation that it can describe the underlying relationship between patterns in the past and value of the variable in the future (see Figure 1). At the current time t, and given the recorded observations of a time series, $X_t, X_{t-h}, X_{t-2h}, \ldots, X_{t-mh}$ the task is to forecast the future value X_{t+h}. Herein, m represents the embedding dimension; h denotes the forecasting horizon. If h is one, the problem is known as single-step-ahead forecast. Meanwhile, problem involved greater value of h is often referred as multiple-step-ahead forecast [25].

Generally, in time series prediction, the historical time series are transformed into high dimensional space to facilitate the exploration of implicit pattern lying in the series. This process of transformation, widely known as state reconstruction [8, 9], is dependent on the embedding dimension (m). Equation (1) illustrates the state reconstruction process for one-step-ahead forecasting in which the original time series D (with N element) is transformed into an input matrix X of the size $(N - m)$-by-m and an output matrix Y of the size $(N - m)$-by-1:

$$
\text{Time series}
\begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_{N-1} \\ X_N \end{bmatrix}
\atop D
$$

$$
\xrightarrow{m}
\overset{\text{Input}}{\begin{bmatrix} X_1 & X_2 & \cdots & X_m \\ X_2 & X_3 & \cdots & X_{m+1} \\ \vdots & \vdots & \vdots & \vdots \\ X_{N-m} & X_{N-(m-1)} & \cdots & X_{N-1} \end{bmatrix}}_{X}
\overset{\text{Desired output}}{\begin{bmatrix} X_{m+1} \\ X_{m+2} \\ \vdots \\ X_N \end{bmatrix}}_{Y} .
$$

$$(1)$$

In time series analysis, the parameter m is crucial because of its influence on the prediction performance. For each time series data, the embedding dimension can be calculated using the "false nearest neighbor" (FNN) approach established by Kennel et al. [26]. However, from the perspective of machine learning, this parameter can play the role as a tuning parameter in the prediction model and its optimal value can be searched by an optimization technique [27].

2.2. Least Squares Support Vector Regression (LS-SVR). This section of the paper describes the mathematical formulation
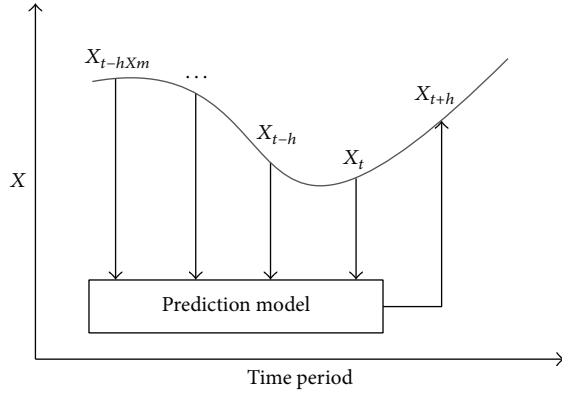
FIGURE 1: Time series prediction model.

of the LS-SVR. Consider the following model of interest, which infers the mapping between a response variable and one or more independent variables [11, 13, 28]:

$$y(x) = w^T \phi(x) + b, \tag{2}$$

where $x \in R^n$, $y \in R$, and $\phi(x) : R^n \rightarrow R^{nh}$ is the mapping to the high dimensional feature space.

In LS-SVR for regression analysis, given a training dataset $\{x_k, y_k\}_{k=1}^N$, the optimization problem is stated as follows:

$$\text{Minimize} \quad J_p(w, e) = \frac{1}{2} w^T w + \gamma \frac{1}{2} \sum_{k=1}^N e_k^2 \tag{3}$$

$$\text{Subjected to} \quad y_k = w^T \phi(x_k) + b + e_k, \quad k = 1, \ldots, N,$$

where $e_k \in R$ are error variables; $\gamma > 0$ denotes a regularization constant.

In (3), it is noticed that the objective function is composed of a sum of squared fitting error and a regularization term. This cost function is similar to standard procedure in training feedforward neural networks and is related to a ridge regression. However, when w becomes infinite, one cannot solve this primal problem. Hence, it is necessary to establish the Lagrangian and derive the corresponding dual problem.

The Lagrangian is given as follows:

$$L(w, b, e; \alpha)$$
$$= J_p(w, e) - \sum_{k=1}^N \alpha_k \left\{ w^T \phi(x_k) + b + e_k - y_k \right\}, \tag{4}$$

where α_k are Lagrange multipliers. The conditions for optimality are given by:

$$\frac{\partial L}{\partial w} = 0 \longrightarrow w = \sum_{k=1}^N \alpha_k \phi(x_k),$$

$$\frac{\partial L}{\partial b} = 0 \longrightarrow \sum_{k=1}^N \alpha_k = 0,$$

$$\frac{\partial L}{\partial e_k} = 0 \longrightarrow \alpha_k = \gamma e_k, \quad k = 1, \ldots, N,$$

$$\frac{\partial L}{\partial \alpha_k} = 0 \longrightarrow w^T \phi(x_k) + b + e_k - y_k = 0, \quad k = 1, \ldots, N. \tag{5}$$

After elimination of e and w, (5) can be represented as the following linear system:

$$\begin{bmatrix} 0 & 1_v^T \\ 1_v & \omega + \dfrac{I}{\gamma} \end{bmatrix} \begin{bmatrix} b \\ \alpha \end{bmatrix} = \begin{bmatrix} 0 \\ y \end{bmatrix}, \tag{6}$$

where $y = y_1, \ldots, y_N$, $1_v = [1; \ldots; 1]$, and $\alpha = [\alpha_1; \ldots; \alpha_N]$.

And the kernel function is applied as follows:

$$\omega = \phi(x_k)^T \phi(x_l) = K(x_k, x_l). \tag{7}$$

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

$$y(x) = \sum_{k=1}^N \alpha_k K(x_k, x_l) + b, \tag{8}$$

where α_k and b are the solution to the linear system (6). The kernel function that is often utilized is radial basis function (RBF) kernel. Description of RBF kernel is given as follows:

$$K(x_k, x_l) = \exp\left(\frac{\|x_k - x_l\|^2}{2\sigma^2} \right), \tag{9}$$

where σ is the kernel function parameter.

When the RBF kernel is used, there are two tuning parameters (γ, σ) that are needed to be determined in LS-SVR. The regularization parameter (γ) controls the penalty imposed to data points that deviate from the regression function. Meanwhile, the kernel parameter (σ) influences the smoothness of the regression function. It is worth noticing that proper setting of these tuning parameters is required to achieve desirable performance of the prediction model.

2.3. Firefly Algorithm (FA). The FA is a stochastic, nature-inspired, and metaheuristic algorithm that can find both the global optima and the local optima simultaneously and effectively [21]. The flashing lights of fireflies are an amazing sight in the summer sky in tropical and temperate regions. The pattern of flashes is often unique for a particular species. In essence, each firefly is attracted to brighter ones as it randomly explores while searching for prey.

The FA algorithm uses the following three idealized rules: (1) all fireflies are unisex, so each firefly is attracted to other fireflies regardless of their sex, (2) the attractiveness of a firefly is proportional to its brightness and decreases as the distance increases. A firefly moves randomly if no other firefly is brighter, and (3) the brightness of a firefly is affected or determined by the landscape of the objective function [22, 29]. The FA algorithm can be illustrated in Pseudocode 1.

```
Begin FA
Define objective function f(x), where x = (x₁,...,x_d)
Generate an initial population of fireflies
Formulate the light intensity I
Define the absorption coefficient γ
While (t < Max_Generation)
For i = 1 to n (all n fireflies)
     For j = 1 to n (all n fireflies)
         If (I_j > I_i), move firefly i towards firefly j
         End if
         Evaluate new solutions and update light intensity;
     End for j
End for i
Rank the fireflies and find the current best
End while;
End FA
```

PSEUDOCODE 1: FA pseudocode.

The brightness of an individual firefly can be defined similarly to the fitness value in the genetic algorithm [30]. The light intensity $I(r)$ varies according to the inverse square law as follows:

$$I(r) = \frac{I_s}{r^2}, \tag{10}$$

where I_s = the light intensity at the source. For a given medium with a fixed light absorption coefficient γ, the light intensity I varies with the distance r. Thus, the light intensity I can be computed in the following way:

$$I = I_S e^{-\gamma r}. \tag{11}$$

The combined effect of both the inverse square law and absorption can be approximated as the following Gaussian form as follows:

$$I(r) = I_S e^{-\gamma r^2}. \tag{12}$$

As the attractiveness of a firefly is proportional to the light intensity seen by adjacent fireflies, the attractiveness β of a firefly is defined as:

$$\beta = \beta_0 e^{-\gamma r^2}. \tag{13}$$

The distance between any two fireflies i and j at x_i and x_j, respectively, is the Cartesian distance as follows:

$$r_{ij} = \|x_i - x_j\| = \sqrt{\sum_{k=1}^{d} (x_{i,k} - x_{j,k})^2}$$
$$= \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \quad \text{(in 2D case)}. \tag{14}$$

The movement of the ith firefly when attracted to another more attractive (brighter) jth firefly is as follows:

$$x_i = x_i + \beta_0 e^{-\gamma r_{ij}^2} (x_j - x_i) + \alpha (\text{rand} - 0.5), \tag{15}$$

where x_i^{g+1} and x_i^g represent the position of the flies i at $g + 1$ and g generations. x_j^g denotes the position of the flies i at $g + 1$ and g generations γ is absorption coefficient and typically varies from 0.1 to 10 in most application; β_0 = the attractiveness at $r_{ij} = 0$; $\alpha = a$ trade-off constant to determine the random behavior of movement; rand represents a random number drawn from Gaussian distribution. In essence, (15) describes the mechanism for updating a firefly in the current population. The movement of a firefly towards another firefly is dependent upon to the attractiveness β and a quantity α (rand $- 0.5$) that reflect the randomness in animal behavior.

3. The Proposed Firefly-Tuned Least Squares Support Vector Regression for Time Series Prediction (FLSVR$_{\text{TSP}}$)

This section dedicates in describing the proposed prediction model, named as FLSVR$_{\text{TSP}}$, in detail. The establishment of the model (see Figure 2) is accomplished by a fusion of the LS-SVR and FA algorithms. The FLSVR$_{\text{TSP}}$ employs the LS-SVR as the supervised learning algorithm for mining the implicit patterns in the series. Furthermore, the FA, an evolutionary optimization algorithm, is utilized to automatically identify the optimal values of tuning parameters. The construction of the prediction model is dependent on a set of tuning parameters. The embedding dimension (m) is needed in the state reconstruction process. The regularization parameter (γ) and the kernel function parameter (σ) are required for the FLSVR$_{\text{TSP}}$.

(1) Input Data. The FLSVR$_{\text{TSP}}$ takes a univariate time series as its input. The data can be recorded at regular time interval, for example, daily, monthly, and quarterly, and so forth. The whole data set is divided into training set, validating set and testing set. In our study, the ratio of the validating set to the training set is 1/5.

(2) Tuning Parameter Initialization. The aforementioned tuning parameters of the model are randomly generated within the range of lower and upper boundaries (see Table 1) in the following manner:

$$X_{i,0} = \text{LB} + \text{rand}[0, 1] \times (\text{UB} - \text{LB}), \tag{16}$$

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

(3) State Reconstruction. With the embedding dimension m being specified, the time series is transformed to the input matrix X and the desired output vector Y (see (1)). After being transformed, the data is used for the LS-SVR's training process.

(4) LS-SVR Training. In this step, the LS-SVR is deployed to learn the mapping function between the input (X) and the output (Y) derived at the previous step. The training process requires the two parameters γ and σ that are acquired from
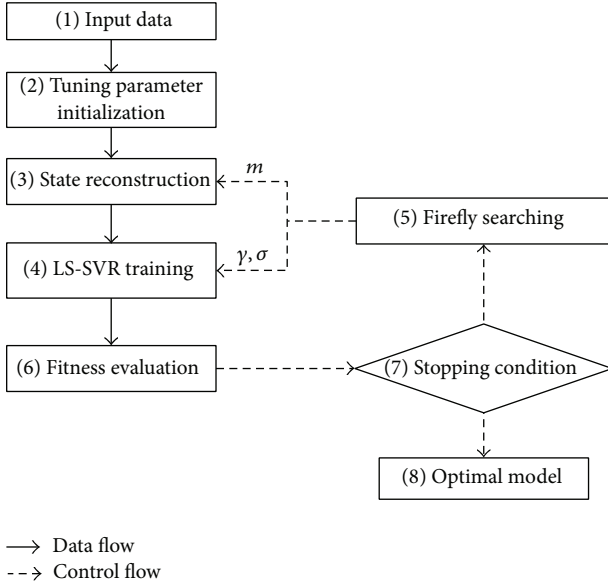
FIGURE 2: Firefly-tuned least squares support vector regression for time series prediction (FLSVR$_\text{TSP}$).

TABLE 1: Ranges of the FLSVR$_\text{TSP}$'s tuning parameters.

Tuning parameter	Notation	Lower bound	Upper bound
Embedding dimension	m	1	20
Regularization parameter	γ	0.001	10000
Kernel function parameter	σ	0.001	1000

the FA searching. The regularization parameter (γ) controls the penalty imposed to data points that deviate from the regression function. Meanwhile, the kernel parameter (σ) affects the smoothness of the regression function. It is worth noticing that proper setting of these tuning parameters is required to ensure desirable performance of the prediction model.

(5) FA Searching. The FA automatically explores the various combinations of the tuning parameters (γ and σ). At each generation, the optimizer carries out the mutation, crossover, and selection processes to guide the population to the optimal solution. By evaluating the fitness of each individual, the algorithm discards inferior combinations of γ and σ, and allows robust combinations of these parameters to be passed on the next generations.

(6) Fitness Evaluation. In the FLSVR$_\text{TSP}$, in order to determine the optimal set of tuning parameters, the following objective function is used in the step of fitness function evaluation:

$$F_\text{fitness} = E_\text{tr} + E_\text{va}. \qquad (17)$$

In (17), E_tr and E_va denotes the training error and validating error, respectively. The training and validating errors

herein are root mean squared error (RMSE) calculated as follows:

$$\text{RMSE} = \sqrt{\sum_{t=1}^{N} \frac{\left(Y_P^t - Y_A^t\right)^2}{N}}, \qquad (18)$$

where Y_P^t and Y_A^t denote predicted and actual value for output tth. In addition, N is the number of data points.

The fitness function, in essence, represents the trade-off between model generalization and model complexity. It is worth noticing that well-fitting of the training set may reflect the model complexity. However, complex model tends to suffer from over-fitting [31]. Thus, incorporating the error of the validating data can help identify the model that features the balance of minimizing training error and generalization property.

(7) Stopping Condition. The FA's optimization process terminates when the maximum number of generation is achieved. If the stopping condition has not met, the FA continues it searching progress.

(8) Optimal Prediction Model. When the program terminates, the optimal set of tuning parameters has been successfully identified. The FLSVR$_\text{TSP}$ is ready to carry out forecasting tasks.

4. Experimental Results

In this section, the newly developed FLSVR$_\text{TSP}$ is applied to forecast three time series: the Mackey-Glass series, the daily water discharge at Palo Verde drain (http://water-data.usgs.gov/), and the monthly USD/TWD exchange rate (http://fx.sauder.ubc.ca/data.html). The Mackey-Glass chaotic time series is defined by (19) [10]. Herein, the parameter τ is set to be 17. In our study, 500 data cases are generated in which 400 cases are used for training and validating process. The rest of the data is used for testing the model as follows:

$$\frac{dx(t)}{dt} = 0.2 \frac{x(t-\tau)}{1 + x(t-\tau)^{10}} - 0.1x(t). \qquad (19)$$

The daily water flow data set consists of 273 data cases of daily water discharge (cubic feet per second) at Palo Verde outfall drain, from 1/1/2011 to 9/30/2011 (see Figure 3). The number of data cases used for testing is 30. The monthly USD/TWD exchange rate includes 260 records from 1/1990 to 8/2011 (see Figure 4). In the experiment, 36 data cases are utilized for testing process. For these two time series, one-step-ahead prediction is carried out.

Moreover, the back propagation neural network (BPNN), the adaptive network based fuzzy inference aystem (ANFIS) [32], and the evolutionary support vector machine inference model (ESIM) [33] are used for result comparison. For the BPNN, it is needed to specify the number of hidden layers and hidden neurons. For the ANFIS, the type of membership function and the number of membership functions for each input are required for constructing the prediction model. The determination of these parameters is often carried out by
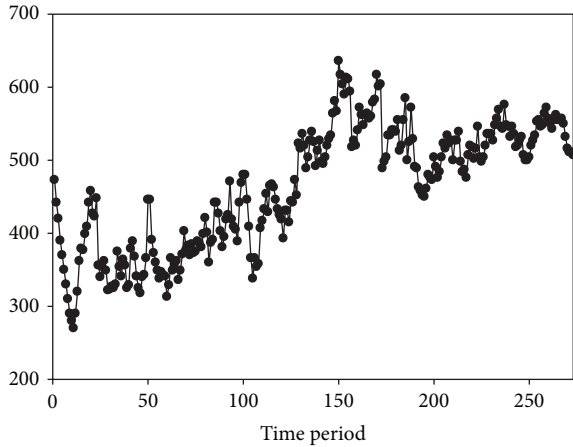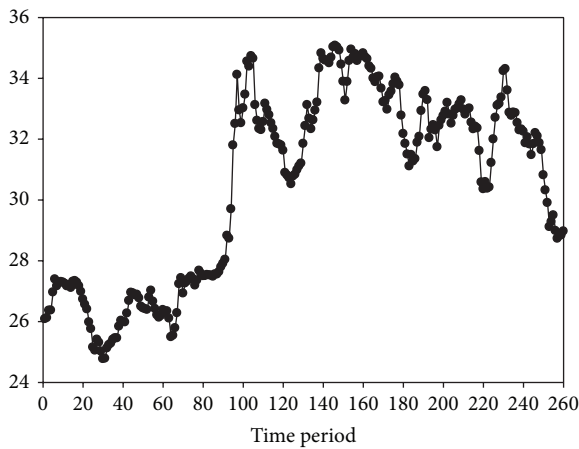
Figure 3: Water flow time series.

Figure 4: USD/TWD exchange rate time series.

Table 2: Optimal tuning parameters of the FLSVR$_{TSP}$.

Parameter	Notation	Mackey-Glass	Water flow	Exchange rate
Embedding dimension	m	4	3	4
Regularization parameter	γ	11.6	632.53	509.1
Kernel function parameter	δ	0.71	5.47	1.29

has better generalization property and it has successfully diminished the problem of over-fitting.

In prediction of Mackey Glass series, RMSE, and MAE of the FLSVR$_{TSP}$ for testing data are 0.005 and 0.004, respectively. The ANFIS shows relatively good forecasting result while performance of the ESIM is poor. In the case of the water flow series, the FLSVR$_{TSP}$ and the ESIM outperform the BPNN and the ANFIS. However, the FLSVR$_{TSP}$ prediction is slightly better than that of the ESIM. Herein, RMSE and MAE of the FLSVR$_{TSP}$ for testing data of the water flow series are 10.33 and 8.35.

In the task of forecasting USD/TWD exchange rate series, although the ANFIS model delivers the smallest error in the training data set, its performance on the testing data set is undesirable. Moreover, the FLSVR$_{TSP}$ yields the best outcome since its RMSE and MAE for the testing data are 0.36 and 0.29. The experimental results have shown that a hybridization of the LS-SVR and FA algorithms can deliver more superior performance compared with other benchmark approach. The FA algorithm has autonomously identified the most appropriate values of the LS-SVR's tuning parameters as well as the embedding dimension. This eliminates the tedious effort for setting the model parameters and also enhances the model prediction performance.

5. Conclusion

This paper has presented a novel hybrid AI model, named as the FLSVR$_{TSP}$, to assist decision-makers in dealing with time series forecasting. The FLSVR$_{TSP}$ was developed by a fusion of the LS-SVR and FA techniques. The LS-SVR is employed to infer the input/output mapping function of time series data. Meanwhile, the FA searching algorithm is utilized to identify the most appropriate set of tuning parameters. This mechanism eliminates the need of expertise or trial-and-error process in parameter setting. Moreover, simulation and performance comparison, for simulated and real-world time series data, have proven the aptitude of the FLSVR$_{TSP}$. These facts demonstrate the strong potential of the proposed model as an alternative for time series forecasting. The future direction of the current work may include improving the current model for solving multistep-ahead time series prediction and applying the hybrid intelligent model to forecast other real-world time series.

repetitive trial-and-error tuning processes. In this study, for each time series, we select the model configuration that yields the smallest prediction error of validating data.

It is noticed that the embedding dimensions (m) for the BPNN, ANFIS, and ESIM models are calculated by the FNN approach [26]. Using this approach, the embedding dimensions for the Mackey-Glass series, the water flow series, and the USD/TWD exchange rate series are 3, 3, and 4, respectively. Meanwhile, in the FLSVR$_{TSP}$, the FA automatically identifies the optimal embedding dimensions. The optimal tuning parameters of the FLSVR$_{TSP}$ for these three time series are shown in the Table 2.

For performance comparison, root mean square error (RMSE) and mean absolute error (MAE) for training and testing data sets are calculated. The forecasting results obtained from the BPNN, ANFIS, ESIM approaches, and the proposed FLSVR$_{TSP}$ are provided in Table 3. It is observable that the FLSVR$_{TSP}$ has achieved a significant improvement in terms of prediction accuracy. The prediction errors of testing data sets yielded by the newly developed model are smaller than that obtained by other AI approaches. This means that FLSVR$_{TSP}$

TABLE 3: Result comparison.

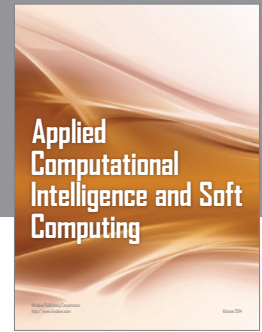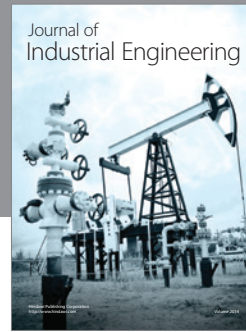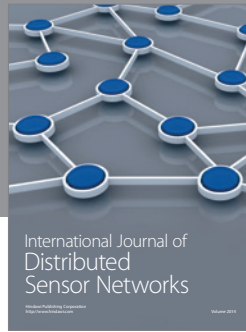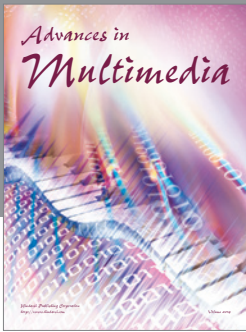| Time series	Result comparison		BPNN	ANFIS	ESIM	FLSVR$_{TSP}$
Mackey-Glass	Training	RMSE	0.029	0.013	0.073	0.005
		MAE	0.024	0.008	0.059	0.003
	Testing	RMSE	0.027	0.011	0.069	**0.005**
		MAE	0.023	0.007	0.057	**0.004**
Water flow	Training	RMSE	26.66	20.66	25.34	25.15
		MAE	21.18	15.69	19.31	19.08
	Testing	RMSE	15.33	14.13	10.71	**10.33**
		MAE	12.55	11.30	8.88	**8.35**
Exchange rate	Training	RMSE	0.42	0.20	0.44	0.39
		MAE	0.31	0.13	0.32	0.26
	Testing	RMSE	0.60	0.67	0.42	**0.36**
		MAE	0.47	0.52	0.33	**0.29**

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

[1] A. Cherif, H. Cardot, and R. Boné, "SOM time series clustering and prediction with recurrent neural networks," *Neurocomputing*, vol. 74, no. 11, pp. 1936–1944, 2011.

[2] J. Zhou, J. Shi, and G. Li, "Fine tuning support vector machines for short-term wind speed forecasting," *Energy Conversion and Management*, vol. 52, no. 4, pp. 1990–1998, 2011.

[3] H. Yoon, S.-C. Jun, Y. Hyun, G.-O. Bae, and K.-K. Lee, "A comparative study of artificial neural networks and support vector machines for predicting groundwater levels in a coastal aquifer," *Journal of Hydrology*, vol. 396, no. 1-2, pp. 128–138, 2011.

[4] L. Barba, N. Rodriguez, and C. Montt, "Smoothing strategies combined with ARIMA and neural networks to improve the forecasting of traffic accidents," *The Scientific World Journal*, vol. 2014, Article ID 152375, 12 pages, 2014.

[5] M. Y. Cheng and A. F. V. Roy, "Evolutionary fuzzy decision model for cash flow prediction using time-dependent support vector machines," *International Journal of Project Management*, vol. 29, no. 1, pp. 56–65, 2011.

[6] S. Ismail, A. Shabri, and R. Samsudin, "A hybrid model of self-organizing maps (SOM) and least square support vector machine (LSSVM) for time-series forecasting," *Expert Systems with Applications*, vol. 38, no. 8, pp. 10574–10578, 2011.

[7] W. Yan, H. Qiu, and Y. Xue, "Gaussian process for long-term time-series forecasting," in *Proceedings of the International Joint Conference on Neural Networks (IJCNN '09)*, pp. 3420–3427, June 2009.

[8] K. Lukoseviciute and M. Ragulskis, "Evolutionary algorithms for the selection of time lags for time series forecasting by fuzzy inference systems," *Neurocomputing*, vol. 73, no. 10–12, pp. 2077–2088, 2010.

[9] M. Zou and H. Tao, "Short term load forecasting with least square support vector regression and PSO," in *Applied Informatics and Communication*, J. Zhang, Ed., vol. 228, pp. 124–132, Springer, Berlin, Germany, 2011.

[10] P. Melin, J. Soto, O. Castillo, and J. Soria, "A new approach for time series prediction using ensembles of ANFIS models,"

Expert Systems with Applications, vol. 39, no. 3, pp. 3494–3506, 2012.

[11] J. Mu, F. Wu, and A. Zhang, "Housing value forecasting based on machine learning methods," *Abstract and Applied Analysis*, vol. 2014, Article ID 648047, 7 pages, 2014.

[12] K. de Brabanter, P. Karsmakers, F. Ojeda et al., "LS-SVMlab toolbox user's guide version 1.8," Internal Report 10-146, ESAT-SISTA, Katholieke Universiteit Leuven, Leuven, Belgium, 2010.

[13] J. Suykens, J. V. Gestel, J. D. Brabanter, B. D. Moor, and J. Vandewalle, *Least Square Support Vector Machines*, World Scientific Publishers, Singapore, 2002.

[14] L. Yu, H. Chen, S. Wang, and K. K. Lai, "Evolving least squares support vector machines for stock market trend mining," *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 1, pp. 87–102, 2009.

[15] J. A. K. Suykens and J. Vandewalle, "Least squares support vector machine classifiers," *Neural Processing Letters*, vol. 9, no. 3, pp. 293–300, 1999.

[16] H.-H. Tran and N.-D. Hoang, "An artificial intelligence approach for groutability estimation based on autotuning support vector machine," *Journal of Construction Engineering*, vol. 2014, Article ID 109184, 9 pages, 2014.

[17] M.-Y. Cheng, N.-D. Hoang, A. F. V. Roy, and Y.-W. Wu, "A novel time-depended evolutionary fuzzy SVM inference model for estimating construction project at completion," *Engineering Applications of Artificial Intelligence*, vol. 25, no. 4, pp. 744–752, 2012.

[18] M. Cheng and N. Hoang, "Groutability estimation of grouting processes with microfine cements using an evolutionary instance-based learning approach," *Journal of Computing in Civil Engineering*, vol. 28, no. 4, Article ID 04014014, 2014.

[19] E. Bonabeau, M. Dorigo, and G. Theraulaz, *Swarm Intelligence: From Natural to Artificial Systems*, Oxford University Press, New York, NY, USA, 1999.

[20] C. Grosan, A. Abraham, and M. Chis, "Swarm intelligence in data mining," *Studies in Computational Intelligence*, vol. 34, pp. 1–20, 2006.

[21] X.-S. Yang, *Firefly Algorithm*, Luniver Press, Bristol, UK, 2008.

[22] S. K. Pal, C. S. Rai, and A. P. Singh, "Comparative study of firefly algorithm and particle swarm optimization for noisy non-linear optimization problems," *International Journal of Intelligent Systems and Applications*, vol. 4, pp. 50–57, 2012.

[23] X.-S. Yang, *Nature-Inspired Optimization Algorithms*, Elsevier, 2014.

[24] G. P. Zhang, "Time series forecasting using a hybrid ARIMA and neural network model," *Neurocomputing*, vol. 50, pp. 159–175, 2003.

[25] Y. Bao, Y. Yang, T. Xiong, and J. Zhang, "A comparative study of multi-step-ahead prediction for crude oil price with support vector regression," in *Proceedings of the 4th International Joint Conference on Computational Sciences and Optimization (CSO '11)*, pp. 598–602, April 2011.

[26] M. B. Kennel, R. Brown, and H. D. I. Abarbanel, "Determining embedding dimension for phase-space reconstruction using a geometrical construction," *Physical Review A*, vol. 45, no. 6, pp. 3403–3411, 1992.

[27] G.-S. Su, "A new intelligent model for nonlinear time series prediction," in *Proceedings of the International Conference on Computer Science and Software Engineering (CSSE '08)*, vol. 1, pp. 435–438, Wuhan, China, December 2008.

[28] M.-Y. Cheng and N.-D. Hoang, "Risk score inference for bridge maintenance project using evolutionary fuzzy least squares support vector machine," *Journal of Computing in Civil Engineering*, vol. 28, no. 3, Article ID 04014003, 2014.

[29] A. Hashmi, N. Goel, S. Goel, and D. Gupta, "Firefly algorithm for unconstrained optimization," *IOSR Journal of Computer Engineering*, vol. 11, pp. 75–78, 2013.

[30] R. L. Haupt and S. E. Haupt, *Practical Genetic Algorithm*, John Wiley & Sons, 2004.

[31] C. M. Bishop, *Pattern Recognition and Machine Learning*, Springer, Singapore, 2006.

[32] J.-S. R. Jang, C.-T. Sun, and E. Mizutani, *Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence*, Prentice Hall, 1997.

[33] M.-Y. Cheng and Y.-W. Wu, "Evolutionary support vector machine inference system for construction management," *Automation in Construction*, vol. 18, no. 5, pp. 597–604, 2009.