Turkish Journal of Electrical Engineering & Computer Sciences

http://journals.tubitak.gov.tr/elektrik/

Research Article

Turk J Elec Eng & Comp Sci
(2018) 26: 2437 – 2447

doi:10.3906/elk-1705-362

Design of an on-chip Hilbert fractal inductor using an improved feed forward neural network for Si RFICs

Akhendra Kumar PADAVALA*, **Bheema Rao NISTALA**
Department of Electronics & Communication Engineering, National Institute of Technology, Warangal, India

Received: 27.05.2017 • **Accepted/Published Online:** 26.05.2018 • **Final Version:** 28.09.2018

Abstract: This paper presents an efficient modeling of Hilbert fractal inductors by improved feed forward neural network trained hybrid particle swarm optimization and gravitational search algorithm (FNNPSOGSA). The proposed model computes the effective inductance value (L) and quality factor (Q) of Hilbert fractal inductors with metal trace width, effective fractal length, frequency, and oxide thickness as input parameters. In contrast to the traditional feed forward neural network, the proposed FNNPSOGSA has been designed with fewer hidden neurons with much-enhanced learning and generalization capabilities. As a consequence, the proposed model achieves better speed and is as accurate as electromagnetic simulations. From the simulation results, it is proved that the proposed model is a good alternative for complex fractal inductor design.

Key words: Hybrid particle swarm optimization and gravitational search algorithm (PSOGSA), high-frequency structural simulator (HFSS), inductance value (L), quality factor (Q)

1. Introduction

The rapid growth in wireless communication systems demands low cost, miniature radio frequency integrated circuits (RFICs). The quality of passive components like inductors, capacitors, and resistors plays a crucial role in determining the performance of CMOS RFICs. Inductors are the most critical component among all passive components. The crucial aspects of an inductor design are higher quality factor (Q) and high self-resonant frequency (SRF) for a specified inductance value (L) with the minimum on-chip area. Inductors designed based on space-filling curves can solve this problem. An exhaustive study of conventional fractal inductors was carried out in [1,2]. The modified fractal inductors are proposed in single and multilayer processes to achieve higher L and Q values [3,4]. These fractal inductors are designed using an EM simulator, which gives accurate results but is time-consuming.

Numerous studies have been done on the tradeoffs of layout and technology parameters to maximize Q at desired L and frequency. The absence of a high-performance computer-aided design tool is still one of the drawbacks of inductor design [5,6]. Mathematical analysis for inductor design is accurately solved by using well-known numerical methods such as the finite-difference time-domain method (FDTD) [7], the moment of method (MOM) [8], and the finite element method (FEM) [9]. However, these numerical methods are very slow. Moreover, a small change in the geometry or material parameters would require a completely new simulation. Analytical methods such as the Greenhouse method [10] and Grover method [11] have been proposed

*Correspondence: akhendra.p@gmail.com

to determine the self and mutual inductance of rectangular cross sections. However, these analytical expressions have very limited applicability due to the complexity of fractal structures and low accuracy.

Artificial neural network (ANN) is a prominent learning tool to model complex relationships between inputs and outputs. The model parameters such as connection weights and biases are tuned based on neural network training. Neural models are proving to be faster than electromagnetic (EM) models. Along with speed, they are more accurate than analytical and empirical models [12]. Several heuristic optimization algorithms have been proposed to train the neural network such as genetic algorithm (GA) and simulated annealing (SA) in data mining [13] and microwave filter [14] applications. Both these algorithms suffer from slow converge rates and tapering at local minima. Hybrid algorithms have been recently proposed to train the neural network, to improve the convergence rate and to eliminate the tapering of local minima. A hybrid gradient descent and PSO training algorithm has also been proposed to predict the antenna gain, bandwidth, and polarization of broadband antennae [15]. A neural network-based hybrid GA and Levenberg Marquardt (GA-LM) algorithm have been proposed for maximum power point tracking (MPPT) [16]. The extended Kalman filter (EKF) and particle swarm optimization (PSO) training algorithms have been proposed for smart grid applications [17]. A hybrid gradient teacher learning optimization (TLBO) is adopted to design a waveguide using a neural network [18].

ANN has emerged as an alternative to conventional spiral inductor modeling techniques. A feed forward neural network trained using a multilayer perception neural network was developed to design an octagonal inductor [19], a rounded spiral inductor [20], and a rectangular inductor [21]. Generalized knowledge-based neural network (GKBNN) [22] and space mapping neural network (SMNN) [23] were developed to design rounded spiral inductors. Similarly, knowledge-based frequency-dependent space-mapping neural network (KB-FDSMN) [24], combined neural network, transfer function [25], and physics-based sampling neural network [26] were proposed for the design of rectangular spiral inductors. However, the modelling of fractal inductors using ANN was not reported earlier.In this paper, an efficient feed forward neural network trained by hybrid particle swarm optimization and the gravitational search algorithm (PSOGSA) is proposed for designing complex Hilbert fractal inductors based on design specifications. The rest of the paper is structured as follows: Section 2 describes the design procedure of the fractal inductor. Section 3 presents a brief overview of the improved feed-forward neural network. Section 4 is devoted to the validation of results. Finally, the conclusions are shown in Section 5.

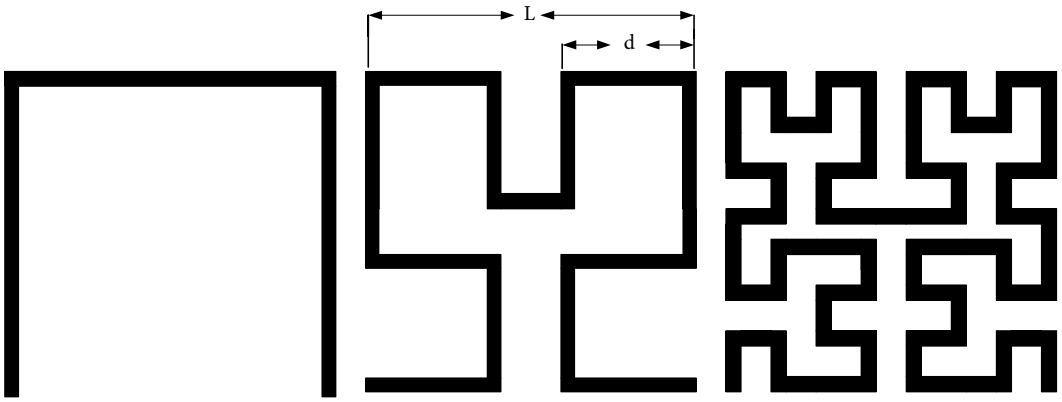
2. Hilbert fractal inductor

2.1. Hilbert curve construction

A fractal is a continuous curve with a characteristic of self-similarity [27]. In 1890, Peano was the first to discover a space-filling curve followed by many such curves over the years. Among them, the Hilbert curve has been the most extensively used space-filling curve. These curves are formed by an iterative process. Figure 1 shows the various stages of Hilbert fractal curves from the first order to the third order.

2.2. Hilbert fractal inductors design

The physical design of a Hilbert fractal inductor is depicted in Figure 2. It is a two-port device characterized by two-port S- or Y-parameters. The layout parameters and technology parameters play an important role in the construction of Hilbert fractal inductors. The layout parameters are the number of iterations (n), the



(a) zeroth , (b) first , (c) second iterations

Figure 1. Construction of Hilbert curve.

width of metal traces, effective fractal length, and frequency. The technology parameters used in the design are summarized in Table 1.

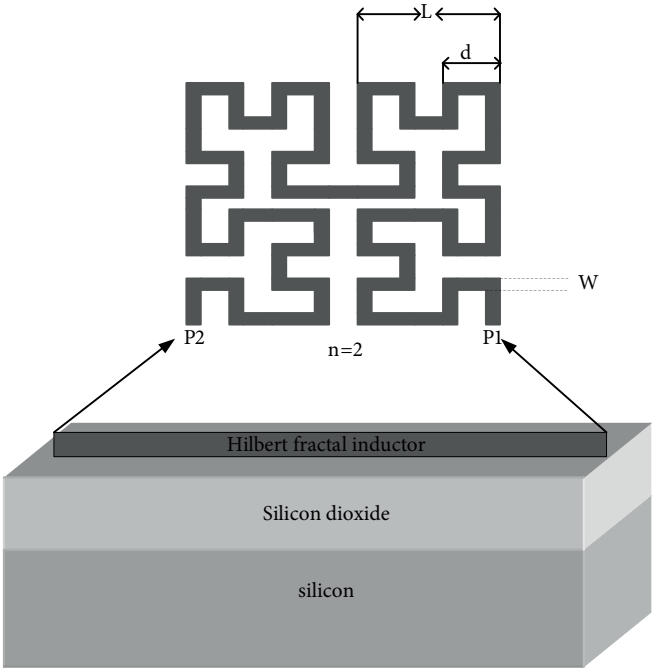


Figure 2. Physical layout of on-chip Hilbert fractal inductor.

3. Improved feedforward neural network based on PSOGSA

3.1. FNNPSOGSA model development

The FNN trained by PSOGSA is a more efficient neural network than the conventional neural network. The improved FNN trained by PSOGSA is shown in Figure 3. It consists of an input layer, a hidden layer, and an output layer. The neurons are connected directly to the next layer via a series of weights. In this paper, the

Table 1. Technology parameters.

Parameter	Value
Silicon dielectric constant (ε_r)	11.9
Silicon dioxide dielectric constant (ε_r)	4
Metal resistance	20 m$\Omega/\square$
Metal thickness	2 μm
Substrate resistivity	10 Ω cm

effective fractal length, the width of metal traces, and frequency are considered as inputs to design the neural model. However, technology parameters are excluded since the designer has no control over them for a given fabrication process. The output neurons represent electrical attributes of the inductor, which are L and Q. The developed neural network has one hidden layer with 15 neurons chosen to optimize the accuracy and speed of convergence of the neural network. Input neurons are activated using the sigmoid function and a linear function is used to activate the hidden neurons. Hilbert fractal inductors are developed to generate training and testing data sets based on the constraints in geometry as shown in Table 2. Given the standard lattice distribution sampling procedure, every input is sampled at an equal interval as given in Table 2. All theoretically possible inductors are developed and simulated using a high frequency structural simulator "ANSYS" HFSS). The L and Q for each inductor were extracted from the EM simulator using Eqs. (1) and (2).

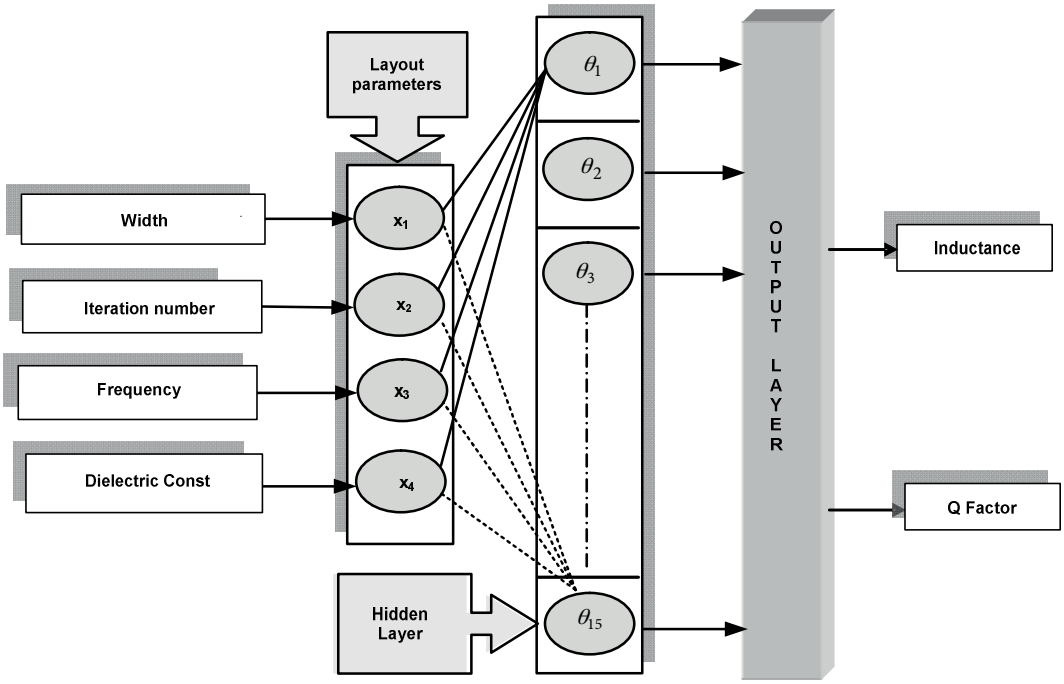


Figure 3. The Proposed FNNPSOGSA model for a Hilbert fractal inductor.

$$L = \frac{Imag(1/Y_{11})}{2\pi f} \tag{1}$$

Table 2. Range of input parameters.

	Width (μm)	Iteration number (n)	Effective fractal length (μm)	Frequency (GHz)	Dielectric thickness
Min	5	0	200	0.1	0.5
Max	15	3	3400	30	8.5
Step	3	1	-	0.1	4

$$Q = \frac{Imag(1/Y_{11})}{Re((1/Y_{11})}, \tag{2}$$

where Y_{11} is an input admittance of the two-port network.

Out of the generated inductor data parameters 75% of data were used for training the neural network and the remaining 25% data for testing. The accuracy was determined using test data. Training and test data were selected in such a way that both covered entire data samples. The input and output parameters for constructing a neural network vary over a wide range as shown in Table 2. Prior to the training process, input and output data were normalized using linear scaling over the [–1 1] range to obtain good results as well as speed up the network calculations. The weights and biases of the neural network were adjusted to minimize the training error during the training process. The calculated training error was a measure of accuracy of the neural network.

The objective of an FNN is to obtain best-fitted weights to minimize the mean squared error based on a learning algorithm. The PSOGSA is considered to train the neural network as it combines the social thinking ability of PSO with local search capabilities of GSA [28]. PSOGSA has an advantage of high convergence rate and it is does not tend to be tapered at local minima.

3.2. PSOGSA learning algorithm

The concept of PSOGSA learning algorithm is explained using the flowchart shown in Figure 4. From Figure 4, all the agents are randomly initialized with random velocity and positions. After the initialization, the gravitational force of an i^{th} particle from the j^{th} particle in d^{th} time interval is calculated using Eq. (3).

$$F_{ij}^{d} = G(t)\frac{M_{pi}(t) \times M_{aj}(t)}{R_{ij}(t) + \varepsilon}(x_j^d(t) - x_i^d(t)), \tag{3}$$

where $G(t)$ is a gravitational constant in generation 't' and is calculated using Eq. (4).

$$G(t) = G_O \times \exp(-\alpha \times iter/\max iter) \tag{4}$$

$R_{ij}(t)$ is the Euclidian distance between the particles 'i' and 'j' given by using Eq. (5).

$$R_{ij}(t) = \|X_i(t), X_j(t)\|_2, \tag{5}$$

where $M_{pi}(t), M_{aj}(t)$ are the passive and active mass coefficients. The total force that acts on the ith particle is given by Eq. (6) and the acceleration of the ith particle is given by Eq. (7).

$$F_i^d(t) = \sum_{j=1, j\neq 1}^{N} rand_j F_{ij}^d(t) \tag{6}$$

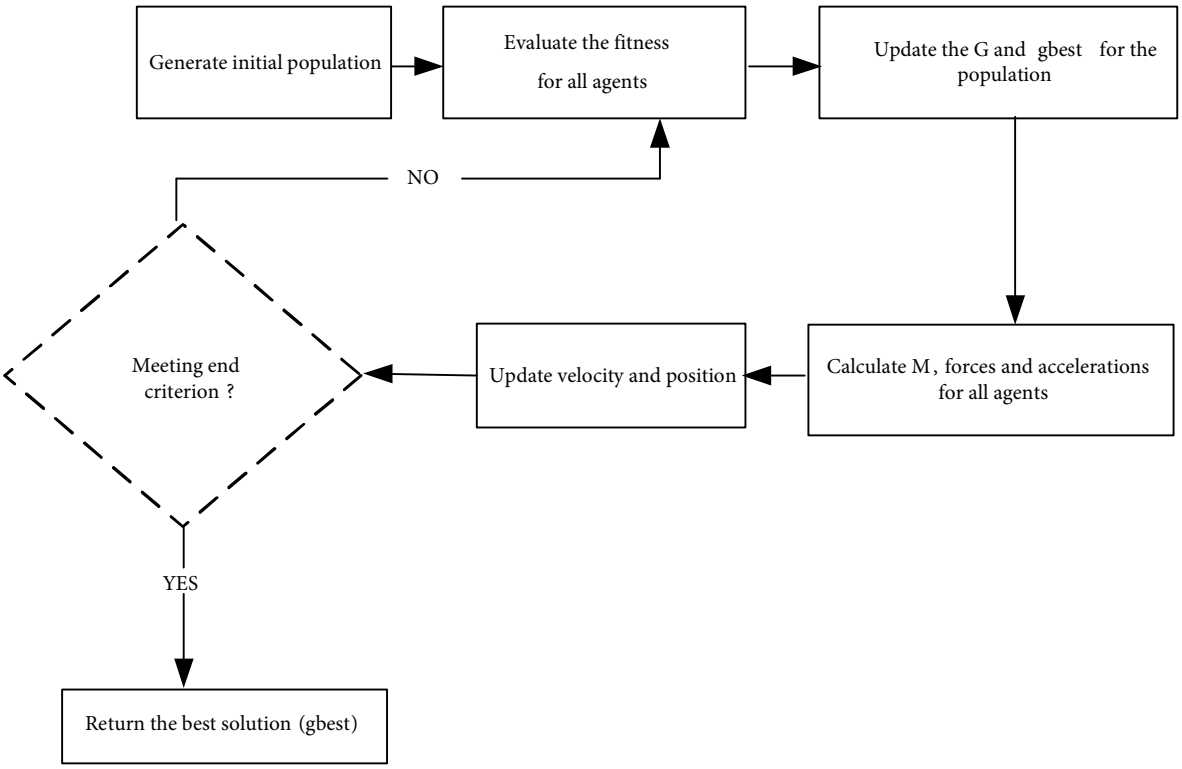


Figure 4. PSOGSA algorithm flow.

$$a_i^d(t) = \frac{F_i^u(t)}{M_{ii}(t)'} \tag{7}$$

In each iteration, the best solution is updated as *gbest*. After calculating the accelerations and updating the best solution so far, the velocities of all agents can be updated using Eq. (8). Finally, the positions of agents are updated using Eq. (9). The process of updating velocities and positions is stopped by meeting an end criterion.

$$V_i(k+1) = w \times V_i(k) + C_1^{'} \times rand \times aC_i(k) + C_2^{'} \times rand \times (gbest - X_i(k)), \tag{8}$$

where 'w' is weight function, $C_1^{'}$ and $C_2^{'}$ are the acceleration constants, and $\mathrm{V}_i(\mathrm{k})$ is the velocity of the i^{th} agent at the k^{th} iteration and *rand* is any random value.

$$X_i(k+1) = X_i(k) + V_i(k+1) \tag{9}$$

The developed training algorithm is used to find the best combination of weights and biases, which provide minimum training error for an ANN. The mean training error is calculated from the fitness function of an agent.

In epoch learning the output of each hidden node is calculated using Eq. (10).

$$f(S_j) = 1/\left(1 + \exp\left(-\left(\sum_{i=1}^{n} w_{ij}.x_i - \theta_j\right)\right)\right), \quad j = 1, 2, ..., h, \tag{10}$$

where 'n' is the number of the input nodes, w_{ij} is the connection weight from the i^{th} node in the input layer to the j^{th} node in the hidden layer, θ_j is the bias (threshold) of the j^{th} hidden node, and x_i is the i^{th} input.

After calculating output of the hidden nodes, the final output is given by Eq. (11).

$$O_k = \sum_{i=1}^{h} w_{kj}.f(S_j) - \theta_k, \qquad k = 1, 2..., m, \tag{11}$$

where W_{kj} is the connection weight from the jth hidden node to the kth output node and θ_k is the bias (threshold) of the kth output node. Finally, the learning error E (fitness function) is calculated from Eqs. **??**) and (13).

$$E_k = \sum_{i=1}^{m} \left(O_i^k - d_i^k\right)^2 \tag{12}$$

$$E = \sum_{k=1}^{q} \frac{E_k}{q}, \tag{13}$$

where q' is the number of training samples, d_i^k is the desired output of the ith input unit when the kth training sample is used and O_i^k is the actual output of the ith input unit when the kth training sample is used. Therefore, the fitness function of the ith training sample can be defined using Eq. (14).

$$Fitness(X_i) = E(X_i) \tag{14}$$

After defining the fitness function for FNNPSOGSA, the matrix encoding and decoding strategy is adopted to represent the weights and biases of the each agent in the FNNPSOGSA due to its easy decoding strategy.

4. Results and discussion

The RF silicon-based Hilbert fractal inductor is designed using the developed FNNPSOGSA and the conventional FNN trained using an LM algorithm (FNNLM). The conventional neural network is a two hidden layer neural network each with 25 hidden neurons having the same input and output data parameters used in FNNPSOGSA. To evaluate the performance of the FNNPSOGSA model, mean squared error is calculated and then compared with FNNLM. The convergence details of the PSOGSA learning algorithm are compared with the LM learning algorithm as shown in Figure 5. The results showed that the PSOGSA algorithm converges after 20 epochs with a mean square error of 0.0006781, whereas the LM algorithm converges after 50 epochs with a mean square error of 0.05. From the results, it can be observed that the FNNLM model entails much longer training time than the developed FNNPSOGSA model because of the higher number of hidden neurons in FNNLM. Moreover, the mean squared error obtained from FNNPSOGSA is lower compared to FNNLM. This shows the excellent generalization capacity of the FNNPSOGSA in inductor modeling.

To illustrate the performance of FNNPSOGSA, a first order Hilbert fractal inductor is considered with an outer diameter of 180 μm, width of 8 μm, dielectric constant of 4.5, and a frequency range of 0.1 GHz to 30 GHz. The electrical attributes of the inductors L and Q are computed from EM simulations, FNNLM, and FNNPSOGSA. Figure 6 plots the inductance computed in different regions from 0.1 GHz to 30 GHz of the Hilbert fractal inductor by using the EM simulations, FNNLM, and FNNPSOGSA. Similarly, the Q factor variation plots are observed in Figure 7. Figures 6 and 7 suggest the significantly improved generalization capability of FNNPSOGSA compared to FNNLM and also from the Figures 6 and 7 it is observed that FNNPSOGSA preserves the accuracy of EM simulations with reduced CPU times. Table 3 quantifies the effectiveness of the

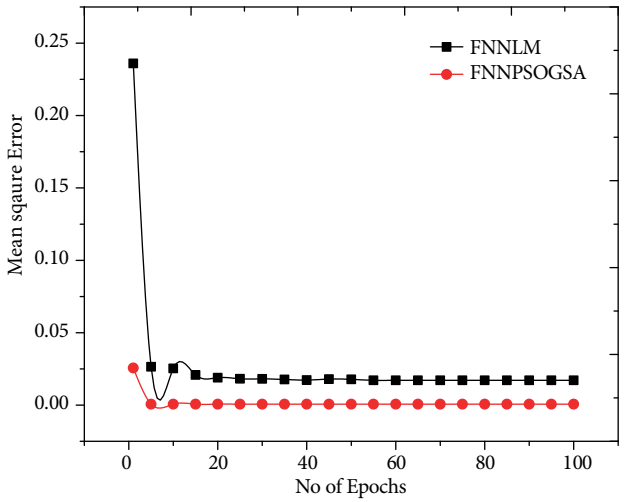


Figure 5. Mean square error plot learning algorithms.

proposed FNNPSOGSA compared to FNNLM. The results show that the proposed FNNPSOGSA requires a single hidden layer with 15 neurons whereas FNNLM requires two hidden layers with 25 neurons in each layer and it reduces the convergence time. The training accuracy of the neural network is measured using relative error and is given by Eq. (15).

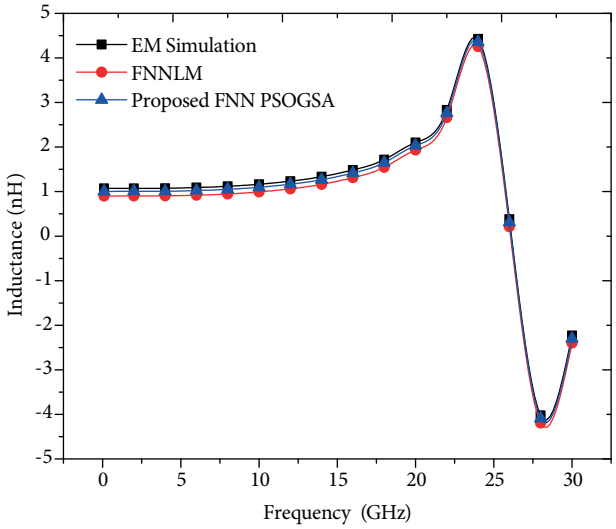


Figure 6. Inductance versus frequency plots for different approaches.

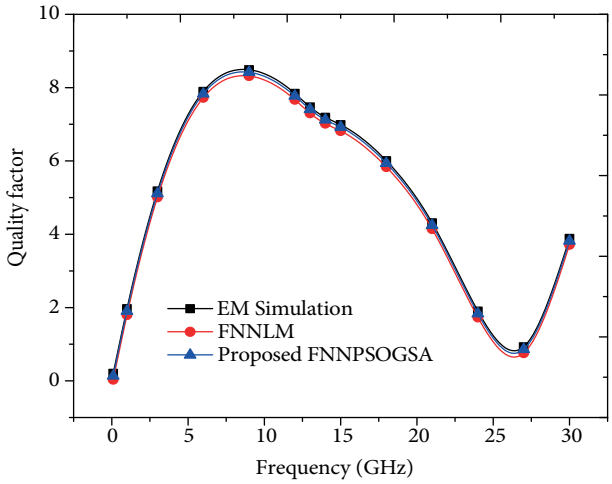


Figure 7. Q-factor versus frequency plots for different approaches.

$$\text{Relative error} = \frac{\sum_{1}^{n} \text{ANN output} - \text{EM simulator output}}{\text{n(EM Simulated output)}} \tag{15}$$

Eq. (15) gives the closeness of the ANN output with EM simulated output. The relative errors are found to be below 5%, which gives the accuracy of the neural network. Also from Table 3 it is observed that FNNPSOGSA attains better training and testing accuracy compared to a conventional neural network with different algorithms.

Table 3. Proposed model accuracy compared with different neural network models.

Training algorithm	Hidden neurons	Training epochs	Epoch time (s)	Type of dataset	%Relative error
Levenberg Marquardt	(50) 4-25-25-2	50	2.5	Training	3.90
				Test	3.70
Levenberg Marquardt	(15) 4-15-2	100	3.2	Training	5.60
				Test	6.50
Gradient decent	(15) 4-15-2	100	3.1	Training	7.60
				Test	8.10
Scaled conjugate gradient	(15) 4-15-2	100	2.9	Training	5.823
				Test	6.94
Quasi-Newton	(15) 4-15-2	100	3.1	Training	5.47
				Test	6.34
FNNPSOGSA	(15) 4-15-2	20	2.2	Training	1.95
				Test	2.05

Similarly, Table 4 shows the average runtime required for FNNPSOGSA to achieve electrical attributes of the Hilbert fractal inductor and also the runtime for HFSS EM simulation of various fractal inductors in the literature. FNNPSOGSA and EM simulations were carried out on a 2.4 GHz Intel Core-i3 machine with RAM of 4 GB. The EM simulations and FNNPSOGSA of the Hilbert fractal were performed for up to 30 GHz frequency range with a step frequency of 0.5 GHz. The EM simulator took a minimum of 30 min to attain lower order Hilbert fractal inductors and the time increases drastically as the iteration order increases. FNNPSOGSA evaluates electrical attributes in less than 1 min. Although initial training data were taken from the EM simulator, it did not adversely affect the synthesis procedure of FNNPSOGSA. The proposed FNNPSOGSA model for the Hilbert fractal inductor is compared with the ANN models for spiral inductor models since ANN models for fractal inductors do not exist in the literature and the complete details are shown in Table 5. From the results shown in Table 5, it is evident that FNNPSOGSA requires fewer hidden layers and hidden neurons and has a relative error almost equal and lower than the existing work in the literature, which shows the efficiency of the proposed FNNPSOGSA in the design of Hilbert fractal inductors.

Table 4. Comparison of CPU time of EM simulations and proposed model.

Inductor type	Type/model	Average run time (s)
Lower order fractal inductor	EM Simulation	1800
Higher order fractal inductor	EM Simulation	6000
Fractal loop inductor [3]	EM Simulation	2200
Series stacked fractal inductor [4]	EM Simulation	8900
Proposed work	FNNPSOGSA	44

5. Conclusion

Efficient modeling of RF on-chip Hilbert fractal inductors using an improved feed-forward network based on PSOGSA has been proposed. The developed neural network does not require any assumptions in terms of

Table 5. Comparison of FNNPSOGSA with existing work in the literature.

Ref.	Inductor type	Hidden neurons	ANN model	No. of epochs	Epoch time (s)	Type of dataset	%Relative error
[19]	Octagonal spiral	(40) 4-20-20-3	MLP (LM)	47	1.89	Training	3.98
						Test	2.962
[20]	Rounded spiral	(12) 4-12-3	MLP (LM)	65	1.01	Training	2.10
						Test	2.90
[21]	Rectangular spiral	(20) 5-20-3	MLP (LM)	68	-	Training	1.97
						Test	2.7
[22]	Rectangular spiral	(16) 7-16-3	GKBNN (LM)	-	-	Training	1.73
						Test	1.84
[24]	Rounded spiral	(8) 6-8-4	KBFDSMN (LM)	12 min		Training	4.48
						Test	4.98
[26]	Rectangular spiral	(40) 4-20-20-6	Physics based (LM)	147	69	Training	1.958
						Test	3.61
		(55) (4-55-6)		150.23	14	Training	4.823
						Test	8.359
Proposed work	Hilbert fractal	(15) 4-15-2	Hybrid PSOGSA	2.2	20	Training	1.95
						Test	2.05

layout or technological parameter values. The proposed FNNPSOGSA is designed with 70-10% fewer hidden neurons and less time is needed to train the network to a desirable accuracy level in a 3 layer perception neural network with much-enhanced learning and generalization capabilities of PSOGSA compared to the traditional feed-forward neural network. In comparison to the EM simulator, the developed ANN model takes very few seconds to generate results. The numerical results also demonstrate that the percentage of relative error has improved to 2.05, which is about 70-10% reduction in relative error when PSOGSA is used as compared to the other conventional algorithms. The results show that the proposed model preserves the accuracy of the full wave EM simulation while it also greatly reduces CPU requirement and is a good alternative for complex fractal inductor design.

References

[1] Lazarus N, Meyer CD, Bedair SS. Fractal inductors. IEEE T Magn 2014; 50: 1-8.

[2] Shoute G, Barlage DW. Fractal loop inductors. IEEE T Magn 2015; 51: 1-8.

[3] Padavala AK, Nistala BR. Fractal series stacked inductor for radio frequency integrated circuit applications. IET Electron Lett 2017; 53: 1387-1388.

[4] Padavala AK, Nistala BR. High inductance fractal inductors for wireless applications. Turk J Elec Eng & Comp Sci 2017; 25: 3868-3880.

[5] Niknejad AM, Meyer RG. Analysis, design, and optimization of spiral inductors and transformers for Si RFICs. IEEE J Solid-St Circ 1998; 33: 1470-1481.

[6] Post JE. Optimizing the design of spiral inductors on silicon. IEEE T Circuits Syst II 2000; 47: 15-17.

[7] Ritter J, Amdt F. Efficient FDTD/matrix-pencil method for the full-wave scattering parameter analysis of waveguiding structures. IEEE T Microw Theory 1996; 44: 2450-2456.

[8] Becks T, Wolff I. Analysis of 3-D metallization structure by a full-wave spectral domain technique. IEEE T Microw Theory 1992; 40: 2219-2227.

[9] Polycarpou AC, Tirkas PA, Balanis CA. The finite-element method for modeling circuits and interconnects for electronic packaging. IEEE T Microw Theory 1997; 45: 1868-1874.

[10] Greenhouse HM. Design of planar rectangular microelectronic inductors. IEEE T Parts, Hybrids, Packag 1974; 10: 101-109.

[11] Grover FW. Inductance Calculations. Phoenix ed. New York, NY, USA: Dover, 1946.

[12] Zhang QJ, Gupta KC, Devabhaktuni VK. Artificial neural network for RF and microwave design-from theory to practice. IEEE T Microw Theory 2003; 51: 1339-1350.

[13] Manaswini P, Sahu RK. Effective classification technique enhanced using Genetic Algorithm for data mining disease in the incumbents to the health centre. IJCSET 2011; 2: 157-166.

[14] Bhabani SN, Subhendu, Pradyumna KP, Mishra RK. Design of high performance low pass filter using neural network and simulated annealing. In: PIERS 2012 Progress in Electromagnetics Research Symposium Proceedings; 27–30 March 2012; KL, Malaysia: PIERS. pp. 968-972.

[15] Young WK, Sean K, Joydeep G, Hao L. Application of artificial neural networks to broadband antenna design based on a parametric frequency model. IEEE T Antennas Propag 2007; 55: 669-674.

[16] Akkaya R, Kulaksız AA, Aydogdu O. DSP implementation of a PV system with GA-MLP-NN based MPPT controller supplying BLDC motor drive. Elsevier Energy Convers Manage 2007; 48: 210-218.

[17] Alma YA, Ricalde LJ, Chiara S, Rancesca O. Neural model with particle swarm optimization Kalman learning for forecasting in smart grids. Hindawi Math Probl Eng 2013; 2013: 1-9.

[18] Erredir C, Mohamed LR, Ammari H, Bouarroudj E. Design of waveguide structures using improved neural networks. J Microw Optoelectron Electromagn Appl 2017; 16: 900-907.

[19] Sushanta KM, Shamik S, Amit P. ANN and PSO based synthesis of on-chip spiral inductors for RFICs. IEEE T Comput Aid D 2008; 27: 188-210.

[20] Mehdi A, Kaabi H, Kavian YS. Optimization of on-chip spiral inductors using coupled NN and TLBO algorithms for low-loss lumped-element CMOS power dividers. J Intell Fuzzy Syst 2016; 30: 2029-2036.

[21] Rakhesh SK, Karthikeyan SS, Krishna MV. ANN for fast and accurate design of spiral inductors. In: National Conference on Communications; 16–18 January 2009; Guwahati, India: NCC. pp. 54-58.

[22] Yazi C, Wang G. Efficient modelling of RF CMOS spiral inductors using the generalized knowledge-based neural network. International Journal of Analog Integrated Circuits and Signal Processing 2007; 52: 71-77.

[23] Han B, Shi X, Li J. Broad band scalable compact circuit model for on-chip spiral inductors by neural network. Int J Numer Model 2016; 30: 1-8.

[24] Liu XC, Wang G, Deng D, Liu F, Tu Z. A new model of on-chip inductors on ferrite film using KB-FDSMN neural network. Int J RF Microw C E 2010; 20: 399-407.

[25] Yazi C, Wang G, Pavan G, Zhang QJ. Parametric modeling of microwave passive components using combined neural networks and transfer functions in the time and frequency. Int J RF Microw C E 2013; 23: 20-33.

[26] Tao L, Zhang W, Zhiping Y. Modeling of spiral inductors using artificial neural network. In: IEEE International Joint Conference on Neural Networks Proceedings; 3 July–4 August 2005; Montreal, Canada. IEEE. pp. 2353-2358.

[27] Sagan H. Space-Filling Curves. New York, NY, USA: Springer-Verlag, 1994.

[28] Kiranyaz S, Ince T, Yildirim A, Gabbouj M. Evolutionary artificial neural networks by multi-dimensional particle swarm. In: 19th International Conference on Pattern Recognition; 8–11 December 2008; Tampa, FL, USA: IEEE. pp. 1-4.