Hindawi
Complexity
Volume 2020, Article ID 6393805, 13 pages
https://doi.org/10.1155/2020/6393805

Research Article

Time Series Prediction Based on Complex-Valued S-System Model

Bin Yang,[1] **Wenzheng Bao**,[2] **and Yuehui Chen**[3]

[1]*School of Information Science and Engineering, Zaozhuang University, Zaozhuang 277160, China*
[2]*School of Information and Electrical Engineering, Xuzhou University of Technology, Xuzhou 221018, China*
[3]*School of Information Science and Engineering, University of Jinan, Jinan 250022, China*

Correspondence should be addressed to Wenzheng Bao; baowz55555@126.com

Received 13 December 2019; Revised 17 April 2020; Accepted 11 May 2020; Published 28 May 2020

Academic Editor: Danilo Comminiello

Symbolic regression has been utilized to infer mathematical formulas in order to solve the complex prediction and classification problems. In this paper, complex-valued S-system model (CVSS) is proposed to predict real-valued time series data. In a CVSS model, input variables and rate constants are complex-valued. The time series data need to be translated into complex numbers. The hybrid evolutionary algorithm based on complex-valued restricted additive tree and firefly algorithm is proposed to search the optimal CVSS model. Three financial time series data and Mackey–Glass chaos time series are collected to evaluate our proposed method. The experiment results show that the predicted data are very close to the target ones and our method could obtain the better RMSE, MAP, MAPE, POCID, R^2, and ARV performances than ARIMA, radial basis function neural network (RBFNN), flexible neural tree (FNT), ordinary differential equation (ODE), and S-system.

1. Introduction

One of the main purposes of time series analysis is to predict the future data based on the existing historical one. Its idea is to search a model or function, in which the past values are set as inputs and the future values are utilized as outputs [1–4]. Time series exist in almost all fields of natural science and social science, so researches of time series analysis methods are of great significance for prediction, control, and diagnosis of practical issues [5–7]. However, due to the nonlinear characteristics of time series such as noise, irregularity, randomness, and chaos in practical application, the traditional time series prediction methods, such as exponential smoothing method and autoregressive integrated moving average (ARIMA) model, are not effective in solving such problems [8,9]. Thus, time series prediction problem has been considered to be a difficult job.

Because the artificial neural network (ANN) has good abilities of learning, generalization, and error tolerance, many ANN models have been utilized widely to capture nonlinear characteristics for time series prediction problems in the past decades, such as radial basis function neural network (RBFNN) [10], Elman neural network [11], wavelet process neural network [12], recurrent predictor neural network (RPNN) [13], beta basis function interval type-2 fuzzy neural network [14], flexible neural tree (FNT) [15], functional link network [16], and deep neural networks [17,18]. However, most of the ANN models are black boxes, in which the relationships between input variables and output variables are not explained and understood easily. Especially for some practical problems, internal mechanism could not be understood from the models obtained, which may lead to restrict the problems to being solved.

Recently, various symbolic regression (SR) methods are being utilized to solve time series prediction problems. Compared with ANN models, with the given sets of independent variables and functions, SR methods could search the hidden mathematical formulas, computer programs, and logical expressions, which could explain well the internal mechanisms of the practical problems [19–22]. Johari et al. presented a genetic programming (GP) approach to predict the soil-water characteristic curves of soils according to terminal and function sets [23]. Azzawi et al. proposed gene expression programming (GEP) and gene extracted approaches to predict lung cancer from biology data. Mahmoodi et al. inferred mathematical equations based on GEP and computational fluid dynamics data in order to predict the marine propeller data [24]. Zhang et al. proposed an improved multiexpression programming (MEP) to predict 28-day cement compressive strength, which performed better than MEP, neural network, and fuzzy logical [25]. We have utilized an ordinary differential equation (ODE) model to forecast the small-scale traffic data and model stock index, and the results shown that the ODE model was more feasible and efficient for predicting time series data [26]. Zhang et al. utilized an S-system model for Shanghai stock exchange composite index prediction and the experiment results revealed that the S-system has the better performance than an ODE model [27].

In order to improve the prediction accuracy of time series and explain the internal mechanisms of the practical problems future, a novel SR method based on complex-valued S-system (CVSS) and complex-valued hybrid evolutionary algorithm are proposed in this paper. A complex number could broaden the dimension of solution, which improves the abilities of modeling and generalization, so complex-valued methods have shown great potential for forecasting time series data [28–35]. Thus, this paper proposes the complex-valued version of S-system (CVSS) to solve the time series prediction problem. CVSS could contain a complex-valued structure and coefficients, which can enhance the prediction ability of the S-system. The representation of a complex-valued restricted additive tree (CVRAT) is very close to the form of a CVSS model, so CVRAT is utilized to search the optimal structure of a CVSS model. In order to improve the optimization ability of firefly algorithm (FA), complex-valued firefly algorithm (CVFA) is presented to optimize the complex-valued and real-valued parameters of CVSS with faster convergence and ability and more population diversity. The novelty in our approach is to predict real time series data by a novel complex-valued model and evolutionary algorithms.

Three real time series datasets collected from the Shanghai stock exchange composite index, NASDAQ index and exchange rate of Hong Kong dollar to renminbi, and Mackey–Glass chaos time series are utilized to test the performance of our method. The experiment results prove that our proposed method is superior to classical statistical method, neural network, and real-valued differential equation models, and our method could also obtain the clear and well-understood mathematical formulas.

2. Methods

2.1. Complex-Valued S-System Model. The complex-valued S-system (CVSS) model is the complex-valued version of S-system, whose coefficients and functions are complex-valued. Suppose that the time series contain N variables. The $i-$th complex-valued ordinary differential equation model in the CVSS model is described as follows:

$$X_i'(t) = \alpha_i \prod_{j=1}^{N} X_j^{g_{ij}}(t) - \beta_i \prod_{j=1}^{N} X_j^{h_{ij}}(t), \tag{1}$$

where X_i and X_j are the $i-$th and $j-$th complex-valued input variables, respectively; α_i and β_i are complex-valued rate constants of the $i-$th variables; and h_{ij} and g_{ij} are real-valued kinetic orders.

2.2. Structure Optimization. A real-valued restricted additive tree model (RVRAT) was proposed to optimize the structure of a real-valued S-system model [36,37]. In order to search the optimal CVSS model, the complex-valued restricted additive tree (CVRAT) is proposed. In CVRAT algorithm, the root node is set to subtraction (−). Other nodes are created by variable (V) and function (F) sets, which are described as follows:

$$\begin{cases} F = \{\times_2, \times_3, \ldots, \times_n\}, \\ V = \{z_1, z_2, \ldots, z_n, C\}, \end{cases} \tag{2}$$

where $\times_n$ is the product of n complex-valued input variables. An example of the complex-valued restricted additive tree model is depicted in Figure 1. In order to represent the parameters of the CVSS model, a real-valued parameter (h_{ij} or g_{ij}) is assigned to each variable node and a complex-valued parameter (α_i or β_i) is assigned to each branch of the root node. The corresponding CVSS model is $\mathrm{d}z_i/\mathrm{d}t = \alpha_i z_1^{g_{i1}} z_2^{g_{i2}} z_3^{g_{i3}} - \beta_i z_1^{h_{i1}} z_2^{h_{i2}} z_3^{h_{i3}} z_4^{h_{i4}}$.

In order to search the optimal CVSS model, genetic operators, such as selection, crossover, and mutation, are utilized. The selection operation is consistent with standard genetic programming. The greater the fitness values of individuals, the greater the probability that the individuals are selected to the next generation. In the crossover operation, two CVSS models are randomly selected, and two subtrees are selected to be crossed, which are shown in Figure 2. In the mutation operation, a mutation node is randomly selected. Another node is created randomly and utilized to replace the selected node, which are shown in Figure 3.

2.3. Parameter Optimization. Complex-valued firefly algorithm (CFA) is the complex-valued variant of firefly algorithm, in which each firefly represents a complex-valued solution containing two parts: real part and imaginary part [38]. The multiplication of individual dimension could increase the information capacity and diversity of population. In this paper, CFA is utilized to evolve the parameters of CVSS.

In CFA, complex-valued firefly can find its collaborators and move to the place of the better firefly according to brightness property. The fireflies with low brightness are attracted by the fireflies with high brightness. For each individual, the attraction and luminance of other individuals vary according to distance [39]. The flowchart of parameters optimization of CVSS with CFA is given in detailed as followed:

(1) In the CVSS model, rate constants (α_i and β_i) are complex-valued and kinetic orders (h_{ij} and g_{ij}) are real-valued. Thus, the parameter vector of CVSS contains complex-valued and real-valued parameters. According to the given CVSS model, add up the number of rate constants R and kinetic orders K.

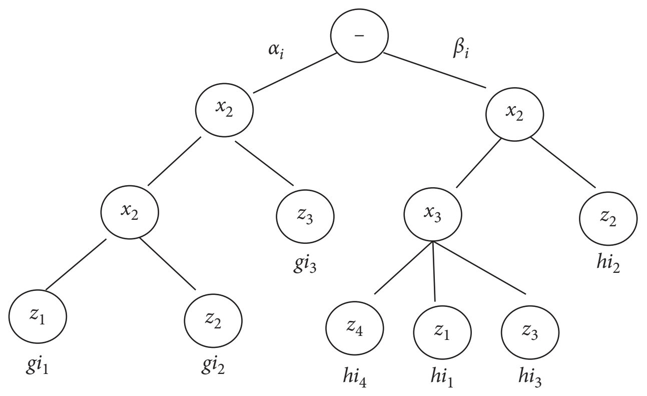


Figure 1: An example of the complex-valued restricted additive tree model.

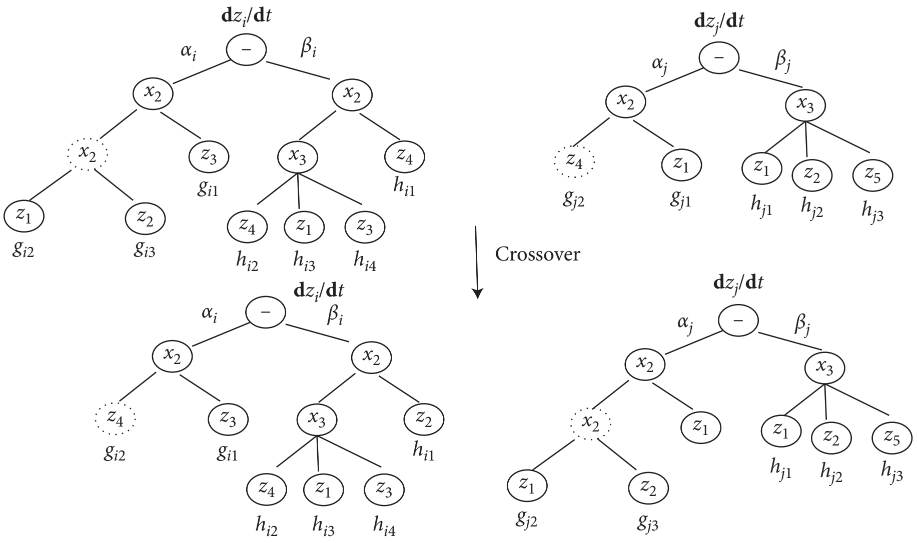


Figure 2: The crossover operator.

Initialize the m complex-valued firefly individuals $[Z_1, Z_2, \ldots, Z_m]$, where $Z_i = [Z_i^1, Z_i^2, \ldots, Z_i^{R+K}]$.

(2) The complex-valued distance matrix $r_{m\times m}$ of m fireflies is calculated.

(3) With fitness function $f(\cdot)$, the fitness values of population $[f_1, f_2, \ldots, f_m]$ are calculated. The firefly of CFA is complex-valued, so complex-valued kinetic orders need to be converted into real-valued ones. Complex-valued kinetic order Z_k is converted as follows [40]:

$$\rho_k = \sqrt{Z_{k,R}^2 + Z_{k,I}^2},$$
$$X_k = \rho_k \times \operatorname{sgn}\left(\sin\left(\frac{Z_{k,I}}{\rho_k}\right)\right) + \frac{a_k + b_k}{2}, \tag{3}$$

where ρ_k is the modulus and a_k and b_k are the minimum and maximum values, respectively.

If the optimal solution is obtained, CFA is stopped. The real and imaginary parts of brightness and attractiveness of the i − th firefly are calculated.

$$B_{i,R} = B_{i,0} \times e^{-\gamma\times r_{ij,R}},$$
$$B_{i,I} = B_{i,0} \times e^{-\gamma\times r_{ij,I}}, \tag{4}$$

where B_{i0} denotes the maximum brightness of the i − th firefly which is equal to the fitness value f_i of the i − th firefly; γ is a coefficient of light absorption; $B_{i,R}$ and $B_{i,I}$ are the real part and imaginary part of the brightness of firefly i, respectively; and $r_{ij,R}$ and $r_{ij,I}$ are the real part and imaginary part of the distance between firefly i and firefly j, respectively.

(4) For each firefly, search the most attractive firefly around it and update its position containing real and imaginary parts.

$$Z_{i,R}(t+1) = Z_{i,R}(t) + B_{i,R}\left(Z_{j,R}(t) - Z_{i,R}(t)\right) + \alpha\times\varepsilon_{i,R},$$
$$Z_{i,I}(t+1) = Z_{i,I}(t) + B_{i,I}\left(Z_{j,I}(t) - Z_{i,I}(t)\right) + \alpha\times\varepsilon_{i,I}, \tag{5}$$

where $Z_{i,R}(t+1)$ and $Z_{i,I}(t+1)$ are the real part and imaginary part of firefly i at the $t+1$ − th time point,

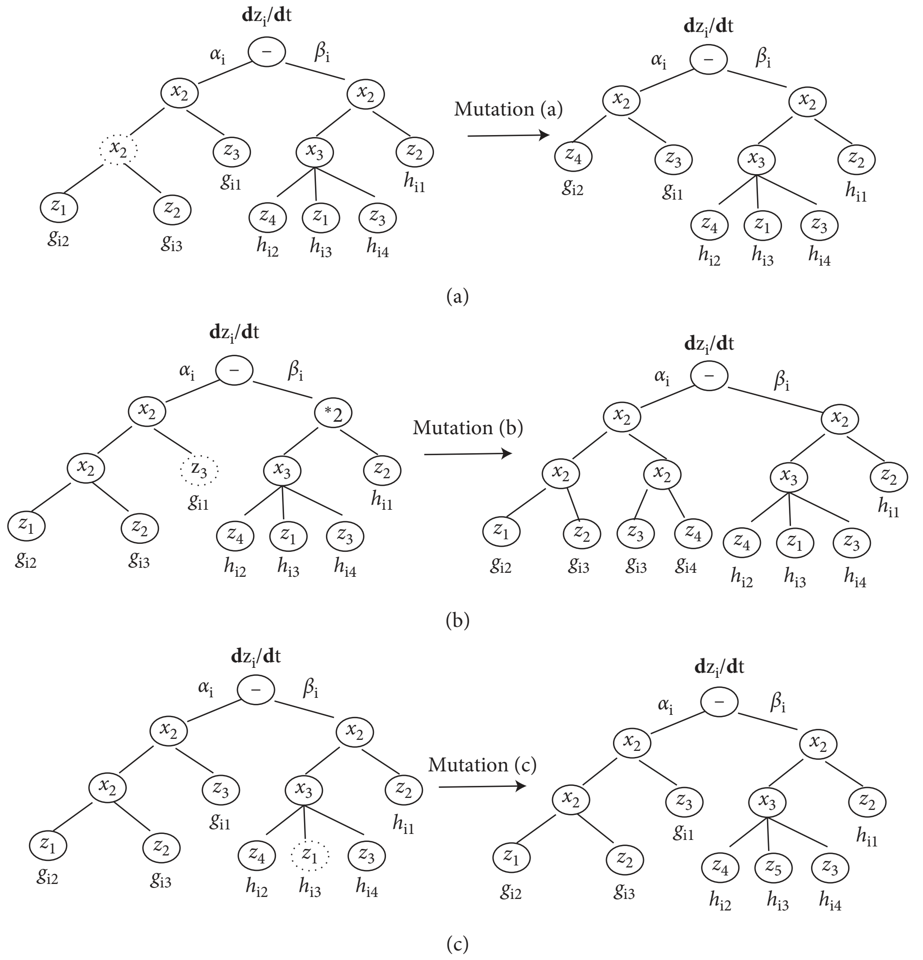


Figure 3: The mutation operators.

respectively; $Z_{i,R}(t)$ and $Z_{i,I}(t)$ are the real part and imaginary part of firefly i at the $t-$th time point, respectively; $\varepsilon_{i,R}$ and $\varepsilon_{i,I}$ are the real part and imaginary part of complex-valued Gaussian random number, respectively; and α is a step size randomly created from the interval [0,1]. Then, Go to step (2).

2.4. Time Series Prediction Based on Complex-Valued S-System Model. The complex-valued S-system model is utilized to solve the time series prediction problem. In the prediction process, real-valued time series are firstly translated into complex-valued data with Algorithm 1. Complex-valued time series data are utilized to search the optimal CVSS model, which predicts the complex-valued outputs. Finally, complex-valued outputs are translated into real-valued data in order to evaluate the prediction performance of the model with Algorithm 2. Suppose that time series data $[D_1, D_2, \ldots, D_m]$ include m variables and each variable includes n time points ($D_i = [D_i^1, D_i^2, \ldots, D_i^n]$). With complex-valued time series data, our proposed hybrid evolutionary algorithm is utilized to evolve the complex-valued structure and parameters of the CVSS model. The overall optimization process is described as follows:

(1) Define the parameters in the algorithm and initialize N CVSS models.

(2) The fitness value of each CVSS model could be calculated according to Algorithm 3.

(3) RVRAT is applied to search the optimal structure of the CVSS model, which is introduced in Section 2.2. At some iterations, CFA is utilized to search the optimal complex-valued parameters of the CVSS model, whose structure is fixed.

(4) If the satisfied condition is achieved, the optimal process stops; otherwise, go to step 2.

```
**Input**: real-valued input vector $[g_1, g_2, \ldots, g_m]$ ($m$ is the number of sample points)
**Output**: complex-valued output vector $[z_1, z_2, \ldots, z_m]$
Count the maximum and minimum values of the input vector ($g_{\max}$ and $g_{\min}$);
**For** $k = 1$; $k \leq m$; $k$++ **do**
    $\varphi_k = (g_k - g_{\min}/g_{\max} - g_{\min})(2\pi - \delta)$; // $\delta$ is a shift angle
    $z_k = e^{i\varphi_k}$
**end**
```

Algorithm 1: Pseudocode of a real number converted into a complex number.

```
**Input**: complex-valued input vector $[z_1, z_2, \ldots, z_m]$ and maximum and minimum values of input data $g_{\max}$ and $g_{\min}$
**Output**: real-valued output data $[y_1, y_2, \ldots, y_m]$
**For** $k = 1$; $k \leq m$; $k$++ **do**
        $\arg z_k = \varphi_k$;
        $y_i = ((\varphi_k \times (g_{\max} - g_{\min}))/(2\pi - \delta_k)) + g_{\min}$;
**end**
```

Algorithm 2: Pseudocode of a complex number converted into a real number.

```
**Input**: real-valued output vector $[o_1, o_2, \ldots, o_m]$, complex-valued input vector $Z = [z_1, z_2, \ldots, z_m]$, step size $h$, total time point $m$,
CVSS model $\mathrm{d}Z/\mathrm{d}t = H(t, Z)$, and fitness function $F(\cdot)$
**Output**: fitness $f_k$ of the $k$ − th CVSS model
**For** $t = 1$; $t \leq m$; $t$++ **do**
    $k_1 = H(t, z_t)$;
    $k_2 = H(t + (h/2), z_t + h \times (k_1/2))$;
    $k_3 = H(t + (h/2), z_t + h \times (k_2/2))$;
    $k_4 = H(t + h, z_t + h \times k_3)$;
    $zo = z_t + h \times ((k_1 + 2k_2 + 2k_3 + k_4)/6)$;
    $ro_t \longleftarrow$ convert $zo$ into real value with Algorithm 2;
**end**
$f_k = F(r_o, Z)$;
```

Algorithm 3: Pseudocode of calculating the fitness of the k − th CVSS model.

3. Experiments

3.1. Data and Evaluation Standard. Three real time series datasets are collected daily to test the performance of the complex-valued S-system model, which include the Shanghai stock exchange composite index (SSEI), NASDAQ index (NASI) and exchange rate of Hong Kong dollar to renminbi (RHKRMB). The data of the past seven days $(x_1, x_2, \ldots, x_7)$ are utilized to predict the current value (y). Mackey–Glass chaos time series are also utilized to evaluate our method. RMSE (root mean square error), MAP (mean absolute percentage), MAPE (mean absolute percentage error), R^2 (coefficient of multiple determinations for multiple regressions), ARV (average relative variance), and POCID (prediction of change in direction) are utilized to evaluate the performance of time series prediction models [16]. Suppose that time series dataset D contains m time points. The six criterions are defined as follows:

$$\text{RMSE} = \sqrt{\frac{1}{m}\sum_{k=1}^{m}\left(y_k^t - y_k^f\right)^2},$$

$$\text{MAP} = \max\left(\frac{\left|y_k^t - y_k^f\right|}{y_k^f}\times 100\right),$$

$$\text{MAPE} = \frac{1}{m}\sum_{k=1}^{m}\left(\frac{\left|y_k^f - y_k^f\right|}{y_k^f}\right)\times 100,$$

$$R^2 = 1 - \frac{\sum_{k=1}^{m}\left(y_k^t - y_k^f\right)^2}{\sum_{k=1}^{m}\left(y_k^t - \overline{y}\right)^2}, \tag{6}$$

$$\text{ARV} = \frac{\sum_{k=1}^{m}\left(y_k^t - y_k^f\right)^2}{\sum_{k=1}^{m}\left(y_k^f - \overline{f}\right)^2},$$

$$\text{POCID} = \frac{\sum_{k=1}^{m}T_k}{m}\times 100,$$

$$T_k = \begin{cases} 1, & \text{if}\left(y_k^t - y_{k-1}^t\right)\times\left(y_k^f - y_{k-1}^f\right) > 0,\\ 0, & \text{otherwise,}\end{cases}$$

where y_k^t is the target value at $k-\text{th}$ time point and y_k^f is the forecasting value at $k-\text{th}$ time point. $\overline{f}$ is the mean of dataset D.

In our experiment, ARIMA model [41], radial basis function neural network (RBFNN) [42], FNT, ordinary differential equation (ODE) [26], and S-system model [27] are also utilized to predict three real time series datasets and Mackey–Glass chaos time series. ARIMA is the most common statistic model used for time series prediction. Real-valued firefly algorithm is utilized to optimize the parameters of RBFNN, FNT, ODE, and S-system, which are the same as the CVSS model. The parameters of the experiment are set consistently, which are chosen based on the experience and references [36–38], and listed in Table 1.

3.2. Shanghai Stock Exchange Composite Index.

3.2. Shanghai Stock Exchange Composite Index. This section collects the Shanghai stock exchange composite index (SSEI) from 1 January 1996 to 29 December 2017. 70% of the data are utilized for training, which contains 3866 time points, and the rest of the data are utilized for testing.

Through several runs, the optimal CVSS model is obtained as follows:

$$\frac{\mathrm{d}y}{\mathrm{d}t} = (3.101448 - 7.744888\,i)x_7^{6.284683}x_3^{-6.021700} - (-1.540345 - 8.878057\,i)x_3^{0.898278}. \tag{7}$$

From equation (7), it could be seen that our method selects two important features (x_3 and x_7), which reveal that the data of the past third and seven days may play the most important role in predicting the Shanghai stock exchange composite index.

Table 1: Configuration of algorithms in our experiment.

Parameters	Values
Population size in CVRAT	30
Maximum iteration in CVRAT	200
p_m in CVRAT	0.3
p_c in CVRAT	0.7
α in CFA	0.5
Number of fireflies in CFA	50
Maximum iteration in CFA	100

The predicted results of 6 methods are depicted in Figure 4, which contains the predicted data and predicted errors. From the results, it could be seen that the predicted data of CVSS, ODE, and S-system are very close to the target ones and the predicted errors are very small and close to zero. The prediction errors of FNT, RBFNN, and ARIMA are relatively large.

The predicted performances of 6 methods are listed in Table 2. From Table 2, we can see that differential equation models (ODE, S-system, and CVSS) have smaller RMSE, MAP, ARV, and MAPE and higher R^2 and POCID than the neural network model (FNT and RBFNN) and classical time series prediction method (ARIMA). Among three kinds of differential equation models, the CVSS model has better performance than real-valued differential equation models except for POCID. In terms of POCID, S-system performs best, but there is little difference among these three methods, which reveals that our method could predict directional tendency of data fluctuation accurately.

3.3. NASDAQ Index. This section collects the NASDAQ index (NASI) from 2 January 1990 to 29 December 2017. 70% of the data are utilized for training, which contains 4936 time points, and the rest of the data are utilized for testing.

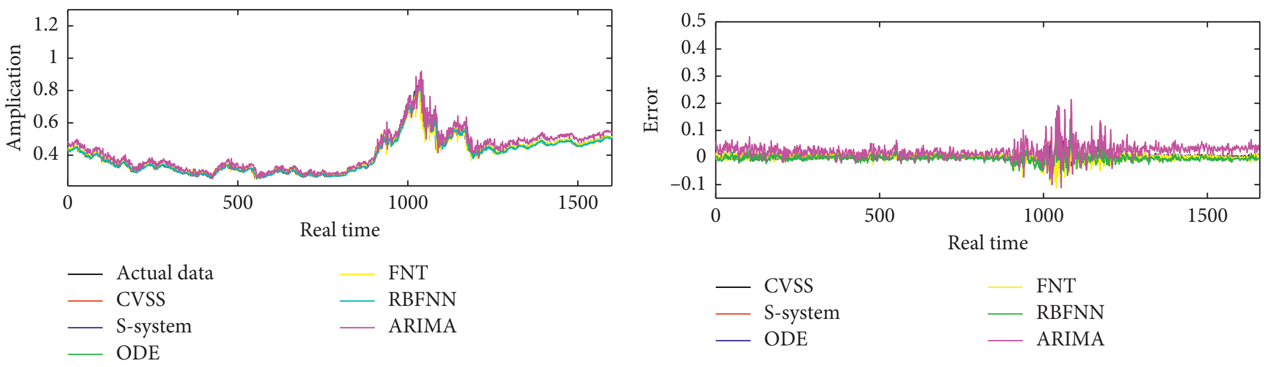
Through several runs, the optimal CVSS model is obtained as follows:

$$\frac{\mathrm{d}y}{\mathrm{d}t} = (-0.264201 - 2.017404\,i)x_3^{1.285517} - (0.200819 - 2.067524\,i)x_6^{0.346046}x_4^{0.554568}. \tag{8}$$

In equation (8), our method selects three important features (x_3, x_4, and x_6) automatically. The stock indexes of the past third, fourth, and sixth days may play the important role in predicting the current one.

The predicted NASI data of 6 methods are depicted in Figure 5, which also contains the predicted errors. For CVSS, the curves of actual data and predicted data are almost coincident. The predicted errors distribute mainly around zero. From the prediction results, we can see that our proposed method could predict the NASI dataset more accurately.

The predicted performances of 6 methods are listed in Table 3. In terms of RMSE, CVSS is 31.7% smaller than S-system, 33.07% smaller than ODE, 61.65% smaller than



(a)

(b)

Figure 4: The predicted results of 6 methods with the SSEI dataset.

Table 2: Performance of ARIMA, RBFNN, FNT, ODE, S-system, and CVSS with the SSEI dataset.

Methods	RMSE	MAP	ARV	MAPE	R^2	POCID
CVSS	**0.007053**	**9.6815**	**0.005206**	**1.1121**	**0.99501**	90.1
S-system	0.008105	9.957	0.00535	1.1271	0.99465	**90.3**
ODE	0.010321	12.349	0.00981	1.391	0.99128	90.1
FNT	0.014559	18.931	0.018903	2.2848	0.9811	89.7
RBFNN	0.014681	20.06	0.018024	2.1804	0.98198	86.5
ARIMA	0.020766	20.533	0.029814	3.9766	0.97019	68.1

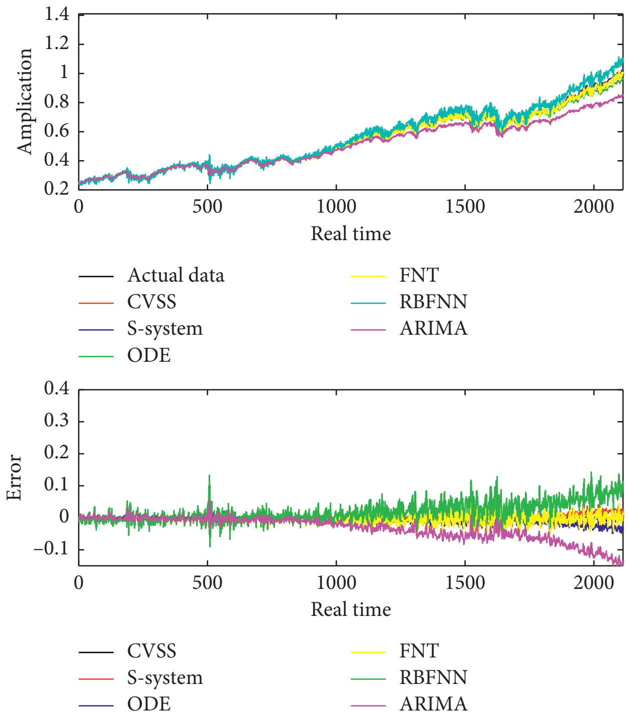


Figure 5: The predicted results of 6 methods with NASI dataset.

FNT, 88.23% smaller than RBFNN, and 91.33% smaller than ARIMA, which reveal that our method has the smallest prediction errors. In terms of MAP, the CVSS model has the smallest performance, which is 7.7941. In terms of ARV, CVSS is 43.77% smaller than S-system, 56.92% smaller than ODE, 89.7% smaller than FNT, 97.274% smaller than RBFNN, and 99.195% smaller than ARIMA, which show that our method has the stronger generalization ability. In terms of MAPE, CVSS is 16.97% smaller than S-system, 42.58% smaller than ODE, 67.11% smaller than FNT, 81.62%

Table 3: Performance of ARIMA, RBFNN, FNT, ODE, S-system, and CVSS with the NASI dataset.

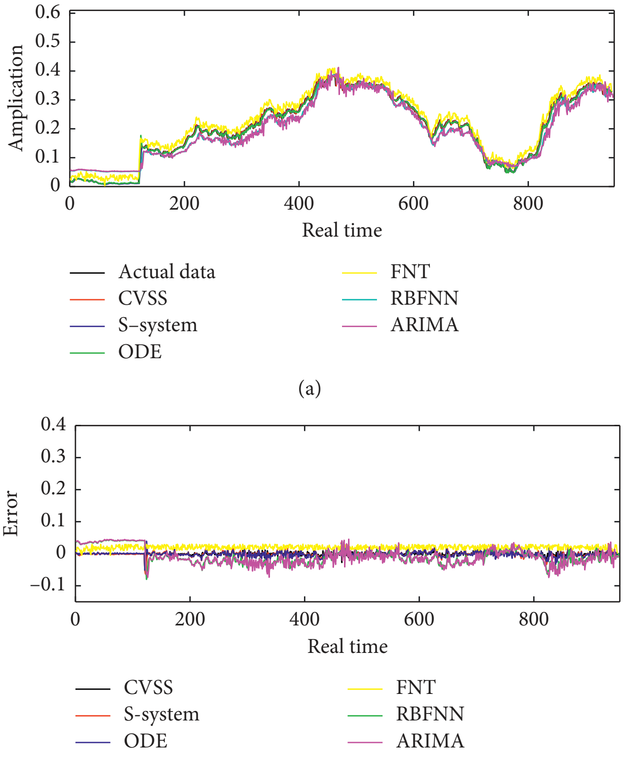
Methods	RMSE	MAP	ARV	MAPE	R^2	POCID
CVSS	**0.00632**	**7.7941**	**0.00075**	**0.83321**	**0.99932**	**88.1**
S-system	0.008324	7.9168	0.001707	1.0035	0.99829	87.2
ODE	0.00841	7.9019	0.001741	1.4512	0.99799	87.7
FNT	0.016468	32.352	0.006859	2.5336	0.99314	87.0
RBFNN	0.03669	37.371	0.027513	4.5327	0.97249	86.7
ARIMA	0.049849	18.46	0.093189	5.31	0.90681	70.0

smaller than RBFNN, and 84.31% smaller than ARIMA. In terms of R^2 and POCID, CVSS has highest performance among 6 time series prediction methods. By analysis, the CVSS model performs best in terms of 6 criterions.

3.4. Exchange Rate of Hong Kong Dollar to Renminbi. Exchange rate refers to the rate of exchange between two currencies, which could regulate a country's import and export trade. In this section, exchange rates of Hong Kong dollar to renminbi (RHKRMB) are collected from 2 January 2006 to 29 December 2018. 70% of the data are utilized for training, which contains 2210 time points, and the rest of the data are utilized for testing. Through several runs, the optimal CVSS model is obtained as follows:

$$\frac{dy}{dt} = (-0.010162 + 0.042573\,i)x_2^{-0.506543}x_1^{0.189222} - (0.01746 + 0.042042)x_1^{0.438178}x_4^{0.501454}. \qquad (9)$$

From equation (9), we could see that our proposed method selects three important features (x_1, x_2, and x_4), which show that the data of the past first, second, and fourth days help to forecast the current exchange rate accurately.



(a)

(b)

Figure 6: The predicted results of 6 methods with the RHKRMB dataset.

The predicted data and errors by 6 models are depicted in Figure 6, which show that the predicted errors by CVSS are very small and the predicted curve is almost the same as the real one. Other models could gain the larger prediction errors. The predicted performances of 6 methods are listed in Table 4. For RMSE, ARV, MAPE, R^2, and POCID, the CVSS model performs best among 6 methods. RBFNN has the lowest MAP. CVSS has worse MAP performance, which reveals that the predicted errors by CVSS may be very large at some time points. In terms of RMSE, CVSS is 6.42% smaller than S-system, 19.59% smaller than ODE, 64.42% smaller than FNT, 73.87% smaller than RBFNN, and 75.88% smaller than ARIMA. In terms of ARV, CVSS is 12.11% smaller than S-system, 43.52% smaller than ODE, 89.61% smaller than FNT, 94.48% smaller than RBFNN, and 95.29% smaller than ARIMA, which show that our method has the stronger generalization ability. In terms of MAPE, CVSS is 5.75% smaller than S-system, 24.54% smaller than ODE, 87.54% smaller than FNT, 93.02% smaller than RBFNN, and 93.05% smaller than ARIMA. In terms of R^2, CVSS is 0.048% higher than S-system, 0.269% higher than ODE, 3.0% higher than FNT, 6.34% higher than RBFNN, and 7.59% higher than ARIMA. In terms of POCID, CVSS is 0.022% higher than S-system, 1.11% higher than ODE, 1.99% higher than FNT, 4.87% higher than RBFNN, and 22.26% higher than ARIMA, which prove that CVSS could predict directional tendency of data fluctuation of exchange rate more accurately.

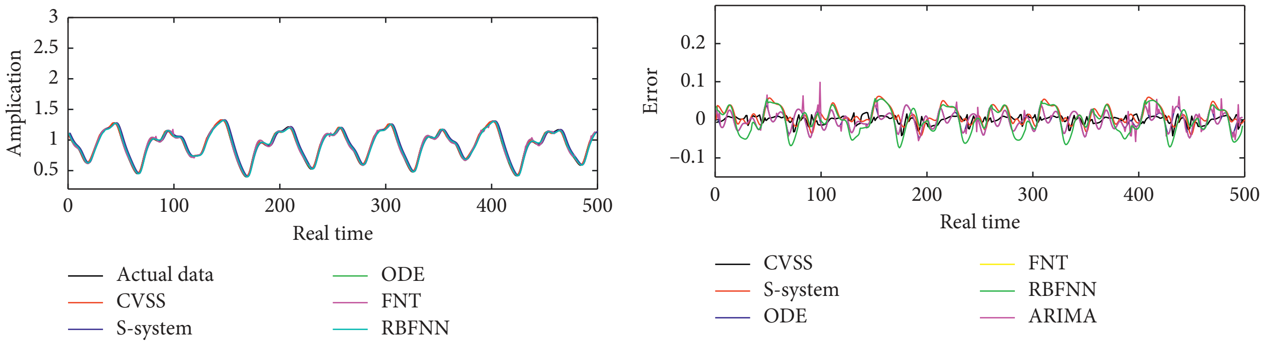
3.5. Mackey–Glass Chaos Time Series. Since Mackey and Glass first discovered the chaotic phenomena in time-delay systems, time-delay chaotic systems have attracted wide attention and are often utilized to test the performances of nonlinear system models. The chaos time series could be generated by the time-delayed differential equation [43,44]:

$$\frac{dx(t)}{dt} = -0.1x(t) + \frac{0.2x(t-17)}{1 + x^{10}(t-17)}. \qquad (10)$$

The variable vector $[x(t-18), x(t-12), x(t-6), x(t)]$ is utilized to forecast variable $x(t+6)$. 500 samples are utilized to search the optimal structures and parameters of the CVSS model and the rest of the 500 samples are utilized as testing data to verify the validity of the CVSS model. The

Table 4: Performance of ARIMA, RBFNN, FNT, ODE, S-system, and CVSS with the RHKRMB dataset.

Methods	RMSE	MAP	ARV	MAPE	R^2	POCID
CVSS	**0.006601**	469.18	**0.003469**	**3.3499**	**0.99653**	90.3
S-system	0.007054	492.67	0.003947	3.5544	0.99605	90.1
ODE	0.008209	511.75	0.006142	4.4394	0.99386	89.3
FNT	0.01855	184.98	0.033387	26.885	0.96661	88.5
RBFNN	0.025259	**85.343**	0.062869	47.991	0.93713	85.9
ARIMA	0.027369	118.26	0.073748	48.231	0.92625	70.2



(a)

(b)

Figure 7: The predicted results of 5 methods with Mackey–Glass chaos time series.

predicted Mackey–Glass chaos time series by 5 methods are depicted in Figure 7, which also displays the predicted errors. The curves of actual data and predicted data obtained by CVSS are almost the same. The predicted errors of CVSS are smaller than S-system, ODE, FNT, and RBFNN. Thus, the CVSS model could predict chaos time series accurately. The predicted performances of RBFNN, FNT, ODE, S-system, and CVSS are listed in Table 5. From Table 5, it could be clearly seen that the CVSS model has the best performances in terms of RMSE, Map, ARV, MAPE, POCID, and R^2.

According to Tables 2–5, we summarize the performances of 6 models in three aspects: fitting the data, generalization ability, and prediction effect. 6 models are divided into three levels: strong, medium, and week. The analysis results are listed in Table 6.

4. Discussion

4.1. Statistical Significance Test. In order to measure the differences of six forecasting models, the Friedman test is utilized. The null hypothesis of the Friedman test is that all the prediction models are equivalent, so these models have the same ranks. In order to overturn this null hypothesis, Friedman statistic χ_F^2 and F-distribution F_F are defined as follows [16]:

$$\chi_F^2 = \left[\frac{12}{N \times k \times (k+1)}\right] \times \sum_{i=1}^{k} R_i^2 - 3 \times N \times (k+1),$$
$$F_F = \frac{(N-1) \times \chi_F^2}{N \times (k-1) - \chi_F^2}, \tag{11}$$

where N is the number of the criterions; k is the number of the prediction models; and R_i is the sum of the averaged ranks of N criterions for the $i-$ th model. If χ_F^2 is larger than F_F, the null hypothesis could be rejected.

According to the criterions of six models for four time series prediction problems, six models are sorted and the averaged ranks are given, which are listed in Table 7. k and N are set as 6. According to Table 7, Friedman statistic χ_F^2 and F-distribution F_F are calculated as 22.67 and 15.46, respectively. χ_F^2 is larger than F_F, so the null hypothesis is rejected, which reveals that there is the significant difference among six prediction models.

In order to find out the significant difference between a pair of models, the Nemenyi post-hoc test is utilized. The null hypothesis of the Nemenyi test is that there is no significant difference between a pair of models. If there is no significant difference, the null hypothesis will be rejected. In the Nemenyi test, critical difference (CD) is calculated to judge whether the significant difference exits, which is defined as follows:

$$\mathrm{CD} = q_\alpha \sqrt{\frac{k \times (k+1)}{6N}}, \tag{12}$$

where q_α is a critical value. When α is selected as 0.01, q_α is found to be 0.4643. Then, the CD could be calculated as 0.502.

If the difference of the average rank values of two models exceeds the CD value, the null hypothesis is rejected with corresponding confidence. According to Table 7, the results of the Nemenyi test by the pairwise comparisons of six models are depicted in Figure 8. If the difference of the average rank values of two models is smaller than CD (0.502), the responding place is set as white box, which

Table 5: Performance of RBFNN, FNT, ODE, S-system, and CVSS with Mackey–Glass chaos time series.

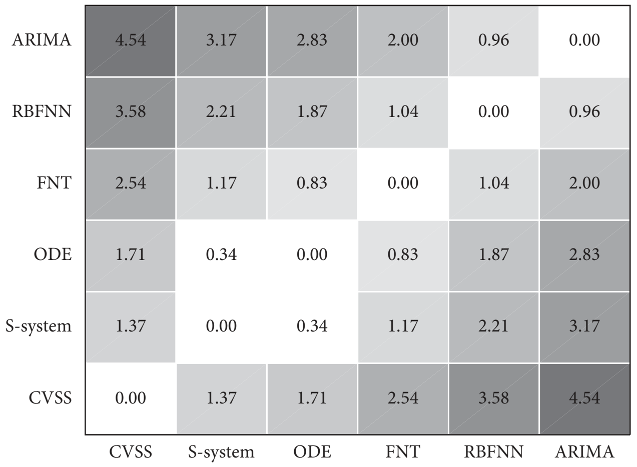
Methods	RMSE	MAP	ARV	MAPE	R^2	POCID
CVSS	**0.009012**	**5.4256**	**0.0019403**	**0.89804**	**0.99806**	**96.177**
S-system	0.024573	6.9489	0.011983	2.1923	0.98802	94.188
ODE	0.019789	7.4946	0.0074549	1.9057	0.99255	92.958
FNT	0.02445	10.187	0.009756	2.1314	0.99024	90.342
RBFNN	0.032639	13.306	0.020886	3.3305	0.97911	93.561

Table 6: Performance comparison of the 6 prediction models.

Methods	Fitting the data	Generalization ability	Prediction effect
CVSS	Strong	Strong	Strong
S-system	Strong	Medium	Medium
ODE	Medium	Medium	Medium
FNT	Medium	Medium	Weak
RBFNN	Weak	Weak	Medium
ARIMA	Weak	Weak	Weak

Table 7: Reranked values of six models for six criterions in four time series prediction problems.

Models	RMSE	MAP	ARV	MAPE	R^2	POCID	Avg. rank	Reranking
CVSS	1	1.75	1.25	1	1	1.25	1.21	1
S-system	2.5	3	2.75	2.5	2.5	2.25	2.58	2
ODE	2.75	3.5	3	2.75	2.75	2.75	2.92	3
FNT	3.75	4	3	3.75	3.75	4.25	3.75	4
RBFNN	5	4.25	5	5	5	4.5	4.79	5
ARIMA	6	4.5	6	6	6	6	5.75	6



ARIMA	4.54	3.17	2.83	2.00	0.96	0.00
RBFNN	3.58	2.21	1.87	1.04	0.00	0.96
FNT	2.54	1.17	0.83	0.00	1.04	2.00
ODE	1.71	0.34	0.00	0.83	1.87	2.83
S-system	1.37	0.00	0.34	1.17	2.21	3.17
CVSS	0.00	1.37	1.71	2.54	3.58	4.54
	CVSS	S-system	ODE	FNT	RBFNN	ARIMA

Figure 8: The results of the Nemenyi test by the pairwise comparisons of six models.

reveals that two models have no significant difference. The gray value of the box indicates the difference between two models, which is also shown in Figure 8. From Figure 8, it could be seen that there is no significant difference between ODE and S-system, while other pairs of models satisfy the Nemenyi test for significant difference.

4.2. Effect of the Number of Training Sets. In order to test the effect of increasing the training for our proposed method, two datasets (SSEI and NASDAQ index) are chosen. The numbers of training sets are set to 500, 1000, 1500, and 2000, respectively, while the number of testing sets is set as 500. The RMSE performances of our method with different training sets for SSEI and NASDAQ prediction problems are listed in Table 8. From Table 8, we could see that with the increase of training sets our method could gain the smaller and more stable prediction RMSE values. When the number of training sets reaches a certain bound, the increase of the training set could improve the prediction ability of our model slightly.

Table 8: The RMSE performances of our method with different training sets for SSEI and NASDAQ prediction problems.

Datasets		500	1000	1500	2000
SSEI	Average	0.0069246	0.006561	0.006493	0.006469
	Best	0.006786	0.006445	0.006418	0.006406
	Worse	0.007541	0.006863	0.006568	0.006505
	Standard deviation	0.000175	0.000106	$3.76E-05$	$3.24E-05$
NASDAQ index	Average	0.006493	0.006483	0.006098	0.006087
	Best	0.006202	0.006201	0.005988	0.006032
	Worse	0.006949	0.006812	0.006243	0.006149
	Standard deviation	0.000198	0.000173	$6.42E-05$	$3.87E-05$

5. Conclusion

Efficient and accurate time series prediction has been considered to be a difficult job. In this paper, we have presented complex-valued version of the S-system model to predict real-valued time series data. To search the optimal CVSS model, time series data need to be converted into complex data firstly. Complex-valued versions of the restricted additive tree and firefly algorithm are proposed to optimize the CVSS model. Stock index data from SSEI and NASI, exchange rate of Hong Kong dollar to renminbi, and Mackey–Glass chaos time series are collected to evaluate our proposed method. The forecasting capability of the CVSS model is compared with ARIMA, RBFNN, FNT, ODE, and S-system. The experiment results show that the predicted curves of the CVSS model are very closer to the target ones than other models. For SSEI dataset prediction, CVSS has the better performance than S-system, ODE, FNT, RBFNN, and ARIMA except for POCID. For NASI dataset prediction, compared with other five methods, CVSS could improve 0.103%–99.195% prediction performances in terms of RMSE, MAP, ARV, MAPE, R^2, and POCID. For RHKRMB dataset prediction, our method could improve 0.022%–95.29% prediction performances in terms of RMSE, ARV, MAPE, R^2, and POCID. Our method gains worse MAP performance, which reveals that the predicted errors by CVSS may be very large at some time points. For Mackey–Glass chaos time series prediction, CVSS could make 1.02%–90.74% prediction improvements in terms of six criterions.

The reasons that our method performs better are analyzed as follows. (1) A complex-valued parameter of the CVSS model contains two real-valued variables (real and imaginary parts), so the CVSS model contains twice more parameters than S-system. CVSS also has complex-valued functions. Complex-valued functions and parameters make the CVSS model to have stronger modeling and generalization abilities. (2) Complex-valued firefly algorithm has more population diversity and faster convergence speed than real-valued firefly algorithm and could search the optimal solution of the model faster. (3) The representation of CVRAT is very close to the form of the CVSS model, so CVRAT could search the optimal structure of the CVSS model faster.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare no conflicts of interest.

Authors' Contributions

W. B. conceived the method. B. Y designed the method and the website of this algorithm. Y. C. conducted the experiments, and W. B. wrote the main manuscript text. All authors reviewed the manuscript.

Acknowledgments

This work was supported by the talent project of "Qingtan scholar" of Zaozhuang University, the Natural Science Foundation of China (nos. 61702445 and 61902337), the Jiangsu Provincial Natural Science Foundation, China (no. SBK2019040953), the Shandong Provincial Natural Science Foundation, China (no. ZR2015PF007), the PhD research startup foundation of Zaozhuang University (no. 2014BS13), and the Zaozhuang University Foundation (no. 2015YY02).

References

[1] V. Cutello, G. Narzisi, and G. Nicosia, "A multi-objective evolutionary approach to the protein structure prediction problem," *Journal of the Royal Society Interface*, vol. 3, no. 6, pp. 139–151, 2006.

[2] J. F. Lawless, "A prediction problem concerning samples from the exponential distribution, with application in life testing," *Technometrics*, vol. 13, no. 4, pp. 725–730, 1971.

[3] G. S. Lingappaiah, "Bayesian approach to the prediction problem IN gamma population," *Nephron Clinical Practice*, vol. 11, no. 4, pp. 907–920, 1978.

[4] Y. Alaudah, M. Alfarraj, and G. Alregib, "Structure label prediction using similarity-based retrieval and weakly-supervised label mapping," *Geophysics*, vol. 84, no. 1, pp. 67–79, 2019.

[5] F. E. H. Tay and L. Cao, "Application of support vector machines in financial time series forecasting," *Omega*, vol. 29, no. 4, pp. 309–317, 2001.

[6] D. Ö. Faruk, "A hybrid neural network and ARIMA model for water quality time series prediction," *Engineering Applications of Artificial Intelligence*, vol. 23, no. 4, pp. 586–594, 2010.

[7] I. Weinreich, H. Rickert, and M. Lukaschewitsch, "Wavelet-based time series prediction for air traffic data," *Proceedings of SPIE—The International Society for Optical Engineering*, vol. 5266, pp. 238–248, 2004.

[8] M. Casdagli, "Chaos and DeterministicVersusStochastic non-linear modelling," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 54, no. 2, pp. 303–328, 1992.

[9] M. P. Hanias and D. A. Karras, “On efficient multistep non-linear time series prediction in chaotic diode resonator circuits by optimizing the combination of non-linear time series analysis and neural networks,” *Engineering Applications of Artificial Intelligence*, vol. 22, no. 1, pp. 32–39, 2009.

[10] C. M. Lee and C. N. Ko, “Time series prediction using RBF neural networks with a nonlinear time-varying evolution PSO algorithm,” *Neurocomputing*, vol. 73, no. 1–3, pp. 449–460, 2009.

[11] X. Zhang, Q. Zhang, G. Zhang, Z. Nie, and Z. Gui, “A hybrid model for annual runoff time series forecasting using elman neural network with ensemble empirical mode decomposition,” *Water*, vol. 10, no. 4, p. 416, 2018.

[12] G. Ding, S. S. Zhong, and Y. Li, “Time series prediction using wavelet process neural network,” *Chinese Physics B*, vol. 17, no. 6, pp. 1998–2003, 2008.

[13] M. Han, J. Xi, S. Xu, and F.-L. Yin, “Prediction of chaotic time series based on the recurrent predictor neural network,” *IEEE Transactions on Signal Processing*, vol. 52, no. 12, pp. 3409–3416, 2004.

[14] N. Baklouti, A. Abraham, and A. M. Alimi, “A Beta basis function interval type-2 fuzzy neural network for time series applications,” *Engineering Applications of Artificial Intelligence*, vol. 71, pp. 259–274, 2018.

[15] Y. H. Chen, B. Yang, J. W. Dong, and A. Abraham, “Time-series forecasting using flexible neural tree model,” *Information Sciences*, vol. 174, no. 3–4, pp. 219–235, 2004.

[16] S. C. Nayak, B. B. Misra, and H. S. Behera, “ACFLN: artificial chemical functional link network for prediction of stock market index,” *Evolving Systems*, vol. 10, no. 4, pp. 567–592, 2018.

[17] J. Gao, Y. L. Murphey, and H. Zhu, “Multivariate time series prediction of lane changing behavior using deep neural network,” *Applied Intelligence*, vol. 48, no. 1, pp. 1–15, 2018.

[18] S. Narejo and E. Pasero, “An application of internet traffic prediction with deep neural network,” *Multidisciplinary Approaches to Neural Computing*, Springer, Berlin, Germany, pp. 139–149, 2018.

[19] E. Hemberg, L. Ho, M. O’Neill, and H. Claussen, “A symbolic regression approach to manage femtocell coverage using grammatical genetic programming,” in *Proceedings of the 13th Annual Conference Companion on Genetic and Evolutionary Computation*, pp. 639–646, Dublin, Ireland, 2011.

[20] B. Can and C. Heavey, “Comparison of experimental designs for simulation-based symbolic regression of manufacturing systems,” *Computers & Industrial Engineering*, vol. 61, no. 3, pp. 447–462, 2011.

[21] E. J. Vladislavleva, G. F. Smits, and D. den Hertog, “Order of nonlinearity as a complexity measure for models generated by symbolic regression via pareto genetic programming,” *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 2, pp. 333–349, 2009.

[22] L. Billard and E. Diday, “Symbolic regression analysis,” *Studies in Classification Data Analysis & Knowledge Organization*, vol. 37, no. 24, pp. 6317–6328, 2002.

[23] A. Johari, G. Habibagahi, and A. Ghahramani, “Prediction of soil-water characteristic curve using genetic programming,” *Journal of Geotechnical and Geoenvironmental Engineering*, vol. 132, no. 5, pp. 661–665, 2006.

[24] H. Azzawi, J. Hou, Y. Xiang, and R. Alanni, “Lung cancer prediction from microarray data by gene expression programming,” *IET Systems Biology*, vol. 10, no. 5, pp. 168–178, 2016.

[25] Q. Zhang, B. Yang, L. Wang, and J. Jiang, “An improved multi-expression programming algorithm applied in function discovery and data prediction,” *International Journal of Information and Communication Technology*, vol. 5, no. 3/4, pp. 218–233, 2013.

[26] Y. Chen, B. Yang, Q. Meng, Y. Zhao, and A. Abraham, “Time-series forecasting using a system of ordinary differential equations,” *Information Sciences*, vol. 181, no. 1, pp. 106–114, 2011.

[27] W. Zhang and B. Yang, “Stock market forecasting using S-system model,” *Advances in Intelligent Systems and Computing*, pp. 397–403, 2017.

[28] T. Xiong, Y. Bao, Z. Hu, and R. Chiong, “Forecasting interval time series using a fully complex-valued RBF neural network with DPSO and PSO algorithms,” *Information Sciences*, vol. 305, pp. 77–92, 2015.

[29] S. Rashid, S. Saraswathi, A. Kloczkowski, S. Sundaram, and A. Kolinski, “Protein secondary structure prediction using a small training set (compact model) combined with a Complex-valued neural network approach,” *Bmc Bioinformatics*, vol. 17, no. 1, p. 362, 2016.

[30] T. K. S. Member, T. Y. Member, and H. S. Member, “Study on output prediction system of wind power generation using complex-valued neural network with multipoint GPV data,” *IEEJ Transactions on Electrical & Electronic Engineering*, vol. 8, no. 1, pp. 33–39, 2013.

[31] B. Yang, “Small-time scale network traffic prediction based on complex-valued neural network,” *IOP Conference Series: Materials Science and Engineering*, vol. 224, Article ID 012044, 2017.

[32] S. L. Goh and D. P. Mandic, “Nonlinear adaptive prediction of complex-valued signals by complex-valued PRNN,” *IEEE Transactions on Signal Processing*, vol. 53, no. 5, pp. 1827–1836, 2005.

[33] B. Shamima, R. Savitha, S. Suresh, and S. Saraswathi, “Protein secondary structure prediction using a fully complex-valued relaxation network,” in *Proceedings of the 2013 International Joint Conference on Neural Networks (IJCNN)*, pp. 3015–3022, Dallas, TX, USA, August 2013.

[34] S. Scardapane, S. Van Vaerenbergh, A. Hussain, and A. Uncini, “Complex-valued neural networks with non-parametric activation functions,” *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 4, no. 2, pp. 140–150, 2018.

[35] B. Yang and Y. Chen, “A new complex-valued polynomial model,” *Neural Processing Letters*, vol. 50, no. 3, pp. 2609–2626, 2019.

[36] Y. H. Chen, J. Yang, Y. Zhang, and J. W. Dong, “Evolving additive tree models for system identification,” *International Journal of Computational Cognition*, vol. 3, pp. 19–26, 2005.

[37] B. Yang, M. Y. Jiang, and Y. H. Chen, “A fast and efficient method for inferring structure and parameters of S-system models,” in *Proceedings of the 11th International Conference on Hybrid Intelligent Systems (HIS2011)*, pp. 235–240, Melacca, Malaysia, December 2011.

[38] C. Song, “A complex-valued firefly algorithm,” *Intelligent Computing Theories and Application*, vol. 11644, pp. 700–707, 2019.

[39] S. Padhan, R. K. Sahu, and S. Panda, “Application of firefly algorithm for load frequency control of multi-area interconnected power system,” *Electric Power Components and Systems*, vol. 42, no. 13, pp. 1419–1430, 2014.

[40] L. Li and Y. Zhou, "A novel complex-valued bat algorithm," *Neural Computing and Applications*, vol. 25, no. 6, pp. 1369–1381, 2014.

[41] A. A. Ariyo, A. O. Adewumi, and C. K. Ayo, "Stock price prediction using the ARIMA model," in *Proceedings of the 16th International Conference on Computer Modelling and Simulation*, pp. 106–112, Cambridge, UK, March 2015.

[42] M. Rout, B. Majhi, U. M. Mohapatra, and R. Mahapatra, "Stock indices prediction using radial basis function neural network," *Swarm, Evolutionary, and Memetic Computing*, vol. 7677, pp. 285–293, 2012.

[43] Z. Shi and M. Han, "Support vector echo-state machine for chaotic time-series prediction," *IEEE Transactions on Neural Networks*, vol. 18, no. 2, pp. 359–372, 2007.

[44] J. Zhao and X. Yu, "Adaptive natural gradient learning algorithms for Mackey-Glass chaotic time prediction," *Neurocomputing*, vol. 157, pp. 41–45, 2015.