Extreme Learning Machine with a Modified Flower Pollination Algorithm for Filter Design

Li-Ye Xiao, Wei Shao, Sheng-Bing Shi, and Zhong-Bing Wang

School of Physics
University of Electronic Science and Technology of China, Chengdu, 610054, China
liyexiao@std.uestc.edu.cn, weishao@uestc.edu.cn, shengbing77@163.com, bzwang@uestc.edu.cn

Abstract ─ In this paper, a modified flower pollination algorithm (FPA) based on the steepest descent method (SDM) is proposed to set the optimal initial weights and thresholds of the extreme learning machine (ELM) for microwave filter design. With the proposed SDM-FPA, the model trained by the ELM can achieve higher accuracy with smaller training datasets for electromagnetic modeling, comparing to that achieved by traditional artificial neural network. The validity and efficiency of this proposed method is confirmed by a parametric modeling example of filter design.

Index Terms ─ Extreme learning machine (ELM), filter design, flower pollination algorithm (FPA), steepest descent method (SDM).

I. INTRODUCTION

In the design of microwave components or circuits, an optimization algorithm is often employed and it usually invokes the electromagnetic simulations repeatedly. The time-consuming full-wave simulations result in a heavy computational burden to complete the design. Fortunately, the artificial neural network (ANN) has been introduced to learn the relationship between geometrical variations and electromagnetic (EM) responses by a training process [1-3]. Once the geometrical parameters are imported into a trained ANN, it can fast obtain the accurate EM responses. An advanced study, which combines the neural network and transfer function (TF), was developed to model the EM behavior of embedded passive components [4]. In [4], the neural network is used for mapping the geometrical variations of the components onto the TF coefficients without having to rely on prior knowledge.

The most time-consuming part of ANN model construction is the collection of training and testing samples for model training and testing. How to reduce computation time and save more costs of model construction is a problem worth studying.

In the training process, the value of initial weights and thresholds is an important factor to determine the convergence of ANN. With the optimal initial weights and thresholds determined, the initial error is substantially smaller and therefore number of training samples that come from the time-consuming EM simulations to achieve the error criterion is reduced [5]. In general, the optimal initial weights and thresholds of ANN can be obtained through an optimization process.

In this paper, an effective ANN model, the extreme learning machine (ELM), is presented for filter design. By fixing input weights and hidden layer bias of a single-hidden layer feed-forward neural network (SLFN), ELM transforms the learning of SLFN into a matrix calculation, which largely improves the learning speed over the traditional feed forward network learning algorithms [6]. To further reduce computation time and save more training costs, a modified flower pollination algorithm (FPA) based on the steepest descent method (SDM) is developed to determine the optimal initial weights and thresholds of ELM. The SDM-FPA-ELM is used to learn the relationship between the geometrical variations of filters and the TF coefficients through the training process. Compared with the ANN model in [7], the proposed ELM can achieve the trained model with small training dataset and accurate results due to its good iterative learning ability. The validity of this proposed model is confirmed with the design of a quadruple-mode filter.

II. PROPOSED MODEL

A. Modified FPA: SDM-FPA

FPA has powerful global exploration and exploitation abilities, and its convergence speed in early period of optimization is fast [8]. However, during the late period of optimization, its convergence speed becomes slow and its accuracy is imprecise. To overcome the weakness and improve the optimized performance of FPA, a new modified FPA based on the steepest descent method (SDM) is developed in this work.

Let $\boldsymbol{x}^{iter}$ be the positions of flowers in FPA, where *iter* represents the current iteration number. Generally, $\boldsymbol{x}^{iter}$ is input into the fitness function directly to evaluate the current best value. To improve the local search ability of FPA, an SDM circulation is added in FPA. In

Submitted On: May 2, 2017
Accepted On: January 22, 2018

this circulation, $\boldsymbol{x}^{iter}$ will be the initial value to continue searching with the steepest descent direction $-\nabla f\left(\boldsymbol{x}_j^{iter}\right)$ and steplength λ_j^{iter}. This iterative loop could be presented as:

$$\boldsymbol{x}_{j+1}^{iter} = \boldsymbol{x}_j^{iter} + \lambda_j^{iter}\boldsymbol{d}_j^{iter}, \left(j = 0,1,\cdots,M-1\right), \quad (1)$$

where $\boldsymbol{d}_j^{iter} = -\nabla f\left(\boldsymbol{x}_j^{iter}\right)$, and M is the total number of SDM iterations.

After processing the SDM circulation, a new position of $\boldsymbol{x}_M^{iter}$ is obtained. $\boldsymbol{x}_M^{iter}$ is input into the fitness function to evaluate a value as the current best result. Then $\boldsymbol{x}_M^{iter}$ is updated to $\boldsymbol{x}_0^{iter+1}$ according to the FPA rules [8]. Meanwhile, to keep the results from trapping in local optimums, a lot of experiments have been done to select the total iteration number M. Finally we find that when M is set in the region from 4 to 6, the optimization performance is good. If M is less than 4, the rate of convergence may not be enhanced. And if M is greater than 6, the results are easily trapped in local optimums.

The statistics of optimal objectives for ten test functions [9] are tabulated in Table 1, and the best results are formatted in bold. It is clear in Table 1 that SDM-FPA works well and it is superior to the traditional FPA.

Table 1: Statistic of optimal objective values for the test functions

Test Function	Method	Min	Mean
Sphere function	FPA	7.0241×10^{-33}	8.2901×10^{-27}
	SDM-FPA	4.4521×10^{-63}	4.6425×10^{-61}
Beale function	FPA	6.4625×10^{-27}	9.6821×10^{-24}
	SDM-FPA	0	0
Griewank's function	FPA	6.3672×10^{-8}	6.5645×10^{-6}
	SDM-FPA	7.5485×10^{-67}	6.4654×10^{-64}
Matyas function	FPA	5.0086×10^{-42}	5.8247×10^{-28}
	SDM-FPA	2.5498×10^{-68}	5.0834×10^{-53}
Schaffer function	FPA	0	0
	SDM-FPA	0	0
Rastrigin's function	FPA	6.5643×10^{-16}	8.7436×10^{-10}
	SDM-FPA	0	2.8594×10^{-72}
Schwefel function	FPA	1.5549×10^{-6}	1.1032×10^{-5}
	SDM-FPA	1.5516×10^{-14}	1.5543×10^{-12}

B. Proposed ELM: SDM-FPA-ELM

ELM is a powerful and fast learning algorithm based on the modification of the traditional SLFNN [10]. It has been proved that the ELM network has better accuracy performance than the ANNs and the support vector machine [11].

Here, the network weights $\boldsymbol{\omega}$ between the input layer and the hidden layer are defined as:

$$\boldsymbol{\omega} = \begin{bmatrix} \omega_{11} & \omega_{12} & \omega_{13} & \cdots & \omega_{1n} \\ \omega_{21} & \omega_{22} & \omega_{23} & \cdots & \omega_{2n} \\ \vdots & \vdots & \vdots & & \vdots \\ \omega_{l1} & \omega_{l2} & \omega_{l3} & \cdots & \omega_{ln} \end{bmatrix}_{l\times n}, \quad (2)$$

where l is the number of the hidden neurons and n is the number of the input neurons. The network weights $\boldsymbol{\beta}$ between the hidden layer and the output layer are defined as:

$$\boldsymbol{\beta} = \begin{bmatrix} \beta_{11} & \beta_{12} & \beta_{13} & \cdots & \beta_{1m} \\ \beta_{21} & \beta_{22} & \beta_{23} & \cdots & \beta_{2m} \\ \vdots & \vdots & \vdots & & \vdots \\ \beta_{l1} & \beta_{l2} & \beta_{l3} & \cdots & \beta_{lm} \end{bmatrix}_{l\times m}, \quad (3)$$

where m is the number of the output neurons. And the thresholds $\boldsymbol{b}$ of the hidden layer are defined as:

$$\boldsymbol{b} = \left[b_1, b_2, \cdots, b_l\right]_{l\times 1}^T. \quad (4)$$

At the beginning of the training process, the initial values of $\boldsymbol{\omega}$, $\boldsymbol{\beta}$ and $\boldsymbol{b}$ are set randomly. However, the final results of ELM are strongly dependent on the initial weights and thresholds, and the bad initial weights and thresholds may lead to a slow convergence of the optimal value in ELM. To avoid wasting more training costs, in this paper, an optimization method is proposed to determine the initial weights and thresholds based on SDM-FPA.

There are two processes in SDM-FPA-ELM: the optimization of the values of $\boldsymbol{\omega}$, $\boldsymbol{\beta}$ and $\boldsymbol{b}$ and ELM training. The structure of SDM-FPA-ELM is shown in Fig. 1.

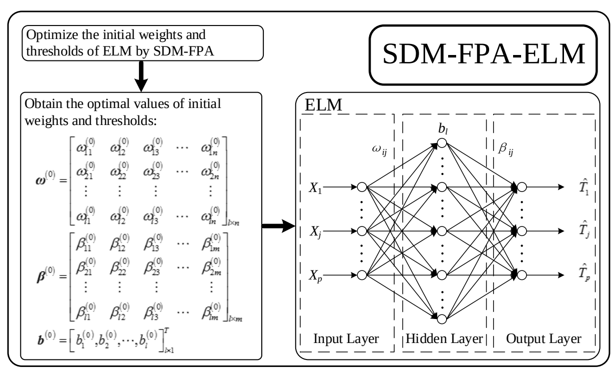

Fig. 1. Structure of SDM-FPA-ELM.

The fitness function in SDM-FPA is expressed as follows:

$$f\left(\boldsymbol{\omega}, \boldsymbol{\beta}, \boldsymbol{b}\right) = \min\left|\boldsymbol{T} - \hat{\boldsymbol{T}}\left(\boldsymbol{\omega}, \boldsymbol{\beta}, \boldsymbol{b}\right)\right|. \quad (5)$$

Here, the elements in $\boldsymbol{T}$ are the actual values of the real and imaginary parts of S-parameters:

$$\boldsymbol{T} = \begin{bmatrix} t_{11} & t_{12} & t_{13} & \cdots & t_{1P} \\ t_{21} & t_{22} & t_{23} & \cdots & t_{2P} \\ \vdots & \vdots & \vdots & & \vdots \\ t_{m1} & t_{m2} & t_{m3} & \cdots & t_{mP} \end{bmatrix}_{m\times P}, \tag{6}$$

and $\hat{\boldsymbol{T}}(\boldsymbol{\omega}, \boldsymbol{\beta}, \boldsymbol{b})$, which represents the outputs of the ELM network, is defined as:

$$\hat{\boldsymbol{T}}(\boldsymbol{\omega}, \boldsymbol{\beta}, \boldsymbol{b}) = \left[\hat{\boldsymbol{t}}_1(\boldsymbol{\omega}, \boldsymbol{\beta}, \boldsymbol{b}), \hat{\boldsymbol{t}}_2(\boldsymbol{\omega}, \boldsymbol{\beta}, \boldsymbol{b}), \cdots, \hat{\boldsymbol{t}}_P(\boldsymbol{\omega}, \boldsymbol{\beta}, \boldsymbol{b})\right], \tag{7}$$

where

$$\hat{\boldsymbol{t}}_j(\boldsymbol{\omega}, \boldsymbol{\beta}, \boldsymbol{b}) = \begin{bmatrix} \hat{t}_{1j} \\ \hat{t}_{2j} \\ \vdots \\ \hat{t}_{mj} \end{bmatrix}_{m\times 1} = \begin{bmatrix} \sum_{i=1}^{l} \beta_{i1} g(\boldsymbol{\omega}_i \boldsymbol{X}_j + b_i) \\ \sum_{i=1}^{l} \beta_{i2} g(\boldsymbol{\omega}_i \boldsymbol{X}_j + b_i) \\ \vdots \\ \sum_{i=1}^{l} \beta_{im} g(\boldsymbol{\omega}_i \boldsymbol{X}_j + b_i) \end{bmatrix}_{m\times 1}, \tag{8}$$

where

$\boldsymbol{X}_j = \left[X_{1j}, X_{2j}, X_{3j}, \cdots, X_{n-1j}, X_{nj}\right]^T, (j = 1, 2, 3, \cdots, P)$.

The elements in $\boldsymbol{X}$ are the inputs of ELM, such as the operation frequency, geometrical sizes and material parameters of microwave passive components.

When an iteration of SDM-FPA finishes, the current best values of $\boldsymbol{\omega}$, $\boldsymbol{\beta}$ and $\boldsymbol{b}$ are updated and substituted into ELM. After the input of $\boldsymbol{X}$ and output of $\boldsymbol{T}$ are applied to this ELM, the fitness function could be employed to evaluate this series of weights and thresholds.

This process will proceed till the objective tolerance of SDM-FPA is satisfied. And the obtained optimal values of weights and thresholds will be set as the initial ones of ELM, named as $\boldsymbol{\omega}^{(0)}$, $\boldsymbol{\beta}^{(0)}$ and $\boldsymbol{b}^{(0)}$.

The details of the next training process of ELM algorithm are shown in [10].

C. SDM-FPA-ELM-TF model

In this paper, the pole-residue-based transfer function, an effective TF form used in EM simulations, is chosen for our proposed model. The transfer function is presented as follows

$$H(s) = \sum_{i=1}^{N} \frac{r_i}{s - p_i}, \tag{9}$$

where p_i and r_i represent the poles and residues of the transfer function, respectively, and N is the function order [7].

The whole process of the SDM-FPA-ELM-TF model in Fig. 2 is as follows:

1) Collect the training and testing data with full-wave EM simulations.
2) Use SDM-FPA to optimize $\boldsymbol{\omega}$, $\boldsymbol{\beta}$ and $\boldsymbol{b}$, and set the initial values of weights and thresholds of ELM, named as $\boldsymbol{\omega}^{(0)}$, $\boldsymbol{\beta}^{(0)}$ and $\boldsymbol{b}^{(0)}$, with the optimal values.
3) Build the SDM-FPA-ELM-TF model for a passive component. The transfer function is used to represent the EM responses versus frequency. SDM-FPA-ELM is used to learn the relationship between the geometrical variations of the component and the coefficients of transfer functions through the training process.
4) Train the SDM-FPA-ELM-TF model with collected training data to find the optimal values of weights and thresholds of ELM.
5) Test the SDM-FPA-ELM-TF model with collected testing data.

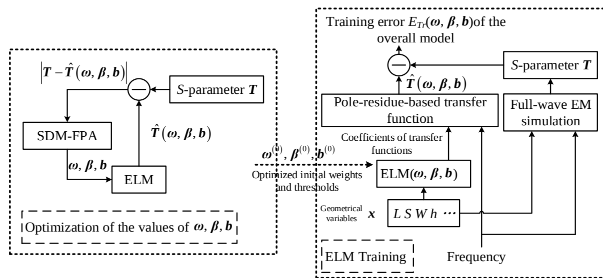

Fig. 2. SDM-FPA-ELM-TF model.

III. APPLICATION EXAMPLES

In this section, a quadruple-mode filter is used as an application example to evaluate the proposed SDM-FPA-ELM model. The *HFSS* 15.0 software performs the full-wave EM simulation and generates the training and testing data for modeling. The design of experiments (DOE) method is used for sampling [12]. All calculations in this paper are performed on an Intel i7-4870 2.50 GHz machine with 16 GB RAM.

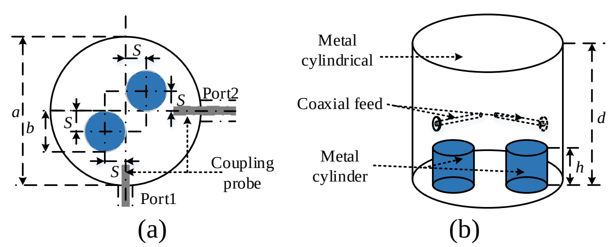

Fig. 3. Structure of the quadruple-mode filter. (a) Top view. (b) Three dimensional view.

The structure of a quadruple-mode filter is illustrated in Fig. 3, where the height and diameter of the cavity are d and a, the height and diameter of the two perturbation metal cylinders of the intra-cavity are h and b, and the distance between the two metal cylinders is $2\sqrt{2}S$ [13]. The model has five input geometrical variations, i.e., $\boldsymbol{x}$ = $[a\ b\ S\ h\ d]^T$. Frequency is an additional input parameter with an original range of 1-5

GHz. The model has two outputs, i.e., $\boldsymbol{y} = [R_{S11}\ I_{S11}]^T$, where R_{S11} is the real part of S_{11} and I_{S11} is the image part of S_{11}.

Table 2: Definition of training and testing data for the quadruple-mode filter

Geometrical Variations	Training Data (49 Samples)			Testing Data (25 Samples)		
	Min	Max	Step	Min	Max	Step
a (mm)	44	56	2	45	53	2
b (mm)	12	18	1	12.5	16.5	1
S (mm)	2	8	1	2.5	6.5	1
h (mm)	14	20	1	14.5	18.5	1
d (mm)	44	56	2	45	53	2

The DOE method with seven levels defines the training samples, i.e., a total of 49 training samples, and DOE with five levels defines the testing data, i.e., a total of 25 testing samples. The information of training data and testing data is shown in Table 2. The total CPU time for training-data generation from EM simulations is about 1.225 hours, and the total time for testing-data generation is about 0.625 hours. After the modeling process, the average training percentage error is 0.385%, while the average testing percentage error is 0.587%.

Table 3: Comparison between the reference model and proposed model for the quadruple-mode filter

49 Training Samples and 25 Testing Samples	Average Training Error	Average Testing Error
Reference model in [7]	2.904%	4.862%
Proposed model	0.385%	0.587%
81 Training Samples and 49 Testing Samples	Average Training Error	Average Testing Error
Reference model in [7]	1.431%	1.509%
Proposed model	0.407%	0.598%

The model in [6] is employed for comparison here. The DOE methods with nine levels (81 samples) and seven levels (49 samples) are respectively used for training and testing in the reference model. The total time for training-data generation from EM simulations is about 2 hours, and the total time for testing-data generation is about 1.225 hours. With 49 training samples, as shown in Table 3, the proposed model achieves the acceptable accuracy, but the desired accuracy cannot be obtained with the reference model. When the number of training sample rises to 81, the accuracy of reference model is enhanced. It is observed that fewer training samples are needed for the proposed model to obtain an accurate trained model. It means that, to achieve the same accuracy, considerable simulation time could be saved with the proposed model for sample collection, as illustrated in Table 4.

Table 4: Running time of the two models for the quadruple-mode filter

	CPU Time of Model Development	
	Reference Model in [7]	Proposed Model
Training process	4 hours (81 samples)	2.45 hours (49 samples)
Testing process	2.45 hours (49 samples)	1.25 hours (25 samples)
Total	6.45 hours	3.7 hours

Figure 4 shows the outputs of two different test geometrical samples of the quadruple-mode filter with the proposed model, comparison model and HFSS simulation. The geometrical variables for the two samples in the range of the training data are $\boldsymbol{x}_1$ = [45.6 15.6 7.2 17.1 48.1]^T and $\boldsymbol{x}_2$ = [52.3 11.8 6.3 19.4 45.8]^T. It is observed that the proposed model can achieve good accuracy for different geometrical samples even though these samples are never used in training.

Meanwhile, two other test geometrical samples, which are selected out of the range of the training data, are chosen to evaluate the proposed model. The geometrical variations for the two samples are $\boldsymbol{x}_1^{'}$ = [40 10 0 12 40]^T and $\boldsymbol{x}_2^{'}$ = [40 9 10 22 60]^T. From Fig. 5, it is observed that our model can achieve good accuracy for different geometrical samples even though these samples are out of the range of the training data.

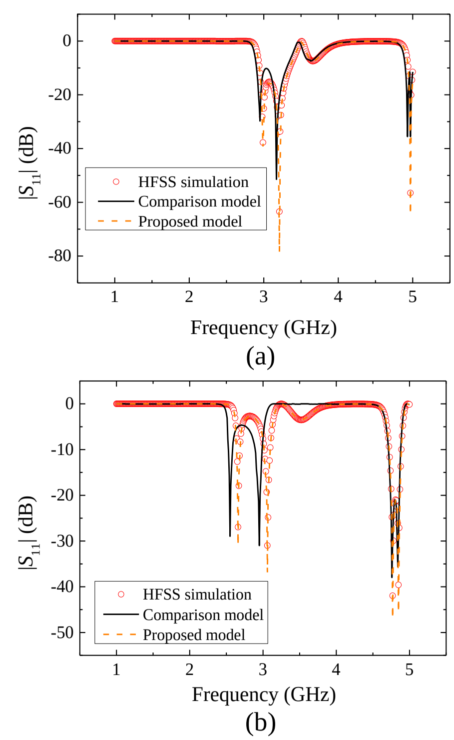

Fig. 4. Comparison of S_{11}: (a) Sample 1 and (b) Sample 2, where the samples are in the range of training data.

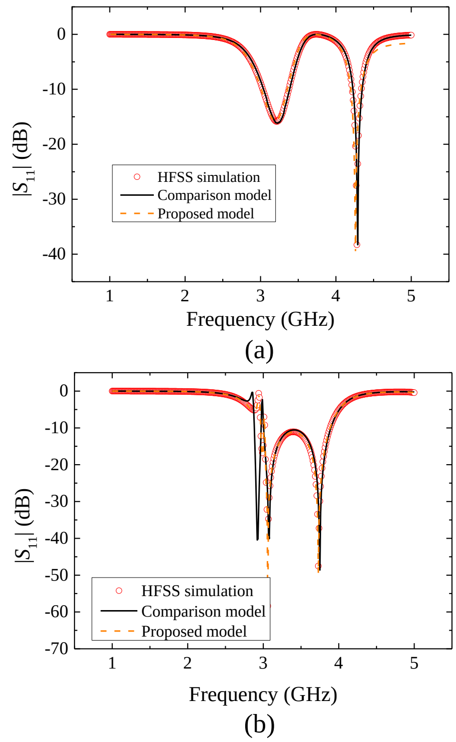

Fig. 5. Comparison of S_{11}: (a) Sample 1 and (b) Sample 2, where the samples are out of the range of training data.

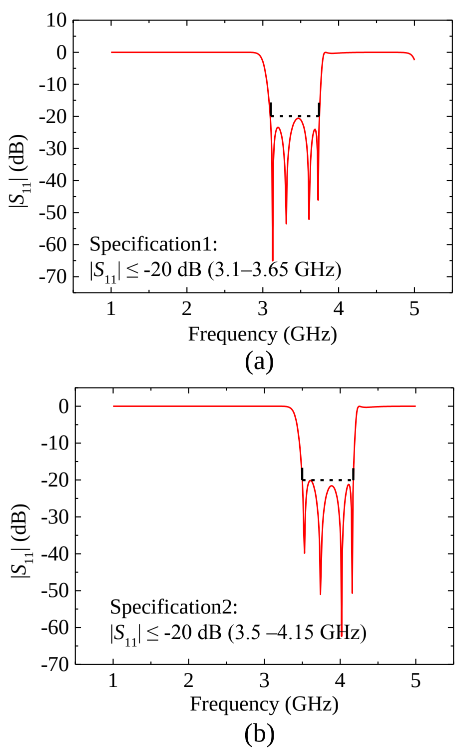

Fig. 6. Optimization results of S_{11} from the proposed model: (a) Filter 1 and (b) Filter 2.

Once the proposed model training is completed, the trained model which is a substitute for the time-consuming EM simulation can be applied to the design optimization. As an example of filter design, two separate filters are optimized to reach two different design specifications. The optimization objectives and results are shown in Fig. 6. The initial values are $\boldsymbol{x}$ = [50 15 4 17 50]^T. The optimization with SDM-FPA of the quadruple-mode filter is performed by calling the trained model repeatedly. The optimized geometrical values for the two separate filters are: $\boldsymbol{x}_{\text{opt1}}$ = [50.0234 14.4081 7.4103 14.8103 44.9081]^T and $\boldsymbol{x}_{\text{opt2}}$ = [49.9802 13.9841 7.2004 14.7903 45.3094]^T.

The optimization spends only about 30 seconds to achieve the optimal solution for each filter, shown in Table 5. The proposed model could save considerable time in optimization design compared with the calling of EM simulations.

Table 5: Running time of the direct EM optimization and the proposed model

	CPU Time of Model Development	
	Direct EM Optimization	SDM-FPA-ELM-TF Model
Filter 1	10 hours	30 s
Filter 2	11 hours	30 s
Total	21 hours	3.7 hours (training)+60 s

IV. CONCLUSION

In this paper, an efficient ELM model based on a modified SDM-FPA is proposed to enhance the learning effectiveness in EM simulations. SDM-FPA is developed to set the optimal initial weights and thresholds of ELM and it saves more training costs. In this method, SDM-FPA-ELM is used to learn the relationship between the geometrical variations of filters and the coefficients of transfer functions through the training process. Compared with the reference model, the proposed model can achieve the trained model with small training samples due to its good iteration learning ability. A quadruple-mode filter is employed as a parametric modeling example to confirm its validity. The proposed model provides its powerful computing ability, especially in the field of EM optimization design.

ACKNOWLEDGMENT

This work was supported by the National Natural Science Foundation of China (No. 61471105) and 973 Project (No. 613273).

REFERENCES

[1] J. E. Rayas-Sánchez, "EM-based optimization of microwave circuits using artificial neural networks: The state-of-the-art," *IEEE Trans. Microw. Theory Techn.*, vol. 52, no. 1, pp. 420-435, Jan. 2004.

[2] V. Rizzoli, A. Costanzo, D. Masotti, A. Lipparini,

and F. Mastri, “Computer-aided optimization of nonlinear microwave circuits with the aid of electromagnetic simulation,” *IEEE Trans. Microw. Theory Techn*., vol. 52, no. 1, pp. 362-377, Jan. 2004.

[3] X. Ding, V. K. Devabhaktuni, B. Chattaraj, M. C. E. Yagoub, M. Deo, J. Xu, and Q. J. Zhang, “Neural-network approaches to electromagnetic based modeling of passive components and their applications to highfrequency and high-speed nonlinear circuit optimization,” *IEEE Trans. Microw. Theory Techn*., vol. 52, no. 1, pp. 436-449, Jan. 2004.

[4] Q. J. Zhang, K. C. Gupta, and V. K. Devabhaktuni, “Artificial neural networks for RF and microwave design—From theory to practice,” *IEEE Trans. Microw. Theory Techn.*, vol. 51, no. 4, pp. 1339-1350, Apr. 2003.

[5] Y. F. Yam, T. W. S. Chow, and C. T. Leung, “A new method in determining initial weights of feed forward neural networks for training enhancement,” *Neurocomputing*, vol. 16, pp. 23-32, 1997.

[6] G. B. Huang, Q. Y. Zhu, and C. K. Siew, “Extreme learning machine: Theory and applications,” *Neurocomputing*, vol. 70, no. 1-3, pp. 489-501, 2006.

[7] F. Feng, C. Zhang, J. Ma, and Q. J. Zhang, “Parametric modeling of EM behavior of micro-wave components using combined neural networks and pole-residue-based transfer functions,” *IEEE Trans. Microw Theory Techn.*, vol. 64, no. 1, pp. 60-77, Jan. 2016.

[8] X. S. Yang, “Flower pollination algorithm for global optimization,” in *Proc. of the 11th Int'l. Conf. on Unconventional Computation and Natural Computation (UCNC)*, Orléans, France, pp. 242-243, Sept. 2012.

[9] E. Nabil, “A modified flower pollination algorithm for global optimization,” *Expert. Syst. Appl.*, vol. 57, pp. 192-203, 2016.

[10] G. B. Huang, Q. Y. Zhu, and C. K. Siew, “Extreme learning machine: Theory and applications,” *Neurocomputing*, vol. 70, no. 1-3, pp. 489-501, 2006.

[11] X. Shi, J. Wang, G. Liu, L. Yang, X. Ge, and S. Jiang, “Application of extreme learning machine and neural networks in total organic carbon content prediction in organic shale with wire line logs,” *J. Nat. Gas. Sci. Eng.*, vol. 33, pp. 687-702, 2016.

[12] S. R. Schmidt and R. G. Launsby, “Understanding Industrial Designed Experiments,” Colorado Springs, CO, USA, Air Force Academy, 1992.

[13] S. W. Wong, S. F. Feng, L. Zhu, and Q. X. Chu, “Triple- and quadruple-mode wideband bandpass filter using simple perturbation in single metal cavity,” *IEEE Trans. Microw Theory Techn.*, vol. 63, no. 10, pp. 1-9, Oct. 2015.

Li-Ye Xiao received the B.S. degree in Electronic Information Science and Technology from the University of Electronic Science and Technology of China (UESTC), Chengdu, China, in 2015. Currently, he is working toward the Ph.D. degree in Radio Physics at UESTC.

His research interest is neural network, optimization algorithm and computational electromagnetics.

Wei Shao received the B.E. degree in Electrical Engineering from UESTC in 1998, and received M.Sc. and Ph.D. degrees in Radio Physics from UESTC in 2004 and 2006, respectively.

He joined the UESTC in 2007 and is now a Professor there. From 2010 to 2011, he was a Visiting Scholar in the Electromagnetic Communication Laboratory, Pennsylvania State University, State College, PA. From 2012 to 2013, he was a Visiting Scholar in the Department of Electrical and Electronic Engineering, Hong Kong University. His research interests include computational electromagnetics and antenna design.

Sheng-Bing Shi received the B.S. degree in Physics from the Yangtze University, Jingzhou, China, in 2013. Currently, he is working toward the Ph.D. degree in Radio Physics at UESTC.

His research interest is computational electromagnetics.

Bing-Zhong Wang received the Ph.D. degree in Electrical Engineering from UESTC in 1988.

He joined the UESTC in 1984 where he is currently a Professor. He has been a Visiting Scholar at the University of Wisconsin-Milwaukee, a Research Fellow at the City University of Hong Kong, and a Visiting Professor in the Electromagnetic Communication Laboratory, Pennsylvania State University, State College, PA. His current research interests are computational electromagnetics and antenna theory.