Kernel Estimation of Volterra Using an Adaptive Artificial Bee Colony Optimization and Its Application to Speech Signal Multi-Step Prediction

YUMEI ZHANG[1,2], **YUYING RONG**[2], **CHENGLING YAN**[3],
JIANGSHAN LIU[2], **AND XIAOJUN WU**[1,2]

[1]Key Laboratory of Modern Teaching Technology, Ministry of Education, Shaanxi Normal University, Xi'an 710062, China
[2]School of Computer Science, Shaanxi Normal University, Xi'an 710062, China
[3]School of Environmental Sciences, Shandong University, Qingdao 266000, China

Corresponding author: Xiaojun Wu (xjwu@snnu.edu.cn)

ABSTRACT In order to solve parameters selection problem when applying recursive least square (RLS), least mean square (LMS) or normalized LMS (NLMS) algorithms to estimate kernels of second-order Volterra filter (SOVF), a novel adaptive gbest-guide artificial bee colony (AGABC) optimization algorithm is used to derive kernels of Volterra, that is a type of the AGABC-SOVF prediction model with an explicit configuration for speech signal is proposed. The AGABC algorithm modifies the solution search equation of ABC algorithm and combines the best solution with neighborhood information at present iteration, which not only ensures the exploration of the global optimization algorithm but also improves the exploitation. The AGABC-SOVF model is performed to predict speech signal series of the given English phonemes, sentences, and chaotic time series. Simulation results based on benchmark function show that AGABC algorithm performs faster convergence in achieving higher quality solutions than original ABC and other improved ABC algorithms. Prediction results of applying the AGABC-SOVF model to multi-step predictions for Lorenz time series reveal its stability and convergence properties. For the measured multi-frame speech signals, prediction accuracy and length of multi-step prediction using the AGABC-SOVF model are better than that of the ABC-SOVF model. The AGABC-SOVF model can better predict chaotic time series and the real measured speech signal series.

INDEX TERMS Speech signal, kernel estimation, Volterra model, ABC algorithm, multi-step prediction.

I. INTRODUCTION

In real-world engineering, people often encounter a large number of nonlinear problems. Over the last few decades, a variety of nonlinear adaptive filters have been established. Among them, the Volterra filter is one of the most commonly used models in system recognition, chaos prediction, image processing, spread-spectrum communications and other fields [1]–[10]. Jian *et al.* [11] proposed a method of combining recursive least square (RLS) and least mean square (LMS) to compensate the nonlinear and memorial distortion of orthogonal frequency division multiplexing (OFDM) power amplifier. Many strategies based on RLS and

LMS algorithms had been designed for system identification of Volterra model [12]–[15]. Batista and Rui [16] combined LMS and normalized LMS (NLMS) algorithms to update the coefficients of the sparse-interpolated Volterra structure. In summary, LMS, NLMS and RLS algorithms have been widely applied in Volterra model. However, neither the convergence of LMS algorithm nor that of RLS algorithm is efficient enough, whilst the improvement of convergence rate by NLMS algorithm is at the expense of computational complexity. Besides, in the perspective of application, there are parameters selection issues on these three algorithms, which may cause instability problem.

Artificial bee colony (ABC) algorithm enlightened by the behavior of honey bees, is introduced by Karaboga and Basturk [17], which is a stochastic optimization algorithm. The algorithm has the advantages of good stability, less control parameters and simple implementation. However, ABC algorithm also existed many problems, such as weak local search ability and easy to fall into evolutionary stagnation. To deal with the problems of ABC algorithm, many scholars have done various experiments and researched on modification strategies. One of discoveries was that the position updating equation of ABC algorithm, which can lead to good capability on exploration at the expense of relatively poor competence on exploitation [18]. Thus, many modified ABC algorithms have been proposed. For example, Gao *et al.* [19] introduced the differential evolution to get a ABC/best/1 algorithm. An improved ABC algorithm named gbest-guide ABC (GABC) inspired by Zhu and Kwong [20]. These improved ABC algorithms improve performance and the candidate solutions generated in algorithms are very close to the current global optimal solution, but exploitation and exploration of ABC algorithm are still unbalanced.

Based on above analysis, an adaptive global-best ABC (AGABC) optimization algorithm is proposed in this paper, which represents a novel approach to Volterra filter by modifying the solution search equation. To avoid solutions trapping in local optimum initially, a variable weight coefficient and the neighbor information are introduced based on the original search equation. Then, the candidate solutions based on the current best solution are generated to find the global optimal. Besides, the aim to enhance global convergence, we add the chaotic model to initialize the population. The AGABC algorithm was proposed by Liu et al. [21], which demonstrated good performance. Based on this previous work, the AGABC optimization algorithm is introduced to second-order Volterra filter (SOVF) for solving the kernel coefficients. We propose a novel AGABC-SOVF prediction model with an explicit configuration for speech signal. In experiments, we apply the AGABC-SOVF model to multi-step prediction of speech signals. For Lorenz chaotic time series and real measured speech signal series, simulations of multi-step prediction employing the AGABC-SOVF model are performed. Also, results of multi-step prediction using the AGABC-SOVF model are compared with the ABC-SOVF model. Then, under certain errors, prediction length comparisons of multi-step prediction between ABC-SOVF model and AGABC-SOVF model are performed by using Lorenz chaotic time series and phonetic symbol [g].

This paper is organized as follows. Section II introduces Volterra filter model. Section III describes the original ABC algorithm. Section IV presents the AGABC-SOVF model. Section V shows and discusses numerical simulation results, and finally, Section VI offers conclusions.

II. VOLTERRA FILTER MODEL
Modeling and identification tools for nonlinear systems are often used for signal processing. One such widely utilized modeling tool is Volterra filter. The salient features of the second-order truncated Volterra filter are briefly described in this section. A second-order truncated Volterra filter [22], used in modeling and analysis of nonlinear systems, is defined as follows:

$$\hat{y}(n) = h_0 + \sum_{i=0}^{m-1} h_1(i; n-1)x(n-i)$$
$$+ \sum_{i=0}^{m-1}\sum_{j=0}^{m-1} h_2(i, j; n-1)x(n-i)x(n-j) \quad (1)$$

where $x(n)$ and $\hat{y}(n)$ are input signal and prediction output signal at time instant n, respectively. h_0 is a constant (for simplicity, let h_0 equals to 0), m is memory length of the filter. $h_1(i; n-1)$ and $h_2(i, j, n-1)$ are filter kernel coefficients to be updated at time instant n, which are linear and quadratic coefficients, respectively. Let

$$H(n-1) = [h_1(0; n-1), h_1(1; n-1),$$
$$\cdots, h_1(m-1; n-1), h_2(0, 0; n-1), h_2$$
$$(0, 1; n-1), \cdots, h_2(m-1, m-1; n-1)]^T$$
$$X(n) = [x(n), x(n-1), \cdots, x(n-(m-1)),$$
$$x^2(n), x(n)x(n-1), \cdots, x^2(n-(m-1))]^T$$

where $[\cdot]^T$ denotes vector transpose. So, Equation (1) can be denoted by the following vector form

$$\hat{y}(n) = H^T(n-1)X(n). \quad (2)$$

Let $e(n)$ be a priori error signal, is desired output at time instant. So square error at time instant can be expressed

$$e^2(n) = (y(n) - \hat{}(n))^2. \quad (3)$$

For the second-order Volterra model formulated by Equation (2), iterative equation for using standard LMS algorithm can be expressed

$$H(n) = H(n-1) + 2\mu e(n)X(n), \quad (4)$$

where μ is a convergence factor that controls convergence speed and stability of LMS algorithm. To ensure convergence of the algorithm, $0 < \mu < 1/\lambda_{max}$, where λ_{max} is the largest eigenvalue of autocorrelation matrix of input signal vector $X(n)$ [23], [24]. So, when the largest eigenvalue is relatively large, has to take a relatively small value, which results in slower convergence speed.

III. CONVENTIONAL ABC ALGORITHM
ABC algorithm, which is stimulated by the foraging behavior of bees, is a kind of swarm intelligence algorithm. In ABC algorithm, the function extrema value required optimization is simulated as food sources and evaluated by fitness function. The solutions generated during the function optimization process are modeled as bees to search within the solution space of the function. There are three types of bees: employed bees, onlooker bees and scout bees according to different divisions of labor. Employed bees randomly select food sources at the

initial stage and search neighbors based on greedy selection to optimize their selections. Onlooker bees choose the employed bees to follow according to the feedback information on food sources qualities and transform into employed bees for neighbor search. Once the selection of food source has not been updated for certain times, employed bees will transform into scout bees and new scout bees will continue to seek new food sources. In forms of transformation among three types, bees cooperated with one another to find the best food source which represents the optimal extremum value of the function. The detailed algorithm is as follows [25].

First, in the search space, initial food sources are generated randomly and spread uniformly. This process can be defined as follow:

$$x_{ij} = x_j^l + rand(0, 1) \cdot (x_j^u - x_j^l) \quad (5)$$

where $i \in \{1, 2, \cdots, SN\}$, SN is the total number of colony, $j \in \{1, 2, \cdots, n\}$, x_j^l and x_j^u are lower bound and upper bound for the jth dimension, respectively.

A. EMPLOYED BEES PHASE

Each employed bee searches the neighbors of initial food sources. A new selection of food source v_{ij} is generated during this process and is updated in accordance with the information provided by neighbors of its present position x_{ij}. v_{ij} can be calculated as follow:

$$v_{ij} = x_{ij} + \varphi_{ij} \cdot (x_{ij} - x_{neighbor}) \quad (6)$$

where $neighbor \in \{1, 2, \cdots, SN\}$ and $j \in \{1, 2, \cdots, n\}$ are random numbers, and $neighbor \neq i$, φ_{ij} is a random number in the range of $[-1, 1]$ that is evenly distributed to control the searching range. v_{ij} and x_{ij} are selected by employed bees based on the principle of greedy selection. Compared to the original food sources, if the new one is better, which means $fit(v_{ij}) \geq fit(x_{ij})$, x_{ij} will be replaced by v_{ij}, otherwise x_{ij} will be retained.

B. ONLOOKER BEES PHASE

The food source shared by employed bees is selected by each onlooker bee and the selection depends on the probability of its fitness value, which can be expressed as following equation:

$$p_i = \frac{fit(x_i)}{\sum_{i=1}^{SN} fit(x_i)} \quad (7)$$

where $fit(x_i)$ is the fitness value at the ith food source. For food sources with high fitness value, there will be more onlooker bees searching around and in this way convergence speed of the algorithm can be improved.

C. SCOUT BEES PHASE

If a food source has not been updated after visiting a certain times, this source will be ignored and the corresponding employed bee will transformed into a scout bee. The scout bees randomly select a new food source in the global scope
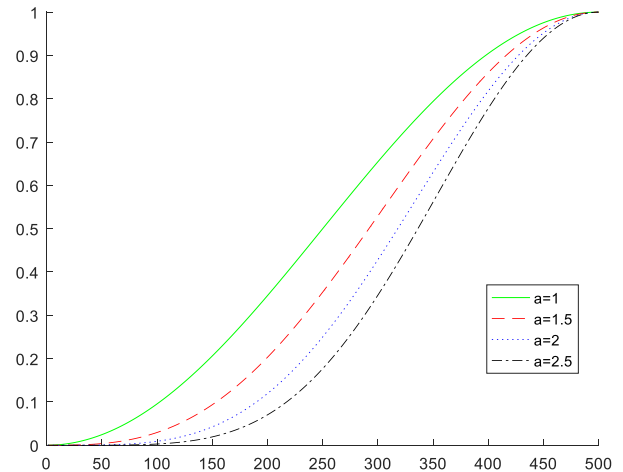

FIGURE 1. The relationship between *cycle* and ρ.

to continue the neighbor search. The generation of new food source selection can be described as Equation (5).

IV. THE AGABC-SOVF MODEL
A. PROPOSED MODEL

ABC algorithm, popular for its simplicity, has strong robustness and good operability. We can use this algorithm to solve the Volterra kernel problem. However, the search formula of ABC algorithm is adept at exploration but less effective in exploitation, which means that ABC algorithm has an excellent capability in finding the optimal extremum value of the function in the global scope but relatively weak in seeking the extremum of the function with higher precision in the local scope. For this reason, we modify the solution of search equation in the onlooker bees' phase of ABC algorithm to better derive the Volterra kernel.

The nonlinear single-input and single-output system can be described as Equation (2). In the linear combination relationship, the kernel coefficients of SOVF model is represented as a vector H, which is substituted into the AGABC algorithm by its role as a food source. In calculation process, the evaluation function for each food source is defined as

$$F(H_i) = \frac{1}{L} \sum_{t=1}^{L} (y(t) - H_i^T \cdot X(t))^2 \quad (8)$$

where $i = 1, 2, 3, \cdots, SN$, and L is the length of signal data. The optimal predictive effect of training is to iteratively compute the minimum value of the objective function. The algorithm process is as follows

1) INITIAL COLONY PHASE

The SN employed bees in the whole search space are evenly and randomly distributed to ensure diversity of solutions, and the position of the search space equals to the food source, corresponding to the kernel vector H of SOVF model.

2) EMPLOYED BEES PHASE

A neighbor node is randomly generated at the current food source, then the fitness values of these two positions are

TABLE 1. Numerical benchmark function.

Function name	Function	Characteristic	Ranges	Global minimum
Sphere	$f(x)=\sum_{i=1}^{D}x_i^2$	Uni-modal, Separable	$-100\le x_i \le 100$	0
Griewank	$f(x)=\frac{1}{4000}\left(\sum_{i=1}^{D}x_i^2\right)-\left(\prod_{i=1}^{D}\frac{x_i}{\sqrt{i}}\right)+1$	Multi-modal, Non-Separable	$-600\le x_i \le 600$	0
Booth	$f(x,y)=(x+2y-7)^2+(2x+y-5)^2$	Multi-modal, Separable	$-10\le x_i \le 10$	0
Schaffer	$f(x)=0.5+\frac{\cos[\sin(|x^2-y^2|)]^2-0.5}{[1+0.001(x^2+y^2)]^2}$	Multi-modal, Non-Separable	$-100\le x_i \le 100$	0

TABLE 2. Performance comparisons of AGABC with other algorithms when the maximum number of iterations is 1000.

Function		ABC	ABC/best/1	GABC	AGABC
Sphere	Mean	7.032e-16	4.766e-42	3.576-30	5.39e-37
	SD	5.218e-16	3.235e-49	3.774e-30	2.830e-36
	Best	6.314e-16	2.354e-53	2.407e-31	3.504e-37
Griewank	Mean	6.214e-3	1.399e-3	3.691e-4	0
	SD	3.154e-3	4.909e-3	1.482e-3	0
	Best	9.214e-15	0	0	0
Booth	Mean	3.145e-10	0	4.358e-14	0
	SD	5.147e-9	0	9.147e-14	0
	Best	96.14-10	0	0	0
Schaffer	Mean	3.215e-9	3.217e-14	9.475e-14	2.146e-14
	SD	3.217e-9	2.694e-15	6.217e-15	6.244e-14
	Best	6.217e-10	8.774e-15	1.865e-14	3.214e-16

TABLE 3. Performance comparisons of AGABC with other algorithms when the maximum number of iterations is 1500.

Function		ABC	ABC/best/1	GABC	AGABC
Sphere	Mean	6.769e-16	7.070e-46	3.576-30	2.731e-35
	SD	1.266e-15	2.796e-43	3.774e-30	1.278e-34
	Best	7.327e-16	1.826e-43	1.865e-14	2.731e-35
Griewank	Mean	2.512e-4	5.634e-19	2.465e-5	0
	SD	1.76e-3	9.254e-9	1.850e-3	0
	Best	1.473e-15	0	0	0
Booth	Mean	9.217e-9	6.324e-10	9.473e-16	0
	SD	6.314e-9	3.458e-9	5.189e-15	0
	Best	7.214e-12	0	0	0
Schaffer	Mean	2.373e-9	8.214e-14	6.214e-14	6.217e-14
	SD	1.958e-9	7.314e-15	6.217e-15	5.12 7e-14
	Best	4.664e-10	6.214e-14	6.217e-14	6.217e-16

compared. If $F(H_i) < F(H_{i-1})$ H(), the current position is replaced by the neighbor node, otherwise an accumulative count NP is performed. The original search equation is written as

$$H_{ij} = H_{ij} + \varphi_{ij} \cdot (H_{ij} - H_{neighbor}) \qquad (9)$$

3) ONLOOKER BEES PHASE
In this stage, a new search equation is proposed to avoid biasing the optimal solution during initialization, which can obtain information from the global optimum and the neighbors. This new equation is inspired by a literature which has proposed the converge-onlookers ABC (COABC) algorithm, in which the new solution is generated based on the current global optimal food source [26]. From simulation results of the algorithm, it can be seen that the idea of sharing global information is beneficial for improving exploitation performance. The current positions of the food source are sorted according to their fitness values. The best position is denoted as H_{best}, and a neighbor food source is randomly selected.
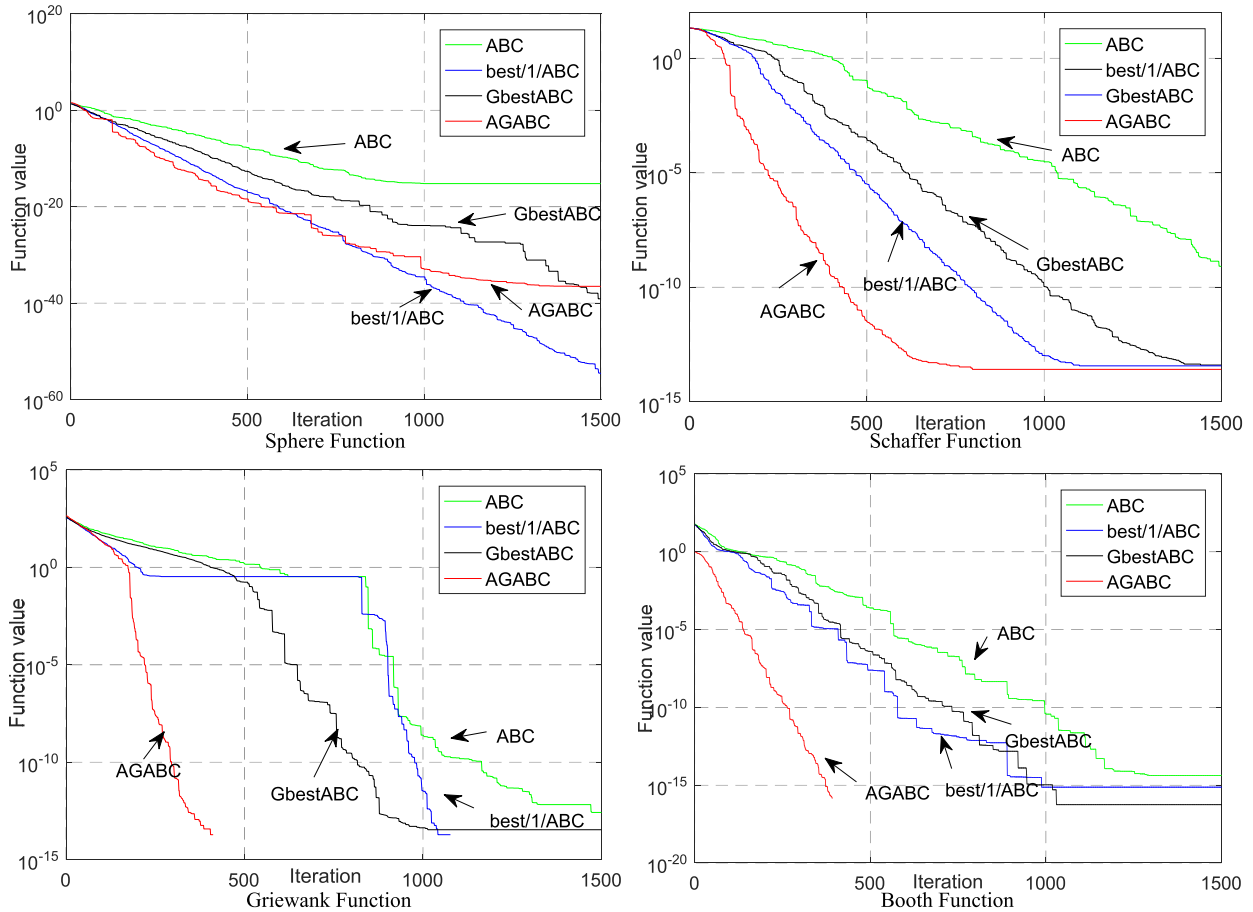
FIGURE 2. Iteration convergence for ABC, GbestABC, ABC/best/1 and AGABC ALGORITHMS.
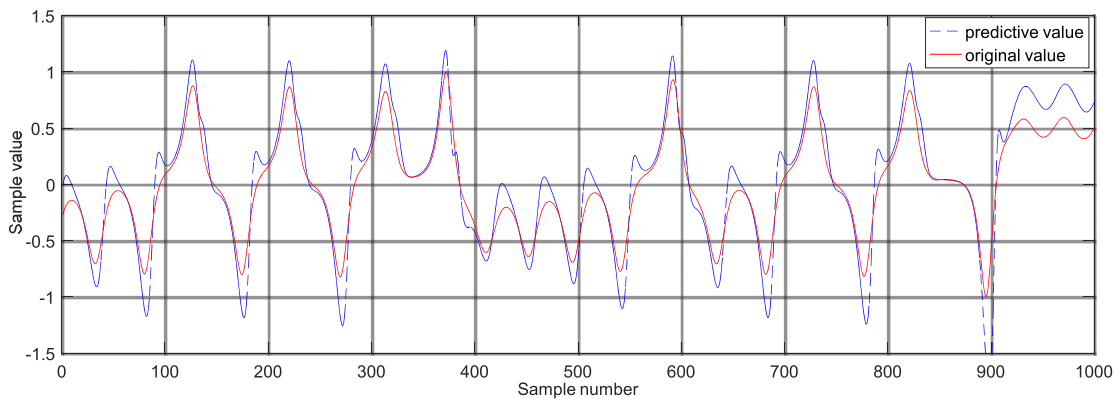

FIGURE 3. Prediction of Lorenz series using ABC-SOVF model.

The search equation can be written as

$$H_{ij} = \omega \cdot H_{best} + c_1 \cdot \phi \cdot (H_{best} - H_{ij})$$
$$+ c_2 \cdot \varphi \cdot (H_{neighbor} - H_{ij}) \quad (10)$$

where ω is the inertia weight that controls the impact of the global best solution at current iteration; c_1 and c_2 are positive constant parameters; ϕ and φ are uniform random numbers in the range $[-1,1]$, $j \in (1, 2, \cdots, n)$. In our proposed method,

the parameter ω is updated adaptively, which is computed by:

$$\omega = \omega_{\min} + \rho \cdot (\omega_{\max} - \omega_{\min}) \quad (11)$$

$$\rho = ((-\cos \frac{cycle}{\max cycle} \cdot \pi + 1) \cdot \frac{1}{2})^a \quad (12)$$

where $\omega_{\min}$ and $\omega_{\max}$ represents the lower bound and upper bound of the ω respectively; max *cycle* is the maximum number of iteration; *cycle* is the number of iteration. ρ changes
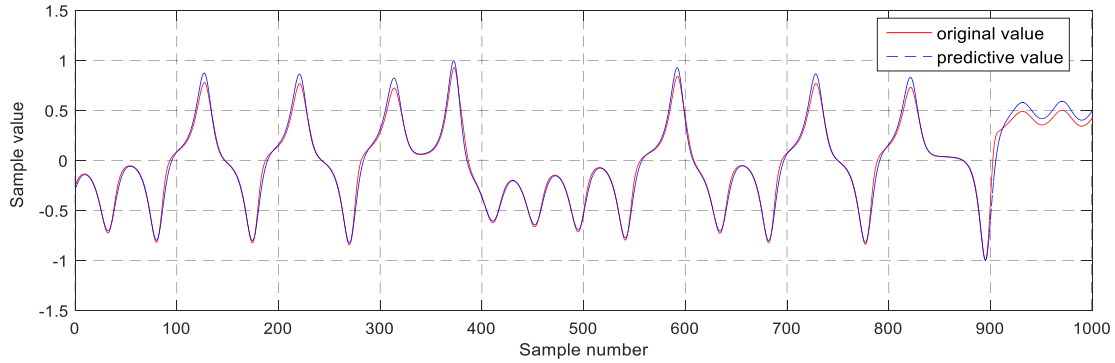
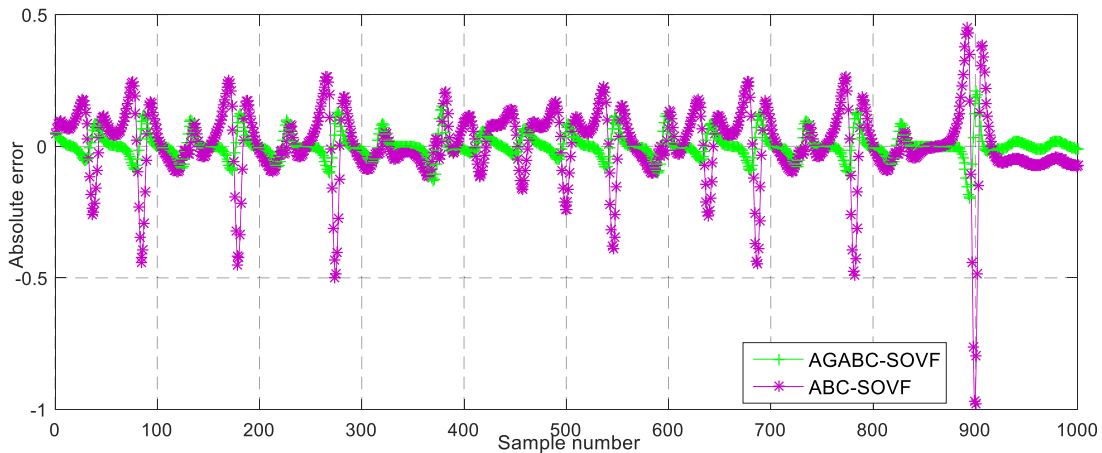**FIGURE 4.** Prediction of Lorenz series using AGABC-SOVF model.

FIGURE 5. Comparison of prediction errors of the two methods.

dynamically with increase of iteration, as shown in Figure 1. The purpose of ρ is to enable space to be entirely searched, and it can release the search method at initial step. Then, the information from best solution and neighbor is obtained by the above formula, which avoids stagnation around local minima.

4) SCOUT BEES PHASE
Whether or not the value of the accumulative count NP reaches the preset value of limit will be judged. If it exceeds, the current food position is randomly assigned, otherwise, the next iteration calculation is continued. The search equation can be written as

$$H_{ij} = H_j^l + rand(0, 1) \cdot (H_j^u - H_j^l) \qquad (13)$$

5) JUDGEMENT
In multiple iterations of the AGABC algorithm, each iteration needs to determine whether the current number of steps is less than or equal to a preset maximum number of iterations or a minimum error requirement. If it meets requirements, the output vector H is terminated; otherwise, the next calculation is continued. The optimal solution of the kernel coefficient of the Volterra model is shown in the following formula:

$$H_{best} = \arg\min_{1 < i < SN} [F(H_{ij})] \qquad (14)$$

B. CHAOS INITIALIZATION
Population initialization is an indispensable and vital part of the bionic evolutionary algorithm. It not only has a significant impact on convergence speed of the algorithm, but also imposes certain constraints on the quality of the final solution to the problem. In traditional ABC algorithm, initialization of the honeybee population is mainly achieved through the use of uniformly distributed random sequence generated by the random function. Because the generated random sequence presents periodic cyclic repetition which causes a risk of degradation in individual diversity of the population, this method of random initialization reveals great limitations. It has been proven in the literature that chaotic time series with its unique characteristics have significant improvements over random series [27]. Therefore, chaotic series can be used to promote diversification of individual populations to further enhance convergence rate of the algorithm. The chaotic map ch_i is generated by the following calculation equation:

$$ch_{i+1} = 4 \cdot ch_i \cdot (1 - ch_i), \quad 1 \leq K \qquad (15)$$

where K is the length of chaotic series, $ch_0 \in (0, 1)$ is a random number. Then ch_i will replace the rand(0,1) in Equation (13).
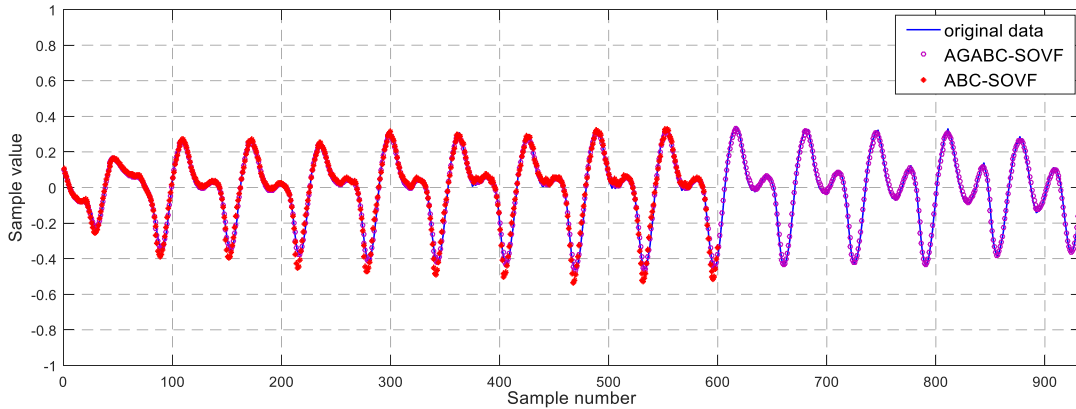
V. PERFORMANCE EVALUATION OF THE

FIGURE 6. Predictive length comparison of multi-step prediction for phonetic symbol [g] when using AGABC-SOVF and ABC-SOVF models.
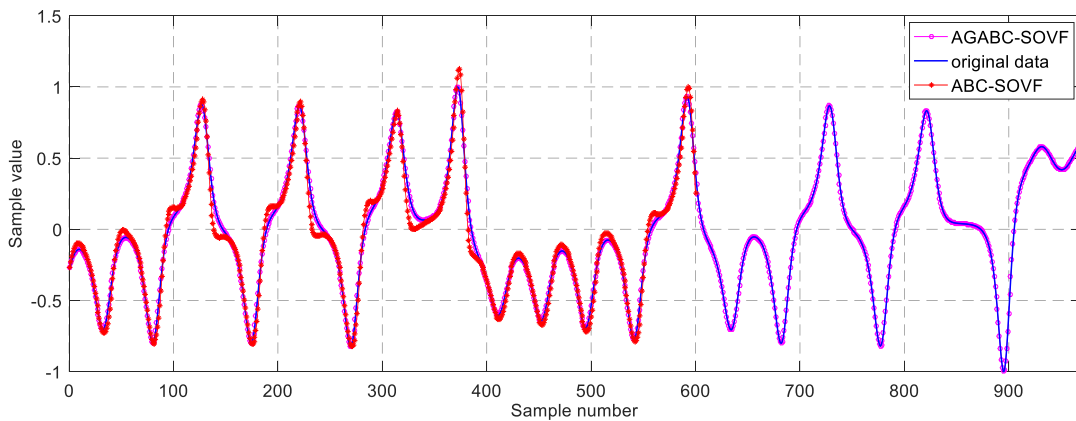

FIGURE 7. Predictive length comparison of multi-step prediction for Lorenz series when using AGABC-SOVF and ABC-SOVF models.

PROPOSED METHOD

A. PERFORMANCE COMPARISONS OF ORIGINAL ABC, ABC/BEST/1, GABC AND AGABC ALGORITHMS

Some classical benchmark functions presented in Table 1 were used to evaluate performance of the proposed algorithm. Results of AGABC algorithm have been compared with the results of the original ABC, ABC/best/1 and GABC algorithms. Each of experiments was repeated 30 times with different random seeds, and experiment stops when the number of iterations reaches max *cycle*.

The common control parameters of the algorithms are given as follows. The population size is 25, and the limit is 100. AGABC algorithm has a few parameters: $\omega_{\min} = 0.3$, $\omega_{\max} = 1.0$, $c_1 = 0.02$, $c_2 = 0.02$ [21]. The best value (Best), the mean value (Mean) and the standard deviation (SD) for different dimensions and iterations are recorded. The maximum number of iterations are 1000 and 1500 for dimension 20 and 30, respectively. In Table 2 and Table 3, as can be seen Mean values of AGABC algorithm are smaller than those of the original ABC algorithm in all test functions. Moreover, the performance of AGABC algorithm is more outstanding than other algorithms in both Mean values and convergence iteration when it minimize the multi-modal

functions. In short, this proves that the AGABC algorithm has better optimization performance.

Figure 2 shows iteration convergence of the four algorithms, and clearly can be seen that comparisons between the algorithms. The curves in the figure represent changes of the function values of the algorithms in the iteration. When the curve becomes a flat line, number of iterations reaches the global optimization. In the optimization of a unipolar value function, the effect is slightly worse than best/1/ABC, but better than the other two. However, the AGABC algorithm has obvious performance advantages over other three multi-modal functions, and is far superior to other three algorithms in convergence speed and solution accuracy.

B. ANALYSIS OF MULTI-STEP PREDICTION OF LORENZ CHAOTIC SERIES

In experiments, chaotic map is directly iterated according to initial values. Integration step size and initial condition for Lorenz series are taken as 0.01 and $[-1\ 0\ 1]$ respectively. Simulation series with 1300 data points is computed by using fourth-order Runge-Kutta integration method. The first 300 data are for training and the remaining 1000 data are
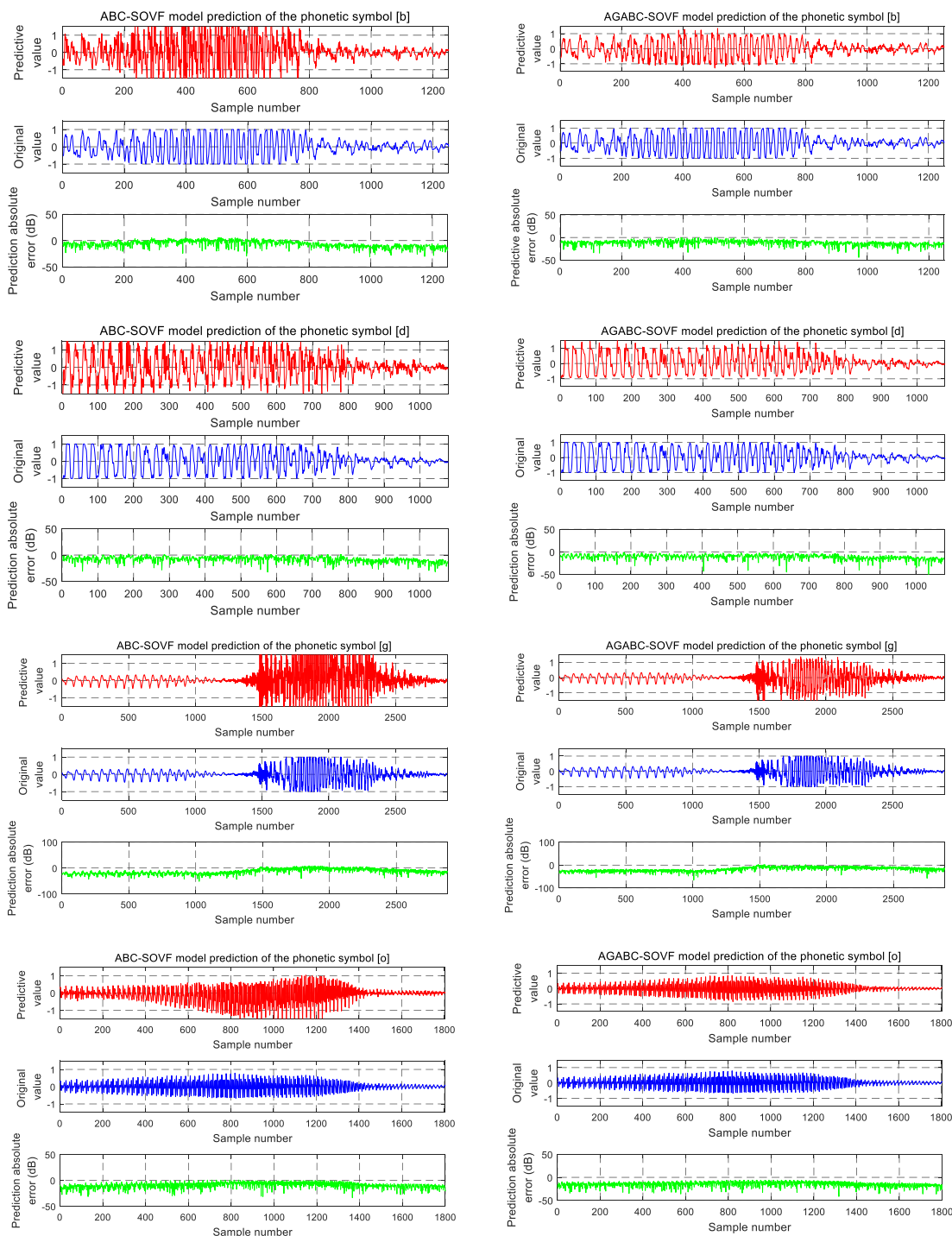
FIGURE 8. Predictive comparison of two models for phonetic symbols [b],[d],[g] and [o].

for test. The Lorenz series equation is defined as follows

$$\begin{cases} \dot{q} = a\,(r - q) \\ \dot{r} = -qs + cs - r \\ \dot{s} = qr - bs \end{cases} \quad (16)$$

where $a = 16, b = 4, c = 45.92, q\,(0) = -1, r\,(0) = 0, s\,(0) = 1$.

Multi-step prediction errors for x component of Lorenz series by applying the ABC-SOVF model and the AGABC-SOVF model are shown in Figure 3 and Figure 4, respectively. It is noted that Lorenz series is first normalized before simulation by the following formula:

$$y\,(i) = \frac{2\,(x\,(i) - \min x\,(i))}{\max x\,(i) - \min x\,(i)}, \quad i = 1, 2, \cdots, n,$$
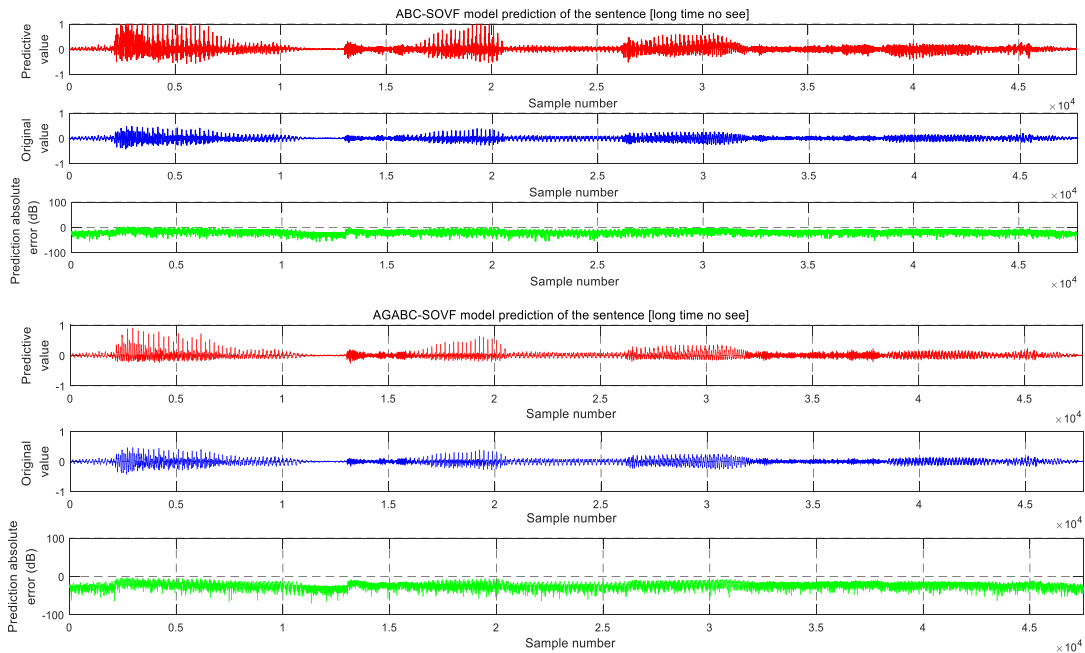
FIGURE 9. Predictive comparison of two models for the sentence [long time no see].

where $x(i)$ is original time series, $\max x(i)$ and $\min x(i)$ denote its maximum and minimum values, respectively. Normalized time series is represented as $y(i)$.

Figure 3 and Figure 4 show that performance of AGABC-SOVF is more prominent, especially for prediction of the first 100 points, that are almost identical to the original series. The predicted values of the two models can reflect the original signal values more intuitively, but the predicted values of the AGABC-SOVF model are closer to the original values than the ABC-SOVF model in the large fluctuation part. When a spike in the original series, both the predicted results of the two models show errors, but the error distance of AGABC-SOVF model is far less than that of ABC-SOVF model. From Figure 5 we can see that absolute error of the AGABC algorithm is much lower than that of the ABC algorithm, especially when sample number is bigger. Based on the above analysis, it can be concluded that the AGABC-SOVF model performs well in prediction, and has an obvious pre-improvement advantage.

C. COMPARISONS OF PREDICTION LENGTH OF MULTI-STEP PREDICTION

This experiment is mainly used to test prediction length of multi-step prediction by applying the AGABC-SOVF model. The first 200 points of speech signal of certain length are used as training to configure the model, which is used for prediction after 200 points, then prediction is terminated when the error is greater than 0.05.

Figure 6 and Figure 7 show that AGABC-SOVF model and ABC-SOVF model are used in predictions of phonetic symbol [g] and Lorenz series, respectively. We note that the

TABLE 4. MSE comparison of different methods for phonetic symbols [b], [d], [g], and [o].

	[b]	[d]	[g]	[o]
ABC-SOVF	0.1434	0.2434	0.1964	0.1134
AGABC-SOVF	0.0812	0.0631	0.0824	0.0724

prediction results of AGABC-SOVF model and ABC-SOVF model for about 600 steps are basically consistent with the original series. In contrast, AGABC-SOVF model is able to continue prediction for the next 300 points and the error is very small. So, the AGABC-SOVF model is obviously better than that of the ABC-SOVF model in length prediction.

D. PREDICTIONS OF REAL MEASURED SPEECH SIGNAL SERIES USING AGABC-SOVF AND ABC-SOVF MODEL

In this experiment, validity of the proposed nonlinear prediction model is verified by the results of the multi-frame speech signal. Experimental data contains four phonetic symbols [b], [d], [g], [o] and the sentence (long time no see). In order to verify the validity and generalization ability of our method, the AGABC-SOVF model is applied to the prediction of sentence sample to verify the effectiveness of the model. The pre-processing of the signal is as same as above, and ABC-SOVF model and AGABC-SOVF model were used to predict and compare the waveform, absolute error and MSE. Experimental results of four phonetic symbols and the sentence are shown in Table 4, Figure 8 and Figure 9, respectively.

From Figure 8 and Figure 9, it can be seen that prediction values of two types of model can be more intuitive response to the original signal. But in the intermediate part of larger fluctuations, prediction values obtained from the AGABC-SOVF model are more close to the original signals. And for each frame data, from Table 4, it can be seen that the AGABC-SOVF model has a much smaller MSE values than that of the ABC-SOVF model, which shows that the AGABC-SOVF prediction model proposed in this paper has better prediction results for multi-frame speech signals.

VI. CONCLUSION

In this paper, the AGABC algorithm based on the search equation of onlooker stage is proposed. In the algorithm, we study a new search equation which is based on the search equation of the original ABC algorithm, that combines a variable weight coefficient with the neighbor information. Experimental results show that the AGABC algorithm guarantee a good balance between exploitation and exploration in global optimization. After that, the AGABC algorithm is applied to SOVF model to estimate kernel coefficients. Performance of the AGABC algorithm was compared with the original ABC and other two improved algorithms ABC/best/1 and GABC. Results show that AGABC algorithm achieves improved performance in multi-modal functions. For Lorenz series and real measured speech signal series, when MSE is used as an evaluation, multi-step prediction accuracy of the proposed AGABC-SOVF model is better than that of the ABC-SOVF model. Within a certain error permissible range, the AGABC-SOVF model predicts longer than the ABC-SOVF model. Meanwhile, the AGABC-SOVF model can better reflect trends and regularities of the speech signal series and fully meet requirements for speech signal prediction. The proposed model can present a nonlinear analysis and more valuable model structure for speech signal series, and opens up a new way to speech signal reconstruction and compression coding.

REFERENCES

[1] D. W. Griffith, Jr. and G. R. Arce, "A partially decoupled RLS algorithm for Volterra filters," *IEEE Trans. Signal Process.*, vol. 47, no. 2, pp. 579–582, Feb. 1999.

[2] S. K. Patel, S. R. Arya, and R. Maurya, "Nonlinear adaptive volterra filter for control of distribution static compensator," *IEEE J. Emerg. Sel. Topics Power Electron.*, vol. 5, no. 1, pp. 559–567, Mar. 2017.

[3] M. Sabry-Rizk and W. Zgallai, "Novel Volterra predictor based on state-space equilibrium of nonlinear single- or multifractal signals," *Proc. SPIE*, vol. 4116, pp. 322–333, Nov. 2000.

[4] S. Boyd and L. Chua, "Fading memory and the problem of approximating nonlinear operators with Volterra series," *IEEE Trans. Circuits Syst.*, vol. CAS-32, no. 11, pp. 1150–1161, Nov. 1985.

[5] Y. Birkelund, A. Hanssen, and E. J. Powers, "On the estimation of nonlinear volterra models in offshore engineering," *Int. J. Offshore Polar Eng.*, vol. 12, no. 1, pp. 12–20, 2003.

[6] V. J. Mathews and G. L. Sicuranza. (2000). *Polynomial Signal Processing.* [Online]. Available: http://www.yidu.edu.cn/

[7] C.-H. Cheng and E. J. Powers, "Optimal Volterra kernel estimation algorithms for a nonlinear communication system for PSK and QAM inputs," *IEEE Trans. Signal Process.*, vol. 49, no. 1, pp. 147–163, Jan. 2001.

[8] A. Guerin, G. Faucon, and R. Le Bouquin-Jeannes, "Nonlinear acoustic echo cancellation based on Volterra filters," *IEEE Trans. Speech Audio Process.*, vol. 11, no. 6, pp. 672–683, Nov. 2003.

[9] C.-M. Huang, Y.-F. Chen, S. Xu, and H. Zhou, "The vehicular social network (VSN)-based sharing of downloaded geo data using the credit-based clustering scheme," *IEEE Access*, to be published.

[10] H. Yang, S. Qu, C. Chen, and B. Yang, "Multiple objects tracking with improved sparse representation and rank based dynamic estimation," *IEEE Access.*, vol. 6, pp. 42264–42278, 2018.

[11] W. Jian, C. Yu, J. Wang, J. Yu, and L. Wang, "OFDM adaptive digital predistortion method combines RLS and LMS algorithm," in *Proc. ICIEA*, May 2009, pp. 3900–3903.

[12] A. K. Kohli and A. Rai, "Numeric variable forgetting factor RLS algorithm for second-order volterra filtering," *Circuits Syst. Signal Process.*, vol. 32, no. 1, pp. 223–232, Feb. 2013.

[13] X. Wen and Y. Chen, "Research of the nonlinear system identification based on the volterra RLS adaptive filter algorithm," *Telkomnika Indonesian J. Electr. Eng.*, vol. 11, no. 5, pp. 2277–2283, 2013.

[14] Z. Sigrist, E. Grivel, and B. Alcoverro, "Estimating second-order Volterra system parameters from noisy measurements based on an LMS variant or an errors-in-variables method," *Signal Process.*, vol. 92, no. 4, pp. 1010–1020, Apr. 2012.

[15] H. Zhang, Y. Qi, H. Zhou, J. Zhang, and J. Sun, "Testing and defending methods against DOS attack in state estimation," *Asian J. Control*, vol. 19, no. 4, pp. 1–11, 2016.

[16] E. L. O. Batista and R. Seara, "A fully LMS/NLMS adaptive scheme applied to sparse-interpolated Volterra filters with removed boundary effect," *Signal Process.*, vol. 92, no. 10, pp. 2381–2393, Oct. 2012.

[17] D. Karaboga and B. Basturk, "A powerful and efficient algorithm for numerical function optimization: Artificial bee colony (ABC) algorithm," *J. Global Optim.*, vol. 39, no. 3, pp. 459–471, Apr. 2007.

[18] N. Imanian, M. E. Shiri, and P. Moradi, "Velocity based artificial bee colony algorithm for high dimensional continuous optimization problems," *Eng. Appl. Artif. Intell.*, vol. 36, pp. 148–163, Nov. 2014.

[19] W. Gao, S. Liu, and L. Huang, "A global best artificial bee colony algorithm for global optimization," *J. Comput. Appl. Math.*, vol. 236, no. 11, pp. 2741–2753, 2012.

[20] G. Zhu and S. Kwong, "Gbest-guided artificial bee colony algorithm for numerical function optimization," *Appl. Math. Comput.*, vol. 217, no. 7, pp. 3166–3173, Dec. 2010.

[21] J. Liu, Y. Zhang, and S. Bai, "An adaptive artifical bee colony algorithms based on global best guide," in *Proc. 13th Int. Conf. Comput. Intell. Secur. (CIS)*, Dec. 2017, pp. 211–215.

[22] R. Eberhart, Y. Shi, and J. Kennedy, *Swarm Intelligence.* San Fransisco, CA, USA: Morgan Kaufmann, 2001.

[23] J. C. Bansal, H. Sharma, and S. S. Jadon, "Artificial bee colony algorithm: A survey," *Int. J. Adv. Intell. Paradigms*, vol. 5, no. 1, pp. 123–159, Jun. 2013.

[24] M. H. Aghdam, N. Ghasem-Aghaee, and M. E. Basiri, "Text feature selection using ant colony optimization," *Expert Syst. Appl.*, vol. 36, no. 3, pp. 6843–6853, Apr. 2009.

[25] F. S. Abu-Mouti and M. E. El-Hawary, "Overview of artificial bee colony (ABC) algorithm and its applications," in *Proc. IEEE SysCon*, Mar. 2012, pp. 1–6.

[26] J. Luo, Q. Wang, and X. Xiao, "A modified artificial bee colony algorithm based on converge-onlookers approach for global optimization," *Appl. Math. Comput.*, vol. 219, no. 20, pp. 10253–10262, Jun. 2013.

[27] L. dos Santos Coelho and V. C. Mariani, "Use of chaotic sequences in a biologically inspired algorithm for engineering design optimization," *Expert Syst. Appl*, vol. 34, no. 3, pp. 1905–1913, Apr. 2008.

YUMEI ZHANG received the Ph.D. degree in control engineering from Northwestern Polytechnical University, Xi'an, China, in 2009. She is currently an Associate Professor at Shaanxi Normal University. Her research interests include signal processing and chaotic signal analysis.

YUYING RONG received the B.S. degree in software engineering from Shaanxi Normal University, Xi'an, China, in 2017, where she is currently pursuing the M.S. degree in computer science. Her research interests include nonlinear dynamics, time series analysis, and signal processing.
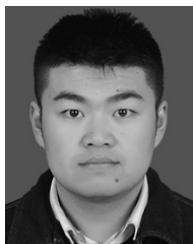
JIANGSHAN LIU received the B.S. degree in computer science and technology from the Xi'an University of Science and Technology, China, in 2016, and the M.S. degree in software engineering from Shaanxi Normal University in 2018. His research interests include nonlinear dynamics, time series analysis, and signal processing.

CHENGLING YAN is currently pursuing the B.S. degree in environmental science with Shandong University. Her research interests include nonlinear dynamics and data analysis.

XIAOJUN WU received the Ph.D. degree in system engineering from Northwestern Polytechnical University, Xi'an, China, in 2005. He is currently a Professor at Shaanxi Normal University. His research interests include pattern recognition, intelligent systems, and system complexity.

● ● ●