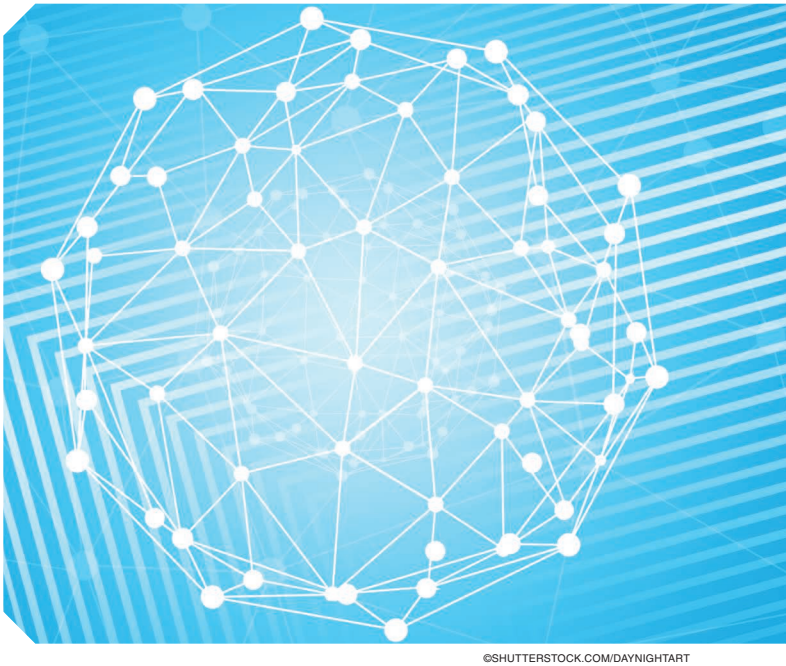
Automated Neural-Network-Based Model Generation Algorithms for Microwave Applications

Weicong Na, Taiqi Bai, Ke Liu, Wanrong Zhang, and Qi-Jun Zhang



©SHUTTERSTOCK.COM/DAYNIGHTART

Artificial neural networks (ANNs) are playing a role in the electronic design automation (EDA) area, especially for microwave modeling and optimization [1], [2], [3]. Their ability to capture complex relationships within the electromagnetic (EM) domain contributes to its effectiveness in simulating and predicting the behavior of microwave devices and circuits, offering insights that are both cost-effective and time-efficient. The wide-ranging applications of ANNs in

Weicong Na (weicongna@bjut.edu.cn) and Taiqi Bai (btaiqi@emails.bjut.edu.cn) are with the Faculty of Information Technology, Beijing University of Technology, Beijing 100124, China. Ke Liu (keliu@tju.edu.cn) is with the School of Microelectronics, Tianjin University, Tianjin 300072, China. Wanrong Zhang (wrzhang@bjut.edu.cn) is with the Faculty of Information Technology, Beijing University of Technology, Beijing 100124, China. Qi-Jun Zhang (qjz@doe.carleton.ca) is with the Department of Electronics, Carleton University, Ottawa, ON K1S5B6, Canada.

Digital Object Identifier 10.1109/MMM.2024.3486596
Date of current version: 13 January 2025

EM modeling and optimization for microwave devices/circuits are well established in areas such as very large-scale integration interconnection [4], [5], [6], microwave filters [7], [8], [9], [10], antennas [11], [12], [13], [14], metasurfaces [15], [16], [17], [18], microwave amplifiers [19], [20], [21], [22], wireless power transfer [23], and multiphysics design [24], [25], [26], [27].

Introduction

The development of ANN models for microwave devices/circuits entails a series of interrelated subtasks, including data sampling and generation, adaptive model structure selection, precise model training, and rigorous real-world testing. This process demands substantial human effort and comprehensive skills encompassing EM principles, microwave technology, and machine learning considerations. To minimize the manual efforts needed and promote broader adoption of ANN technology in microwave applications, the automated model generation (AMG) algorithm [28] has emerged as a popular technique for systematically developing ANN models for microwave devices/circuits. AMG integrates all the subtasks in ANN development into a unified automated algorithm. The assessments of the ANN training phenomena related to underlearning, overlearning, and good learning are automated, and the quantitative links among the accuracy of the ANN model, the amount/distribution of training/testing data, and the size of the neural network are established. In this way, the AMG algorithm automatically creates an ANN model with user-desired accuracy, significantly reducing the human time required for modeling, and without requiring users to understand neural network issues.

In AMG, two key problems of ANN model development need to be addressed, i.e., determining the suitable amount/distribution of the data samples and determining the suitable structure of the ANN model. In recent years, with the development of ANN theory, the AMG algorithm has made significant progress in both aspects. AMG incorporated with efficient interpolation technique [29], [30] is one of the advances in dynamic data sampling. To avoid training intermediate ANN models in every sampling stage, local interpolation models of different subregions in the modeling range are created to assess the adequacy of training data and determine the distribution of training/testing data. The interpolation technique automatically distinguishes highly nonlinear subregions from smooth subregions, and then additional data are generated extensively in the nonlinear subregions, while only a limited number of data are generated in the smooth subregions. ANN training is performed only after the completion of the data sampling process, resulting in a significant reduction in the time cost of data sampling compared with the original AMG algorithm.

Much work has been done in ANN model structure adjustment of AMG to determine the appropriate neural model structure for different microwave applications in a faster way [31], [32], [33]. Bayesian-assisted AMG algorithms [34], [35] and AMG algorithms incorporating l_1 regularization [36], [37] have been proposed for the automated determination of the optimal multilayer ANN structure, simultaneously adjusting the number of hidden layers and the number of hidden neurons in each layer. Bayesian-assisted AMG algorithms dynamically adapt the ANN structure by recognizing the correlation between the number of effective weight parameters and the number of hidden layers/neurons in the model, thus producing the most compact ANN model with the highest accuracy. AMG algorithms employing l_1 regularization take advantage of the distinctive feature selection property during the ANN training process to automatically determine the optimal neural model structure, improving the ANN model development efficiency. In addition, to further improve the accuracy and reliability of ANN modeling and design optimization, the scope of the AMG algorithm has also been extended beyond the generation of pure ANN models to the generation of knowledge-based neural network (KBNN) models [29], [38], [39]. KBNN combines ANNs with prior knowledge, such as analytical formulas, equivalent circuits, and empirical models, to construct comprehensive models. The ANNs within the KBNN model are mapping ANNs that serve to refine the behavior of the prior knowledge to better match the specific microwave applications. In [38], an elegant and unified AMG algorithm utilizing l_1 optimization is presented to automatically identify the optimal type and topology of the mapping structure in a KBNN for different microwave modeling problems. The final KBNN structure may encompass flexible combinations of linear mapping, nonlinear mapping, input mapping, frequency mapping, and output mapping. Consequently, KBNN model development can be conducted in a more systematic and faster manner compared with manual KBNN modeling methods.

In this article, an overview of AMG algorithms for microwave applications is presented. The fundamental concepts of AMG are described in the "Fundamental Concepts of AMG Algorithm" section. Recent advanced AMG algorithms for microwave modeling are reviewed in the "AMG Incorporated With Interpolation Technique" to "AMG of KBNN" sections, and several microwave application examples using AMG algorithms are highlighted in the "Examples of AMG for Microwave Modeling" section. The "Conclusion"

section discusses future directions of AMG algorithms in microwave EDA.

Fundamental Concepts of AMG Algorithm

For microwave parametric modeling, let $\boldsymbol{x} = [x_1, x_2, \ldots x_n]^T$ represent the vector containing n geometrical or physical parameters of a microwave device/circuit. Let $\boldsymbol{d} = [d_1, d_2, \ldots d_m]^T$ represent the vector containing m EM responses of the microwave device/circuit. The goal of AMG is to automatically develop an ANN that represents the nonlinear and multidimensional relationship between $\boldsymbol{x}$ and $\boldsymbol{d}$, achieving the accuracy specified by the user [28]. The input—output relationship of the ANN is defined as

$$\boldsymbol{y} = f_{\text{ANN}}(\boldsymbol{x}, \boldsymbol{w}) \tag{1}$$

where $\boldsymbol{y} = [y_1, y_2, \ldots y_m]^T$ represents the vector containing m ANN outputs and $\boldsymbol{w}$ represents the trainable weights within the ANN. Let $(\boldsymbol{x}, \boldsymbol{d})$ represent one training/testing data point. Let E_{train} and E_{test} represent the training error and the testing error of the ANN, respectively, which are both mean squared errors. Let E_d represent the user-desired model accuracy (testing error). Starting with a zero amount of training/testing data and a relatively small ANN size, the AMG process of an ANN model is carried out stage by stage. In each stage, the mutual dependence and correlation among different subtasks of ANN modeling like data sampling/generation, ANN model size, and ANN training and testing are identified, and then either dynamic incremental data generation or ANN structure adjustment is performed based on different ANN learning

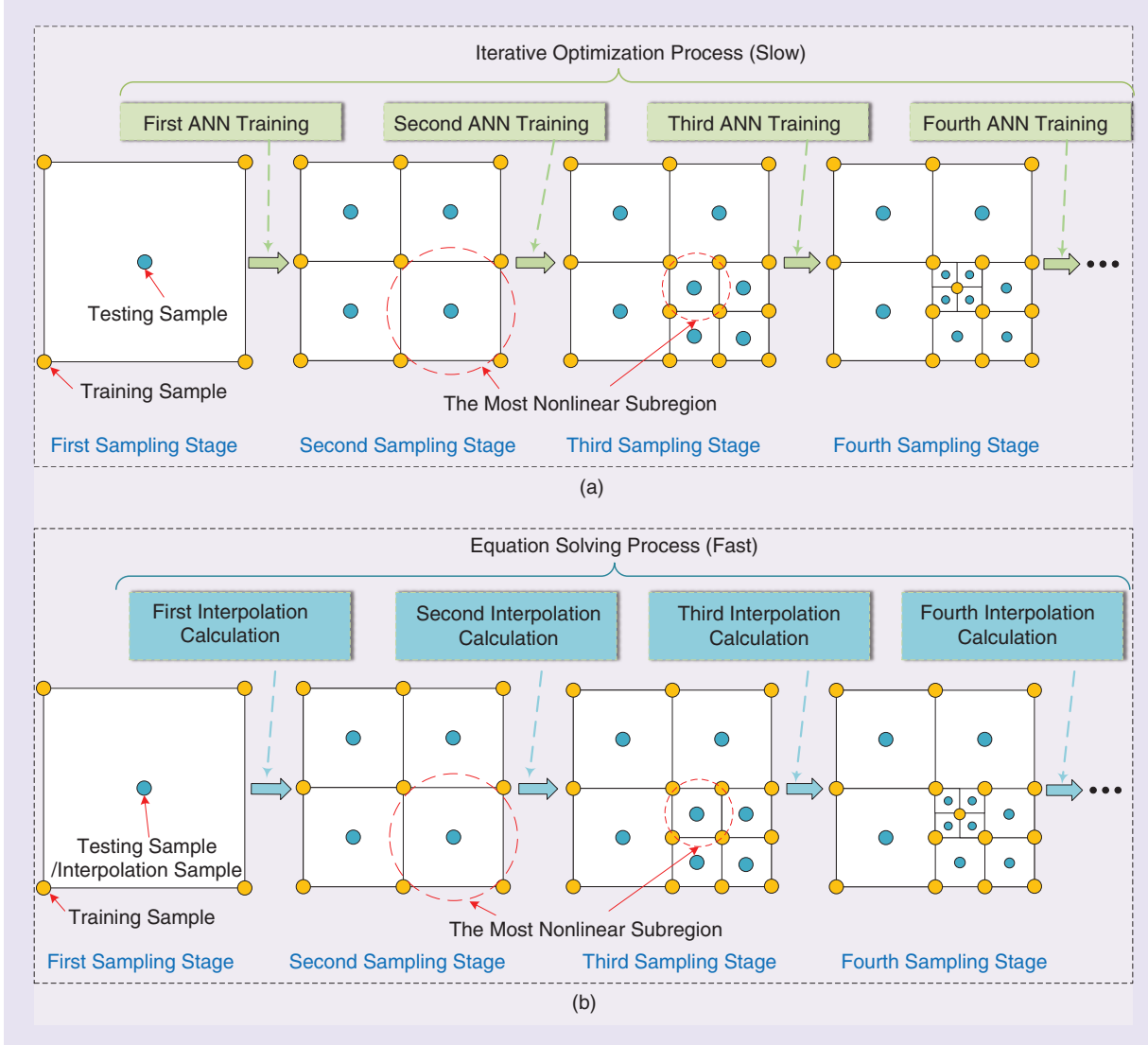


Figure 1. *The sampling distribution in each sampling stage by (a) the original AMG algorithm and (b) the AMG algorithm with interpolation for a two-dimensional example.*

criteria (e.g., overlearning, underlearning, or good learning) observed in each stage.

Specifically, in the first stage of AMG, the original bounded n-dimensional input space of interest is regarded as one region. The initial training and testing data for this region are generated as shown in Figure 1(a) by automatically driving the EM simulator for data generation. Then the obtained training data are used to train an ANN with a single hidden layer and relatively fewer hidden neurons, and the resulting ANN is tested with testing data. The AMG algorithm stops if $E_{\text{test}} \le E_d$. Otherwise, the algorithm automatically advances to the next stage and takes suitable actions according to different ANN error criteria.

1) If $E_{\text{train}} > E_d$, the resulting ANN is underlearning, and the next stage is for ANN structure adjustment. The AMG algorithm automatically adds more hidden neurons to provide increased freedom for the ANN structure to better learn the nonlinearities in training data.
2) If $E_{\text{train}} \le E_d$ but $E_{\text{test}} > E_d$, the resulting ANN is overlearning, and the next stage is for dynamic data generation. The AMG algorithm adaptively adds additional training data and testing data to improve the generality of the ANN model. The pattern of dynamic sampling in AMG is shown in Figure 1(a). In each sampling stage, the subregion with the maximum testing error is identified and split into 2^n new subregions by adding new training data. The final sampling distribution of AMG is an efficient distribution with more data in highly nonlinear subregions and fewer data in relatively smooth subregions.

The systematic AMG framework is illustrated in Figure 2. Within the AMG framework, all steps involved in ANN model development for microwave devices/circuits are automated by the computer, eliminating the need for human intervention. This results in a substantial acceleration of the ANN model development process when compared with the conventional manual modeling method.

AMG Incorporated With Interpolation Technique

In the original AMG algorithm [28], ANN training is performed stage by stage to determine whether to add training data or hidden neurons. To avoid training intermediate ANNs in data sampling stages, an interpolation technique is incorporated into the AMG process [30]. We use simpler interpolation models in local subregions to replace the ANN model throughout the entire region, assessing the adequacy of training data. The interpolation function for a subregion is formulated as follows:

$$g(\boldsymbol{x}) = p_1 + \sum_{i=1}^{n} p_{i+1} x_i + p_{n+2} x_1 x_2 + p_{n+3} x_1 x_3 + \cdots + p_K x_1 x_2 \ldots x_n + \sum_{i=1}^{n} p_{K+i} x_i^2 = \boldsymbol{p}^{\mathrm{T}} \cdot \boldsymbol{h} \quad (2)$$

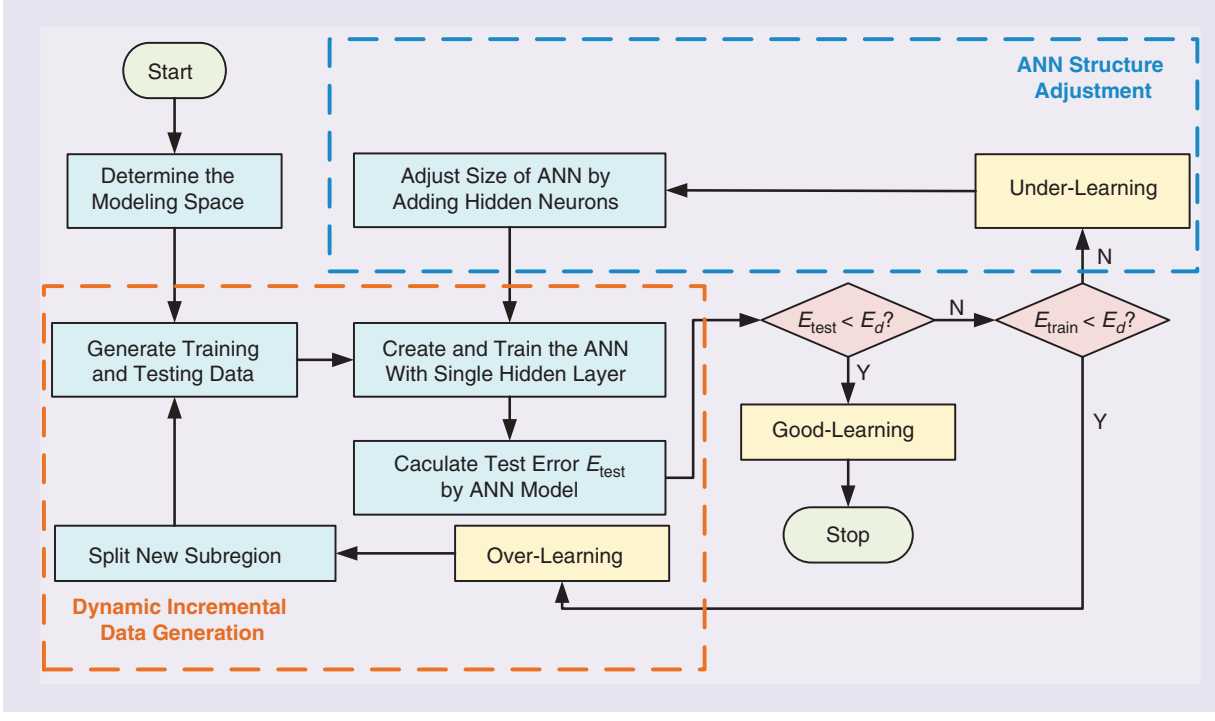


Figure 2. *The systematic framework of the AMG algorithm [28].*

where $\boldsymbol{p} = [p_1, p_2, \ldots, p_N]^T$ denotes a vector with N coefficients, and $\boldsymbol{h}$ denotes a vector containing the products of various combinations of $x_1, x_2, \ldots$ and x_n. We utilize the training data in and around the interpolated subregion to calculate the unknown coefficients $\boldsymbol{p}$ in the interpolation function $g(\boldsymbol{x})$. Once $g(\boldsymbol{x})$ functions for all subregions are obtained, we evaluate the testing error of each subregion by comparing the desired output value and the interpolation output at the testing point. The subregion with the maximum testing error will be further split in the next stage by adding new training data, as shown in Figure 1(b). The dynamic data sampling process stops once the testing error meets the user-required modeling accuracy. Subsequently, the training of an ANN is executed solely after the sampling process for structure adaptation. The resulting model is an ANN model valid across the entire region. The flowchart of the AMG algorithm incorporating interpolation is illustrated in Figure 3. By producing localized interpolation models to replace intermediate ANN training during dynamic sampling, the entire AMG process of the ANN model is further sped up.

AMG With Efficient Multilayer ANN Structure Adjustment Algorithm

In AMG, in addition to dynamic sampling, the ANN structure adjustment is also an important aspect that affects the modeling efficiency. As microwave modeling problems become increasingly complicated, the shallow ANN with one or two hidden layers is not sufficient to learn the high-dimensional and highly nonlinear input–output relationship of these challenging modeling problems [40]. The multilayer ANN is more flexible and has better performance than the shallow ANN when both have the same number of weights. Subsequently, how to simultaneously adjust the number of hidden layers and hidden neurons in an ANN model to cater to diverse requirements in various microwave modeling problems has become an important issue in the AMG process, which has a significant impact on the final model accuracy.

Recently, an efficient multilayer ANN structure adjustment algorithm [37] for AMG has been presented to address this problem. This AMG algorithm introduces a novel cascaded multilayer ANN structure and an l_1 regularization-based training algorithm. It facilitates the automatic adjustment of the multilayer ANN structure by selectively adding or removing the hidden layers/neurons in batch, thus making a balance between the number of hidden layers and the number of hidden neurons.

The cascaded multilayer ANN structure is an extension of the fully connected multilayer neural network by adding direct connections between the output layer

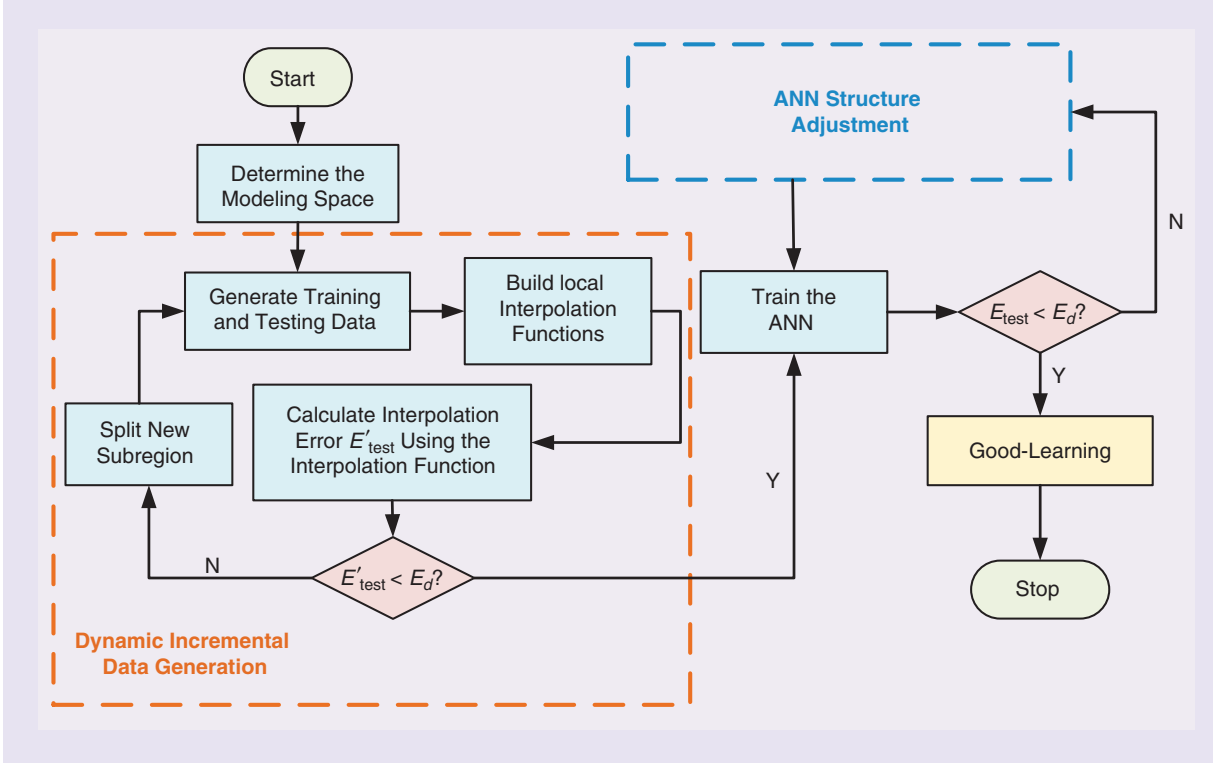


Figure 3. *The flowchart of the AMG algorithm incorporating interpolation [30], where E'_{test} is calculated by comparing the desired output value and the interpolation output at the testing point.*

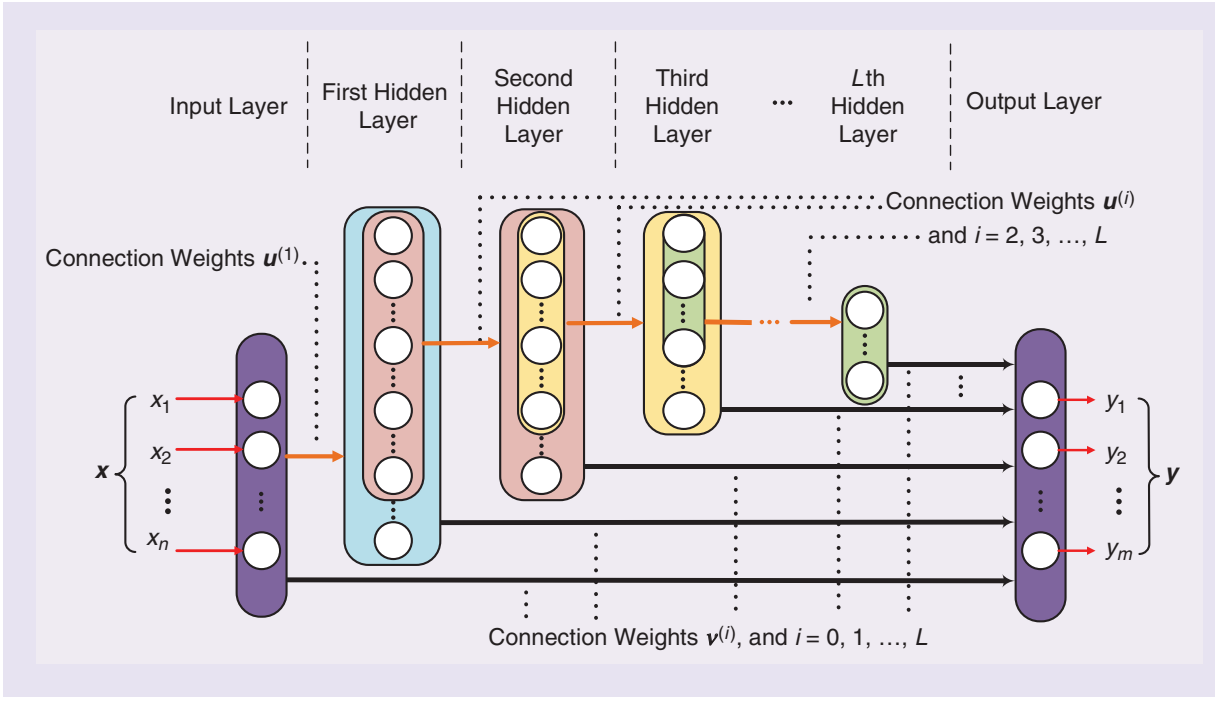


Figure 4. *The cascaded structure of the multilayer ANN model [37].*

and the input layer, as well as each hidden layer, as shown in Figure 4. In this multilayer ANN structure, L and $H_i\,(1 \le i \le L)$ are defined as the number of hidden layers and the number of hidden neurons in the ith hidden layer, respectively. The initial values of L and H_1 can be any positive integer set by the user, and $H_i\,(2 \le i \le L)$ is calculated as

$$H_i = \begin{cases} \left\lfloor H_{i-1}(m+n+1) + \left(\frac{m+n}{2}+1\right)^2 \right\rfloor^{\frac{1}{2}} - \left(\frac{m+n}{2}+1\right) & \text{when } i = 2 \\ \left\lfloor H_{i-1}(m+H_{i-1}+1) + \left(\frac{m+H_{i-1}}{2}+1\right)^2 \right\rfloor^{\frac{1}{2}} - \left(\frac{m+H_{i-1}}{2}+1\right) & \text{when } i > 2. \end{cases} \tag{3}$$

The input layer of the cascaded multilayer ANN is fully connected to the first hidden layer and the output layer, while each hidden layer is fully connected to the output layer but partially connect to the next hidden layer [i.e., H_{i+1} neurons in the ith hidden layer are connected to the $(i+1)$th hidden layer]. All the connection weights between two neighboring layers are defined as the vector $\boldsymbol{u} = [\boldsymbol{u}^{(1)}, \boldsymbol{u}^{(2)}, \ldots, \boldsymbol{u}^{(L)}]^{\mathrm{T}}$, where $\boldsymbol{u}^{(i)}$ represents the connection weights between the ith layer and the $(i+1)$th layer, and $\boldsymbol{v}$ denotes a vector containing all the trainable weights between each input/hidden layer and the output layer. To automatically adjust the model structure, the cascade ANN uses the tanh function rather than the sigmoid function as the activation function. Since a neuron with a tanh activation function may produce zero output after processing its input, we use this feature to distinguish whether a neuron is activated or not after training. If $\boldsymbol{u}^{(i)} = 0$, all neurons in the $(i+1)$th layer are not activated, which means that all hidden layers above the ith layer in the cascaded multilayer ANN structure can be deleted in a batch. And because the output layer is connected to every hidden layer of the ANN, it is still a complete ANN structure after deleting these internal hidden layers.

To adaptively determine the final model structure, a stage-by-stage training scheme with l_1 regularization is introduced to train the cascaded multilayer ANN. In each stage, l_1 regularization is performed using the following error function, i.e.,

$$E_{\text{train}}^{(1)} = \sum_{j=1}^{N_d} \left(\frac{1}{2} \| \boldsymbol{y}_j - \boldsymbol{d}_j \|^2 \right) + \sum_{i=1}^{L} \sum_{p=1}^{H_i} \sum_{q=0}^{H_i'} \lambda^{(i)} | u_{p,q}^{(i)} | \tag{4}$$

where N_d represents the total number of training samples, $u_{p,q}^{(i)}$ represents the weight between the qth neuron in the $(i-1)$th hidden layer and the pth neuron in the ith hidden layer, and $\lambda^{(i)}$ represents the

nonnegative coefficient to penalize any large values of $u_{p,q}^{(i)}$. $H_i' = H_{i-1}$ when $i > 1$, and $H_i' = n$ when $i = 1$.

Due to the feature selection properties of l_1 norms [41], some large weights in $\boldsymbol{u}$ can be ignored while other weights are forced to zeros after l_1 regularization, which means the important weights in $\boldsymbol{u}$ are automatically selected by l_1 regularization. For any hidden neuron with tanh activation function in the ith hidden layer, if all weights between this hidden neuron and all hidden neurons in the (i-1)th hidden layer are equal to zero, this hidden neuron is an inactivated neuron with zero output. If all the hidden neurons in the ith hidden layer are inactivated neurons, the ith hidden layer and all hidden layers above it are useless and can be deleted in a batch. In this way, benefiting from the tanh activation function and the direction connections between the output and each layer, the activation situations of hidden neurons/layers and the training status of the cascaded multilayer ANN can be automatically detected after l_1 regularization in each stage, then the training algorithm takes distinct actions to modify the model structure in the following stage. The AMG with efficient multilayer ANN structure adjustment algorithm is summarized in Figure 5. With the flexible multilayer ANN structure and l_1 regularization-based training algorithm, the presented AMG algorithm balances the number of neurons and the number of layers and automatically produces an ANN model with optimal structure to meet the microwave modeling accuracy requirement in the shortest possible time.

AMG of KBNN

In KBNN, the prior knowledge (e.g., empirical formulas, equivalent circuits, etc.) from the microwave field is combined with the ANNs to reduce the amount of required training/testing data for modeling and enhance the model generalization performance [2]. The ANNs in KBNN are used to bridge the difference between the response of the prior knowledge and that of the EM training data, which can be the input mapping ANN, the frequency mapping ANN, and the output mapping ANN. For different microwave modeling problems, the number and topology of the mapping ANNs in the KBNN model can be different, and each mapping ANN can be either linear mapping or nonlinear mapping. Since there is no clear quantitative relationship between the model accuracy and the mapping structure, the AMG algorithm for KBNN is introduced [38] to address the problem of automatically generating a KBNN model with appropriate mapping types and topological structures to meet the user's expectation for model accuracy in different modeling problems.

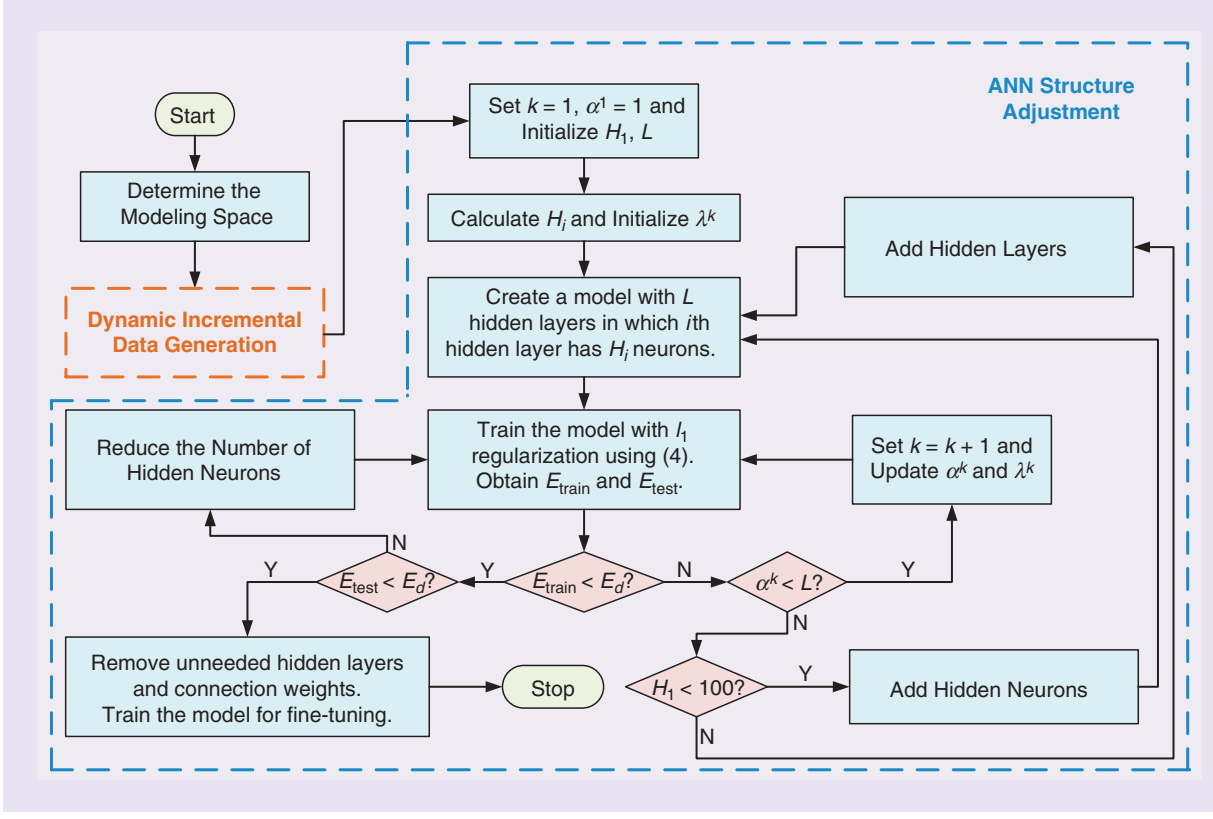


Figure 5. *The AMG with efficient multilayer ANN structure adjustment algorithm [37], where α^k is the number of activated hidden layers in the kth stage and $\boldsymbol{\lambda}^k = [\lambda^{(1)}, \lambda^{(2)}, \ldots, \lambda^{(L)}]^T$.*

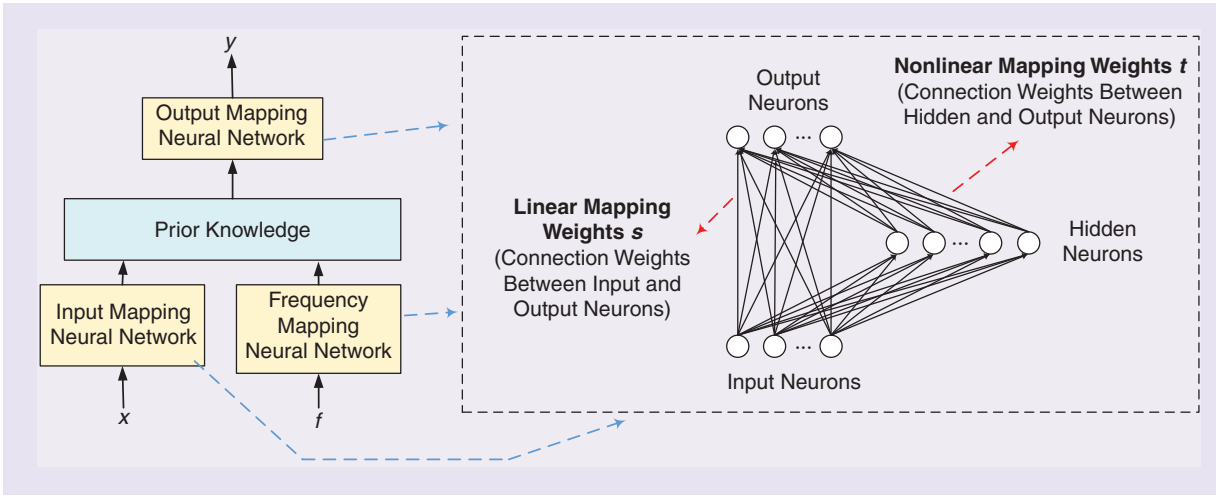


Figure 6. *The general structure of the KBNN with AMG [38].*

The general structure of the KBNN in AMG is shown in Figure 6. Various mapping ANNs are superimposed on the prior knowledge to include all cases of mappings, and each ANN mapping includes linear and nonlinear mapping simultaneously by adding direction connections between the input neurons and output neurons. This general KBNN structure can be easily converted to KBNNs with different mapping structures after a two-stage training with l_1 optimization, which is illustrated in Figure 7. In the first stage, the error function containing l_1 regular terms is used for overall KBNN training, i.e.,

$$E_{\text{train}}^{(2)} = \sum_{j=1}^{N_d}\left(\frac{1}{2}\|\boldsymbol{y}_j - \boldsymbol{d}_j\|^2\right) + \sum_{i=1}^{n}\sum_{k=0}^{H_1}\lambda_{\text{map1},ik}\,|t_{\text{map1},ik}| + \sum_{c=0}^{H_2}\lambda_{\text{map2},1c}\,|t_{\text{map2},1c}| + \sum_{r=1}^{m}\sum_{l=0}^{H_3}\lambda_{\text{map3},rl}\,|t_{\text{map3},rl}| \quad (5)$$

where H_1, H_2, and H_3 represent the number of hidden neurons of the three mapping ANNs, respectively; $t_{\text{map1},ik}$, $t_{\text{map2},1c}$, and $t_{\text{map3},rl}$ represent the connection weights between hidden neurons and output neurons in three mapping ANNs, respectively; and $\lambda_{\text{map1},ik}$, $\lambda_{\text{map2},1c}$, and $\lambda_{\text{map3},rl}$ are nonnegative coefficients. Using the properties of l_1 optimization, the first stage of training automatically determines whether a nonlinear mapping is needed or not. For each mapping ANN, if all weights in $\boldsymbol{t}$ are zeros after l_1 optimization, the mapping is linear; otherwise, the mapping is nonlinear. If all three mapping ANNs are all nonlinear, the AMG algorithm stops; otherwise, the second-stage training is activated. The algorithm considers whether the linear mappings after the first-stage training can be further deleted by adding the l_1 regular terms of weights in $\boldsymbol{s}$. In this way, the AMG of KBNN can always automatically and efficiently find a suitable mapping scheme for different microwave KBNN modeling problems.

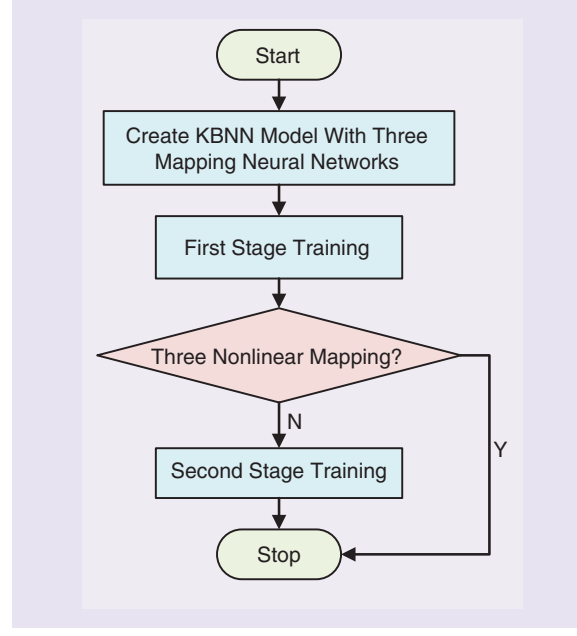


Figure 7. *A flowchart of the AMG for KBNN [38].*

Examples of AMG for Microwave Modeling

Different AMG algorithms have different advantages and are applicable to different microwave modeling problems. The AMG algorithm with interpolation [30] improves the sampling efficiency during the AMG process and is suitable for most of the microwave modeling problems (e.g., example A) to automatically develop a three-layer perception (3LP) model. However, for the microwave modeling problem that has a higher input dimension and a higher degree of nonlinearity (e.g., example B), the simple 3LP structure cannot meet the model accuracy requirement. In this case, the AMG with multilayer ANN adjustment [37] is a better choice to automatically produce an accurate multilayer perception (MLP) model with more than one hidden

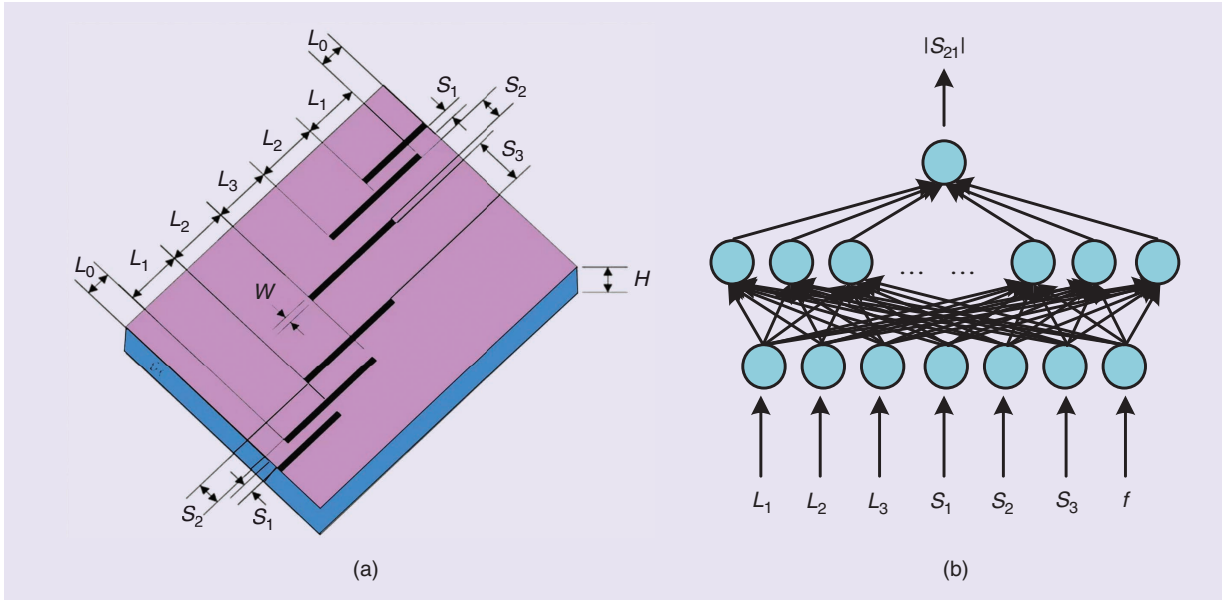


Figure 8. *(a) The EM structure of the bandpass HTS microstrip filter. (b) The three-layer MLP structure used to model the behaviors of the bandpass filter.*

TABLE 1. The modeling results for the bandpass filter example by different methods.

Modeling Algorithm	Manual Modeling Method	Original AMG Without Interpolation [28]	AMG With Interpolation [30]
No. of stages	9	12	6
Initial no. of hidden neurons	25	25	25
Final no. of hidden neurons	33	33	27
No. of training data	4,096 * 1,001	1,394 * 1,001	1,394 * 1,001
No. of testing data	64 * 1,001	64 * 1,001	64 * 1,001
Test error	1.55%	1.31%	1.59%
CPU time (includes EM data generation time)	85.35 h	26.76 h	22.12 h

layer. In addition, for the microwave modeling problem that has prior knowledge such as the equivalent circuits and empirical model (e.g., example C), AMG of KBNN [38] is more suitable to automate the KBNN development process and improve the KBNN development efficiency.

Example of AMG With Interpolation for a Bandpass HTS Microstrip Filter

In this example, the AMG with interpolation algorithm [30] is applied to develop a parametric ANN model of an HTS quarter-wave parallel coupled-line microstrip filter [42], as shown in Figure 8. The model inputs are $x = [L_1, L_2, L_3, S_1, S_2, S_3, f]^T$, which are the lengths of the parallel coupled-line section, the gaps between two sections, and the frequency parameter. The model output is the magnitude of S_{21}, i.e., $y = |S_{21}|$. The user-required testing error of the model for this example is below 1.8%. Training and testing data are generated by driving the EM simulation tool (i.e., CST).

During the AMG with interpolation, the amount and the locations of the data samples are dynamically determined by the interpolation calculations in a stage-wise manner. In each stage, the existing training data are used to build local interpolation models, and the existing testing data are used to evaluate the testing errors of these local interpolation models. More data are added in the region where the testing errors are large. After six stages, errors of these interpolation models are less than 1.8%, and all of the generated data are used to train a three-layer ANN model. After adaptive tuning of the ANN structure, the final model for the bandpass HTS microstrip filter is achieved with a 1.59% testing error. It is important to note that the above model development process is automated by an algorithm-driven computer, and the entire process is completely free of human involvement.

For comparison purposes, the manual modeling method and the original AMG without interpolation [28] are also used to develop ANN models for this example. From the modeling results shown in Table 1, we can see that both AMG algorithms use fewer training data and shorter time than the manual modeling

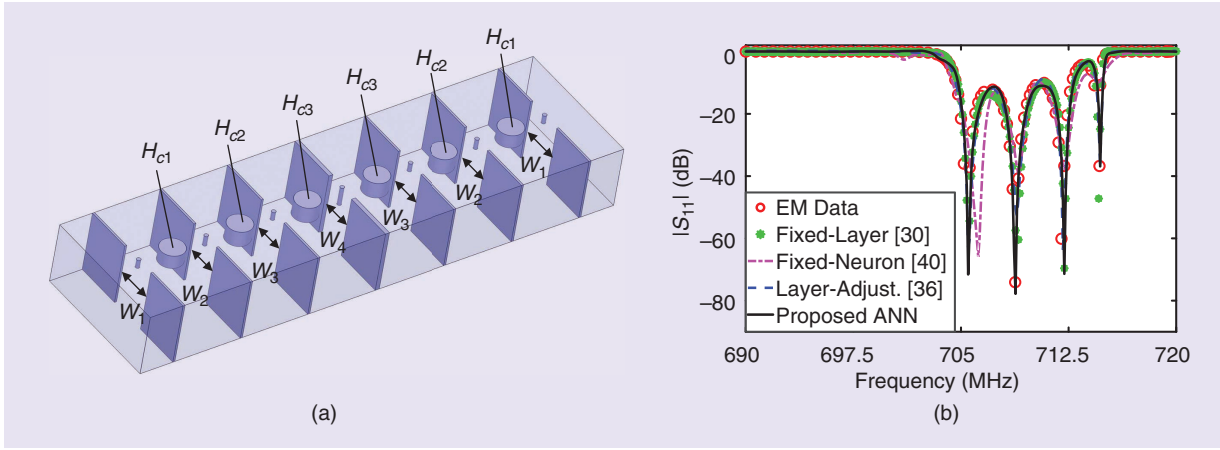


Figure 9. *(a) The EM structure of the iris coupled cavity filter. (b) The responses of the cavity filter at the geometrical sample* $[43.34, 50.09, 49.97, 117.23, 49.37, 44.18, 48.09]^T$ *(mm).*

method to achieve the ANN models with similar accuracy. More importantly, the AMG with interpolation uses interpolation calculations to avoid intermediate ANN training during the dynamic sampling process, so the time for AMG is further reduced, and the efficiency of ANN model development is effectively improved compared with the AMG algorithm without interpolation.

Example of AMG With Multilayer ANN Structure Adjustment for an Iris Coupled Cavity Filter

In this example, the AMG algorithm with multilayer ANN structure adjustment is applied to develop a surrogate model of an iris-coupled cavity filter [28], [43], as shown in Figure 9(a). This model has seven geometrical parameters and the frequency parameter as its inputs, i.e., $\boldsymbol{x} = [H_{c1}, H_{c2}, H_{c3}, W_1, W_2, W_3, W_4, f]^T$, and the real and imaginary parts of S_{11} as its outputs, i.e., $\boldsymbol{y} = [\mathrm{Re}(S_{11}), \mathrm{Im}(S_{11})]^T$. The user-defined testing error threshold for this example is 2%. For comparison, various AMG algorithms are performed to develop an ANN model for this cavity filter. Table 2 and Figure 6(b) show the modeling results of different AMG algorithms. With the same initialization of the ANN setup, the AMG algorithm with multilayer ANN structure adjustment consumes the shortest CPU time for achieving an accurate ANN final model, therefore improving the modeling efficiency.

TABLE 2. A comparison of the cavity filter modeling results by different AMG algorithms.

Modeling Method	Final Model: No. of Hidden Layers	Final Model: No. of Neurons in Each Hidden Layer	Testing Error	CPU Time
AMG with MLP [30] (fixed no. of layers)	4	$H_i = 15$, $1 \le i \le 4$	1.83%	29.31 h
AMG with MLP [40] (fixed no. of neurons in each layer)	1	$H_1 = 70$	5.07%	26.09 h
AMG with multilayer ANN structure adjustment [37]	3	$H_1 = 70$, $H_2 = 22$, $H_3 = 13$	1.78%	5.00 h

Example of AMG of KBNN for a Two-Section Low-Pass Elliptic Microstrip Filter

In this example, the AMG algorithm of the KBNN is applied to develop a parametric ANN model for a low-pass filter with double microstrip sections [44], [45]. As shown in Figure 10(a), the model inputs are $\boldsymbol{x} = [L_1, L_2, L_{c1}, L_{c2}, W_c, G_c]^T$, i.e., the lengths and widths of the microstrip lines of the filter, and the model output is $y = |S_{21}|$, which is the magnitude of S_{21}. The prior knowledge in the KBNN is shown in Figure 10(b), which is the equivalent circuit of the low-pass filter combined by simple transmission lines. We use the AMG algorithm [38] to develop the KBNN for two different cases (i.e., small modeling range as case 1 and large modeling range as case 2). After the l_1 regularization-based training, the final KBNN model is achieved with different mapping structures for different cases. For case 1, only one linear input mapping is sufficient to achieve an accurate KBNN model, while for case 2, one nonlinear input mapping and one nonlinear frequency mapping are chosen by the AMG algorithm to achieve an accurate KBNN model. The modeling results for two cases are shown in Figure 7(c). It is demonstrated

that the AMG algorithm of KBNN can adaptively find the appropriate mapping type and topology for different microwave modeling problems.

Conclusion

Different AMG algorithms for microwave applications are reviewed in this article. The advantages and disadvantages of these AMG algorithms compared with the manual ANN modeling method are discussed in Table 3. By integrating all subtasks required in ANN modeling into a unified framework, AMG algorithms convert the conventional human intensive modeling process into an automated process, thus significantly improving the ANN modeling efficiency. Compared with the original AMG algorithm [28], AMG with interpolation [30] avoids the intermediate

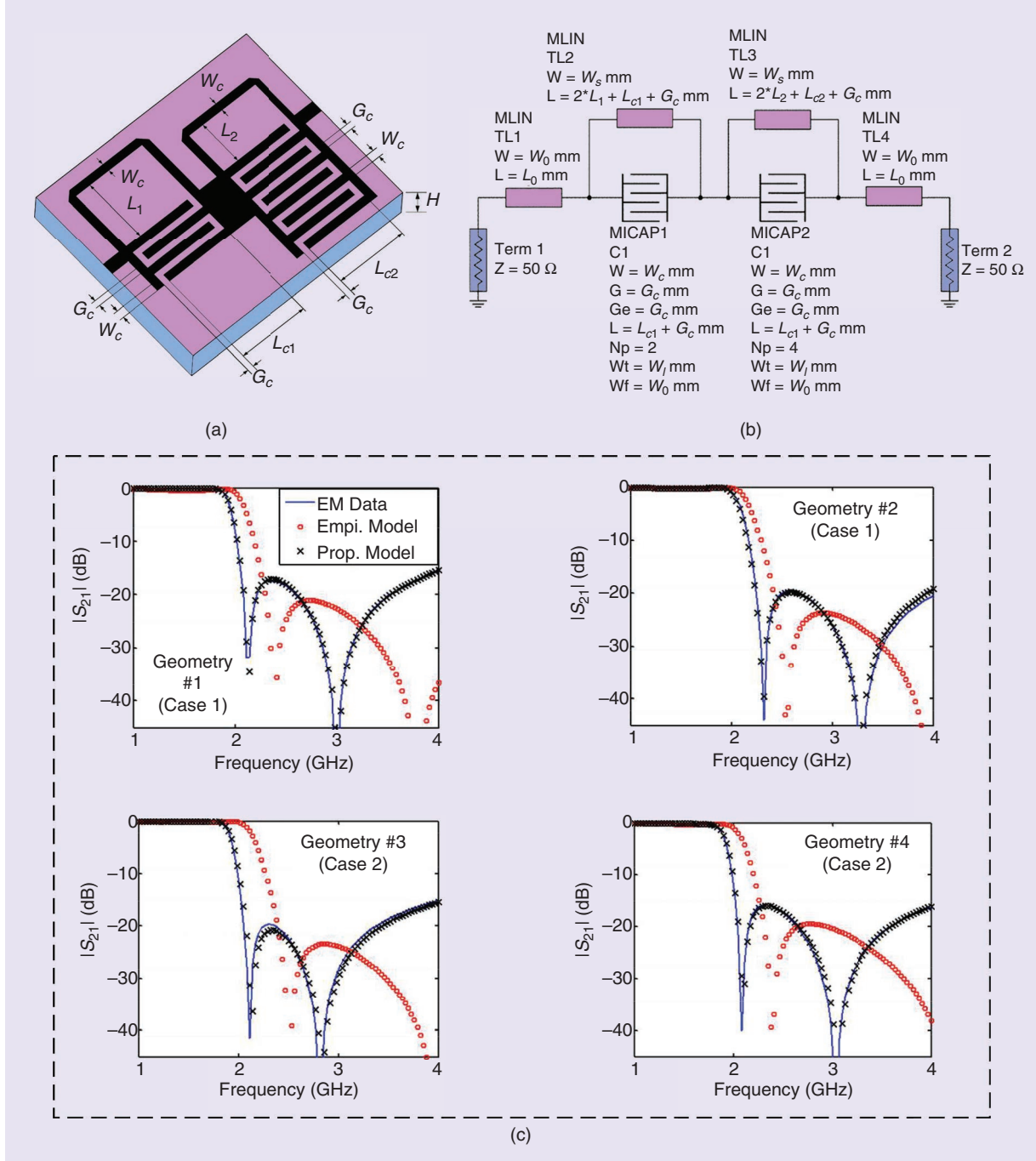


Figure 10. *(a) The EM structure of the low-pass filter. (b) The equivalent circuit of the low-pass filter. (c) The KBNN modeling results of the low-pass filter at four different geometrical samples for cases 1 and 2.*

TABLE 3. Comparisons among manual modeling and different AMG modeling algorithms.

	Modeling Method				
	Manual ANN Modeling	**Original AMG Algorithm [28]**	**AMG With Interpolation [30]**	**AMG With Multilayer ANN Structure Adjustment [37]**	**AMG of KBNN [38]**
Requirement of human intervention	Yes	No	No	No	No
Training of intermediate ANN models	Yes	Yes	No	No	No
Automated adjustment of the hidden neurons number	No	Yes	Yes	Yes	Yes
Automated adjustment of the hidden layers number	No	No	No	Yes	No
Effective weights computation using l_1 regularization	No	No	No	Yes	Yes
Automated mapping types and topologies adjustment	No	No	No	No	Yes
Modeling speed	Slow	Medium	Medium	Fast	Fast

ANN training and speeds up the dynamical sampling process during AMG. It is applicable for most of the microwave modeling problems to automatically develop a 3LP model with one hidden layer. However, when the microwave modeling problem is high dimensional or wide ranging, a 3LP structure with only one hidden layer is not sufficient to learn the complicated relationship between the inputs and outputs and satisfy the user-desired model accuracy. In this case, the AMG with multilayer ANN adjustment [37] is a good choice to automatically produce an accurate MLP model in the shortest possible time. In addition, when the microwave modeling problem has prior knowledge such as equivalent circuits and empirical models, AMG of KBNN [38] is more suitable to automatically generate a KBNN model with appropriate mapping types and topological structure that meets the user-desired accuracy. The future directions may be to incorporate EM sensitivity and more efficient sampling methods into AMG algorithms to address more challenging microwave modeling problems (e.g., high-dimensional and wide-ranging modeling problems), further accelerating the ANN development process.

Acknowledgment

The corresponding author is Weicong Na.

References

[1] Q.-J. Zhang and K. C. Gupta, *Neural Networks for RF and Microwave Design*. Norwood, MA, USA: Artech House, 2000.

[2] J. E. Rayas-Sanchez, "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, doi: 10.1109/TMTT.2003.820897.

[3] 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, doi: 10.1109/TMTT.2003.809179.

[4] Q. Wang, H. Tamukoh, and T. Morie, "A time-domain analog weighted-sum calculation model for extremely low power VLSI implementation of multi-layer neural networks," 2018, *arXiv:1810.06819*.

[5] Y. Sakemi, K. Morino, T. Morie, and K. Aihara, "A supervised learning algorithm for multilayer spiking neural networks based on temporal coding toward energy-efficient VLSI processor design," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 34, no. 1, pp. 394–408, Jan. 2023, doi: 10.1109/TNNLS.2021.3095068.

[6] M. Swaminathan, H. M. Torun, H. Yu, J. A. Hejase, and W. D. Becker, "Demystifying machine learning for signal and power integrity problems in packaging," *IEEE Trans. Compon., Packag. Manuf. Technol.*, vol. 10, no. 8, pp. 1276–1295, Aug. 2020, doi: 10.1109/TCPMT.2020.3011910.

[7] Y. Zhuo, F. Feng, J. Zhang, and Q.-J. Zhang, "Parametric modeling incorporating joint polynomial-transfer function with neural networks for microwave filters," *IEEE Trans. Microw. Theory Techn.*, vol. 70, no. 11, pp. 4652–4665, Nov. 2022, doi: 10.1109/TMTT.2022.3207761.

[8] Z. Ye, W. Shao, X. Ding, B.-Z. Wang, and S. Sun, "Knowledge-based neural network for multiphysical field modeling," *IEEE Trans. Microw. Theory Techn.*, vol. 71, no. 5, pp. 1967–1976, May 2023, doi: 10.1109/TMTT.2022.3227333.

[9] F. Feng et al., "Feature and EM sensitivity co-assisted neuro-TF surrogate optimization for microwave filter design," *IEEE Trans. Microw. Theory Techn.*, vol. 71, no. 11, pp. 4749–4761, Nov. 2023, doi: 10.1109/TMTT.2023.3275232.

[10] J. Jin, F. Feng, J. Zhang, J. Ma, and Q.-J. Zhang, "Efficient EM topology optimization incorporating advanced matrix Padé via Lanczos and genetic algorithm for microwave design," *IEEE Trans. Microw. Theory Techn.*, vol. 69, no. 8, pp. 3645–3666, Aug. 2021, doi: 10.1109/TMTT.2021.3088870.

[11] A. Gupta, E. A. Karahan, C. Bhat, K. Sengupta, and U. K. Khankhoje, "Tandem neural network based design of multiband antennas," *IEEE Trans. Antennas Propag.*, vol. 71, no. 8, pp. 6308–6317, Aug. 2023, doi: 10.1109/TAP.2023.3276524.

[12] J. Li et al., "An adaptive evolutionary neural network-based optimization design method for wideband dual-polarized antennas," *IEEE Trans. Antennas Propag.*, vol. 71, no. 10, pp. 8165–8172, Oct. 2023, doi: 10.1109/TAP.2023.3301882.

[13] B. Zhang, C. Jin, K. Cao, Q. Lv, and R. Mittra, "Cognitive conformal antenna array exploiting deep reinforcement learning method," *IEEE Trans. Antennas Propag.*, vol. 70, no. 7, pp. 5094–5104, Jul. 2022, doi: 10.1109/TAP.2021.3096994.

[14] J.-J. Sun, S. Sun, L.-J. Yang, and J. Hu, "Enhanced misalignment estimation of orbital angular momentum signal based on deep recurrent neural networks," *IEEE Trans. Antennas Propag.*, vol. 71, no. 6, pp. 5522–5527, Jun. 2023, doi: 10.1109/TAP.2023.3263945.

[15] J. W. Zhang et al., "A novel two-stage optimization framework for designing active metasurfaces based on multi-port microwave network theory," *IEEE Trans. Antennas Propag.*, vol. 72, no. 2, pp. 1603–1616, Feb. 2024, doi: 10.1109/TAP.2023.3340769.

[16] J. Zhang et al., "Physics-driven machine-learning approach incorporating temporal coupled mode theory for intelligent design of metasurfaces," *IEEE Trans. Microw. Theory Techn.*, vol. 71, no. 7, pp. 2875–2887, Jul. 2023, doi: 10.1109/TMTT.2023.3238076.

[17] S. Li, Z. Liu, S. Fu, Y. Wang, and F. Xu, "Intelligent beamforming via physics-inspired neural networks on programmable metasurface," *IEEE Trans. Antennas Propag.*, vol. 70, no. 6, pp. 4589–4599, Jun. 2022, doi: 10.1109/TAP.2022.3140891.

[18] E. Wen, X. Yang, and D. F. Sievenpiper, "Real-time 2-D beamforming with rotatable dielectric slabs enabled by generative neural network," *IEEE Trans. Antennas Propag.*, vol. 70, no. 9, pp. 8360–8367, Sep. 2022, doi: 10.1109/TAP.2022.3161365.

[19] Q. Lin, H.-F. Wu, L.-S. Liu, L.-N. Jia, and X.-Z. Wang, "Optimization design of RF power amplifier based on ANN," in *Proc. Int. Conf. Microw. Millimeter Wave Technol. (ICMMT)*, Piscataway, NJ, USA: IEEE, May 2021, pp. 1–3, doi: 10.1109/ICMMT52847.2021.9617982.

[20] P. Chen, B. M. Merrick, and T. J. Brazil, "Bayesian optimization for broadband high-efficiency power amplifier designs," *IEEE Trans. Microw. Theory Techn.*, vol. 63, no. 12, pp. 4263–4272, Dec. 2015, doi: 10.1109/TMTT.2015.2495360.

[21] T. Kobal, Y. Li, X. Wang, and A. Zhu, "Digital predistortion of RF power amplifiers with phase-gated recurrent neural networks," *IEEE Trans. Microw. Theory Techn.*, vol. 70, no. 6, pp. 3291–3299, Jun. 2022, doi: 10.1109/TMTT.2022.3161024.

[22] C. Jiang, G. Yang, R. Han, J. Tan, and F. Liu, "Gated dynamic neural network model for digital predistortion of RF power amplifiers with varying transmission configurations," *IEEE Trans. Microw. Theory Techn.*, vol. 71, no. 8, pp. 3605–3616, Aug. 2023, doi: 10.1109/TMTT.2023.3241612.

[23] K. Lee and W. Lee, "Deep learning framework for two-way MISO wireless-powered interference channels," *IEEE Trans. Wireless Commun.*, vol. 22, no. 10, pp. 6459–6473, Oct. 2023, doi: 10.1109/TWC.2023.3243611.

[24] S. Qi and C. D. Sarris, "Physics-informed neural networks for multiphysics simulations: Application to coupled electromagnetic-thermal modeling," in *Proc. IEEE/MTT-S Int. Microw. Symp. (IMS)*, Piscataway, NJ, USA: IEEE, Jun. 2023, pp. 166–169, doi: 10.1109/IMS37964.2023.10188015.

[25] W. Zhang, F. Feng, J. Jin, and Q.-J. Zhang, "Parallel multiphysics optimization for microwave devices exploiting neural network surrogate," *IEEE Microw. Wireless Compon. Lett.*, vol. 31, no. 4, pp. 341–344, Apr. 2021, doi: 10.1109/LMWC.2021.3053600.

[26] J. Dávalos-Guzmán, J. L. Chavez-Hurtado, and Z. Brito-Brito, "Neural network learning techniques comparison for a multiphysics second order low-pass filter," in *Proc. IEEE MTT-S Latin America Microw. Conf. (LAMC)*, Piscataway, NJ, USA: IEEE, Dec. 2023, pp. 102–104, doi: 10.1109/LAMC59011.2023.10375591.

[27] W. Zhang, F. Feng, S. Yan, W. Na, J. Ma, and Q.-J. Zhang, "EM-centric multiphysics optimization of microwave components using parallel computational approach," *IEEE Trans. Microw. Theory Techn.*, vol. 68, no. 2, pp. 479–489, Feb. 2020, doi: 10.1109/TMTT.2019.2955117.

[28] V. K. Devabhaktuni, M. C. E. Yagoub, and Q. J. Zhang, "A robust algorithm for automatic development of neural-network models for microwave applications," *IEEE Trans. Microw. Theory Techn.*, vol. 49, no. 12, pp. 2282–2291, Dec. 2001, doi: 10.1109/22.971611.

[29] W. C. Na and Q. J. Zhang, "Automated knowledge-based neural network modeling for microwave applications," *IEEE Microw. Wireless Compon. Lett.*, vol. 24, no. 7, pp. 499–501, Jul. 2014, doi: 10.1109/LMWC.2014.2316251.

[30] W. Na and Q. Zhang, "Automated parametric modeling of microwave components using combined neural network and interpolation techniques," in *Proc. IEEE MTT-S Int. Microw. Symp. Dig. (MTT)*, Piscataway, NJ, USA: IEEE, Jun. 2013, pp. 1–3, doi: 10.1109/MWSYM.2013.6697547.

[31] W. Na, W. Liu, K. Liu, and F. Feng, "Recent advances in automated multilayer ANN model generation for microwave components," in *Proc. Int. Appl. Comput. Electromagnet. Soc. Symp. (ACES-China)*, Piscataway, NJ, USA: IEEE, Aug. 2023, pp. 1–3, doi: 10.23919/ACES-China60289.2023.10249861.

[32] I. Lee, S. Hong, G. Ryu, and Y. Park, "Automated neural network accelerator generation framework for multiple neural network applications," in *Proc. TENCON IEEE Region 10 Conf.*, Piscataway, NJ, USA: IEEE, Oct. 2018, pp. 2287–2290, doi: 10.1109/TENCON.2018.8650190.

[33] W. Na, W. Zhang, S. Yan, F. Feng, W. Zhang, and Y. Zhang, "Automated neural network-based multiphysics parametric modeling of microwave components," *IEEE Access*, vol. 7, pp. 141,153–141,160, 2019, doi: 10.1109/ACCESS.2019.2944162.

[34] J. Cui, F. Feng, W. Na, and Q.-J. Zhang, "Bayesian-based automated model generation method for neural network modeling of microwave components," *IEEE Microw. Wireless Compon. Lett.*, vol. 31, no. 11, pp. 1179–1182, Nov. 2021, doi: 10.1109/LMWC.2021.3087163.

[35] J. Cui, F. Feng, J. Zhang, L. Zhu, and Q.-J. Zhang, "Bayesian-assisted multilayer neural network structure adaptation method for microwave design," *IEEE Microw. Wireless Technol. Lett.*, vol. 33, no. 1, pp. 3–6, Jan. 2023, doi: 10.1109/LMWC.2022.3201123.

[36] W. Na, K. Liu, W. Zhang, F. Feng, H. Xie, and D. Jin, "Automated multilayer neural network structure adaptation method with $l1$ regularization for microwave modeling," *IEEE Microw. Wireless Compon. Lett.*, vol. 32, no. 7, pp. 815–818, Jul. 2022, doi: 10.1109/LMWC.2022.3153058.

[37] W. Na, K. Liu, J. Zhang, D. Jin, H. Xie, and W. Zhang, "An efficient batch-adjustment algorithm for artificial neural network structure adaptation and applications to microwave modeling," *IEEE Microw. Wireless Technol. Lett.*, vol. 33, no. 8, pp. 1107–1110, Aug. 2023, doi: 10.1109/LMWT.2023.3276746.

[38] W. Na, F. Feng, C. Zhang, and Q.-J. Zhang, "A unified automated parametric modeling algorithm using knowledge-based neural network and l_1 optimization," *IEEE Trans. Microw. Theory Techn.*, vol. 65, no. 3, pp. 729–745, Mar. 2017, doi: 10.1109/TMTT.2016.2630059.

[39] W. Na and Q. Zhang, "Unified automated knowledge-based neural network modeling for microwave devices," in *Proc. IEEE MTT-S Int. Conf. Numer. Electromagnet. Multiphys. Model. Optim. (NEMO)*, Piscataway, NJ, USA: IEEE, Aug. 2015, pp. 1–3, doi: 10.1109/NEMO.2015.7415029.

[40] J. Jin, C. Zhang, F. Feng, W. Na, J. Ma, and Q.-J. Zhang, "Deep neural network technique for high-dimensional microwave modeling and applications to parameter extraction of microwave filters," *IEEE Trans. Microw. Theory Techn.*, vol. 67, no. 10, pp. 4140–4155, Oct. 2019, doi: 10.1109/TMTT.2019.2932738.

[41] J. Hald and K. Madsen, "Combined LP and quasi-Newton methods for nonlinear l_1 optimization," *SIAM J. Numer. Anal.*, vol. 22, no. 1, pp. 68–80, Feb. 1985, doi: 10.1137/0722004.

[42] J. W. Bandler et al., "Space mapping: the state of the art," *IEEE Trans. Microw. Theory Techn.*, vol. 52, no. 1, pp. 337–361, Jan. 2004, doi: 10.1109/TMTT.2003.820904.

[43] F. Feng et al., "Multifeature-assisted neuro-transfer function surrogatebased EM optimization exploiting trust-region algorithms for microwave filter design," *IEEE Trans. Microw. Theory Techn.*, vol. 68, no. 2, pp. 531–542, Feb. 2020, doi: 10.1109/TMTT.2019.2952101.

[44] R. Esfandiari, D. Maku, and M. Siracusa, "Design of interdigitated capacitors and their application to gallium arsenide monolithic filters," *IEEE Trans. Microw. Theory Techn.*, vol. 31, no. 1, pp. 57–64, Jan. 1983, doi: 10.1109/TMTT.1983.1131429.

[45] F. Feng, C. Zhang, V.-M.-R. Gongal-Reddy, Q.-J. Zhang, and J. Ma, "Parallel space-mapping approach to EM optimization," *IEEE Trans. Microw. Theory Techn.*, vol. 62, no. 5, pp. 1135–1148, May 2014, doi: 10.1109/TMTT.2014.2315781.