Artificial bee colony algorithm for small signal model parameter extraction of MESFET

Samrat L. Sabat [a,*], Siba K. Udgata [b], Ajith Abraham [c]

[a] School of Physics, University of Hyderabad, Hyderabad 500046, India
[b] Department of Computer and Information Sciences, University of Hyderabad, Hyderabad 500046, India
[c] Machine Intelligence Research Labs, Scientific Network for Innovation and Research Excellence, P.O. Box 2259, Auburn, WA 98071-2259, USA

ARTICLE INFO

ABSTRACT

This paper presents an application of swarm intelligence technique namely artificial bee colony (ABC) to extract the small signal equivalent circuit model parameters of GaAs metal extended semiconductor field effect transistor (MESFET) device and compares its performance with particle swarm optimization (PSO) algorithm. Parameter extraction in MESFET process involves minimizing the error, which is measured as the difference between modeled and measured S parameter over a broad frequency range. This error surface is viewed as a multi-modal error surface and robust optimization algorithms are required to solve this kind of problem. This paper proposes an ABC algorithm that simulates the foraging behavior of honey bee swarm for model parameter extraction. The performance comparison of both the algorithms (ABC and PSO) are compared with respect to computational time and the quality of solutions (QoS). The simulation results illustrate that these techniques extract accurately the 16—element small signal model parameters of MESFET. The efficiency of this approach is demonstrated by a good fit between the measured and modeled S-parameter data over a frequency range of 0.5–25 GHz.

© 2010 Elsevier Ltd. All rights reserved.

1. Introduction

Small signal model parameter extraction of MESFET involves extraction of extrinsic and intrinsic model element values (Lin and Kompa, 1994) by minimizing the difference between modeled and measured S-parameter over a broad range of frequencies. In the recent past, different techniques (Yaser, 2000; Van Niekerk et al., 2000) have been reported in the literature for extracting the model parameters of MESFET. These techniques are normally based on either analytical or numerical optimization techniques. Although analytical methods provide faster solution, the quality of solution (QoS) is normally poor. To improve QoS, methods based on numerical optimization are being increasingly used for parameter extraction. Numerical optimization techniques are either gradient-based or gradient-free. Possibility of having a multimodal error surface is an important extraction challenge in parameter extraction problem. In order to find a quality solution, an extraction algorithm that can achieve the global minima in multimodal error surface is required. However, the conventional gradient based approach that are used in past, can easily be trapped in local minima. Many researchers have proposed global

optimization techniques like genetic algorithm (GA) (Gao, 2001; Watts et al., 1999; Menozzi et al., 1996) to extract the small signal model parameters of the MESFET.

Since a typical small signal parameter extraction problem has a multimodal error surface and it involves a large set (i.e. 16 number of elements), conventional gradient based techniques fails to provide QoS. In some cases, GA cannot guarantee global solution due to the diversity of population (Leung et al., 1997). Swarm intelligence has become a research interest to different domain of researchers in recent years. These algorithms simulate the food foraging behavior of a flock of birds or swarm of bees. Particle swarm optimization and its variants have been introduced for solving numerical optimization problems and successfully applied to solve many real world problems (Eberhart and Kenedy, 1995b; Sabat et al., 2009, 2010). PSO algorithm is a population based stochastic optimization technique and suitable for optimizing nonlinear multimodal error function. Motivated by the foraging behavior of honeybees, researchers have (Riley et al., 2005; Karaboga, 2009) initially proposed artificial bee colony (ABC) algorithm for solving various optimization problems. ABC is a relatively new population-based meta-heuristic approach and is further improved by Karaboga and Basturk (2008). This algorithm is easy to implement and found to be robust. Some recent results illustrate that artificial bee colony (ABC) algorithms outperforms basic PSO algorithm in terms of QoS (Karaboga and Basturk, 2008). The PSO and ABC algorithms are population based

---

* Corresponding author. Tel.: +91 40 23134326.
E-mail addresses: slssp@uohyd.ernet.in (S.L. Sabat), udgatacs@uohyd.ernet.in (S.K. Udgata), ajith.abraham@ieee.org (A. Abraham).
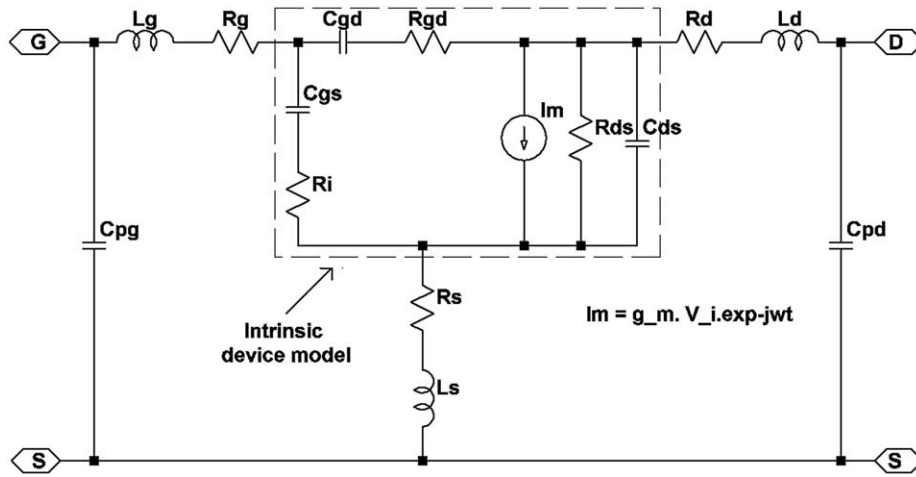
Fig. 1. Schematic diagram of MESFET small signal model.

evolutional meta-heuristic optimization algorithms that avoids trapping of solution in local minima. The objective of this paper is to use ABC algorithm for extracting the small signal parameters and to compare relative performances in terms of computational cost and QoS with that of basic PSO algorithm.

Fig. 1 shows a 16 element small signal equivalent circuit adopted for parameter extraction. It has eight extrinsic and eight intrinsic parameters. Extrinsic parameters are bias independent whereas intrinsic parameters are bias dependent. The methodology for extracting these parameters essentially involves minimization of the difference between measured and modeled S-parameter values under dc bias conditions.

Rest of the paper is organized as follows: Section 2 presents a brief description of the model parameter extraction strategies and formulation of the problem. Brief descriptions of PSO and ABC algorithms are provided in Sections 3 and 4, respectively. Section 5 presents the simulation results. Conclusions are drawn in Section 6.

2. Parameter extraction problem in MESFET

The main objective of small signal model parameter extraction problem is to minimize the difference between measured and simulated S-parameter at different bias points.

The fitness function is defined as

$$F(x) = \sum_{i=1}^{N} [\alpha_i(x)]^2 \tag{1}$$

where

$$\alpha_i(x) = \sum_{t=1}^{M} \sum_{j=1}^{2} \sum_{k=1}^{2} \frac{1}{\sigma_{jk}(t, w_i)} |S_{jk}(t, w_i) - \hat{S}_{jk}(t, w_i)|^2 \tag{2}$$

and

$$\sigma_{jk}(t, w_i) = |\hat{S}_{jk}(t, w_i)|_{\max} \tag{3}$$

Eq. (1) is the sum of errors at all the frequencies and Eq. (2) is the modeling error at all the bias points for all the four S-parameters at i th frequency w_i. The measured and modeled S-parameters of the MESFET at i th frequency and t th bias point are $S_{jk}(t, w_i)$ and $\hat{S}_{jk}(t, w_i)$, respectively. M is the number of bias points and N is the number of frequency points used in measurement. j and k are the indices of the four S-parameters. $\sigma_{jk}(t)$ is a normalization constant at t th bias point. x is the vector that has all the bias independent and bias dependent parameters need to be extracted.

Table 1
Model element/subfunction used for 16 element MESFET model.

No.	Model element	Bias dependent	subfunction to be minimized
1	C_{gs}, R_i	Yes	S_{11} at specific bias point
2	C_{gd}	Yes	S_{12} at specific bias point
3	g_m, τ	Yes	S_{21} at specific bias point
4	C_{ds}, R_{ds}	Yes	S_{22} at specific bias point
5	C_{pg}, R_g, L_g	No	S_{11} at all the bias point
6	C_{pd}, R_d, L_d	No	S_{22} at all the bias point
7	R_s, L_s	No	S_{12} at all the bias point

This fitness function (see Eq. (1)) can be divided into two parts, one for extracting bias independent model parameters and the other for extracting bias dependent model parameters. Bias dependent model parameters are associated with a modeling error due to the associated S-parameters at specified bias point as in Table 1. The fitness sub-function is given by

$$f_{ext}(w) = \sum_{i=1}^{N} |S_{jk}(t, w_i) - \hat{S}_{jk}(t, w_i)|^2 \tag{4}$$

similarly for obtaining bias independent model parameters, the fitness sub-function is defined as

$$f_{int}(x) = \sum_{i=1}^{N} \left(\sum_{t=1}^{M} \frac{1}{\sigma_{jk}(t, w_i)} |S_{jk}(t, w_i) - \hat{S}_{jk}(t, w_i)| \right)^2 \tag{5}$$

Table 1 presents different model parameters and their dependence with different S-parameters. For analysis purposes, the small signal equivalent circuit elements of MESFET, see Fig. 1, are grouped into two set of parameters: extrinsic and intrinsic. The extrinsic set consist of bias independent elements associated with the leads and contacts to the device such as $C_{pg}, C_{pd}, L_g, L_d, L_s, R_g, R_d,$ and R_s. The intrinsic set consists of bias dependent parameters such as $C_{gs}, R_i, C_{gd}, R_{gd}, C_{ds}, R_{ds}$ and g_m, τ.

The technique employed for parameter extraction is implemented in two stages. In the first stage, the extrinsic parameters are extracted from measured S-parameters under cold bias conditions i.e., $V_{ds}=0$ V and $V_{gs} < V_p$, where V_p is pinch-off voltage. Intrinsic parameters are extracted from measured S-parameters under hot bias condition ($V_{ds} > 0$ V and $V_{gs} < 0$ V) using suitable de-embedding technique in the second stage. Parameters to be extracted are initialized, each within carefully chosen and well defined range at the beginning of the extraction algorithm.

Different S parameter values are optimized for obtaining the different set of model elements of the equivalent circuit (Lin and

Kompa, 1994). The parameter extraction procedure is summarized below:

1. Measure S-parameter data in cold bias condition.
2. Optimize S_{11} to extract L_g, R_g and C_{pg}; optimize S_{22} to extract L_d, R_d, C_{pd} and S_{12} to extract L_s and R_s.
3. Measure S-parameter data in hot bias condition
4. Subtract the effect of extrinsic parasitics from the measured S-parameter data by the de-embedding technique (Lin and Kompa, 1994).
5. Optimize S_{11} to extract C_{gs}, R_i; Optimize S_{12} to extract C_{gd}, R_{gd}; similarly optimize S_{21} to extract g_m and S_{22} to extract C_{ds}, R_{ds}.

3. Particle swarm optimization (PSO) algorithm

This section provides a brief introduction of the PSO technique and its application to model parameter extraction of MESFET.

PSO algorithm is a stochastic population based algorithm proposed by Eberhart and Kenedy (1995a). This algorithm is inspired by the social behavior of bird flocking and fish schooling. The PSO algorithm is simple, robust and easy to implement. In PSO, a set of particles (NP) in the swarm is defined. Each particle is characterized by its position and velocity and constitutes a potential solution in the solution space. The number of parameters to be optimized determines the dimension of the problem. Each particle is evaluated based on the fitness function to be optimized. In the beginning, all the particles are initialized randomly in the D-dimension search space within the predefined range. The position and velocity of i th particle ($i = 1, 2, \ldots, NP$) in the D th dimension are represented as $X_i = (x_{i1}, x_{i2}, x_{i3}, \ldots x_{iD})$ and $V_i = (v_{i1}, v_{i2}, v_{i3}, \ldots v_{iD})$, respectively. Each particle updates its position and velocity based on its own best position, ($pbest$) as well as the best position of the entire swarm ($gbest$).

While in the search process, each particle remembers its previous best solution, and if the current solution is better than the previous best solution, the position corresponding to current solution becomes the $pbest$; otherwise it remains unchanged. The best among all $pbest$ solution is the $gbest$ solution and the position corresponding to $gbest$ value are the potential solutions of the problem. Each particle updates its velocity and position according to its flying trajectory by the following:

$$V_{i,d}^{t+1} = w^t * V_{i,d}^t + c_1 * rand_1 * (pbest_{i,d}^t - X_{i,d}^t) + c_2^t * rand_2 * (gbest_d^t - X_{i,d}^t)$$

(6)

$$X_{i,d}^{t+1} = X_{i,d}^t + V_{i,d}^{t+1}$$

(7)

where w represents inertia weight, c_1 and c_2 are learning factors which determine the relative influence of cognitive and social components, respectively. In this paper, the values of w, c_1 and c_2 are linearly decreased (adapted) with iteration (Shi and Eberhart, 1998; Ratnaweera et al., 2004; Sabat and Ali, 2008). $rand_1$ and $rand_2$ are independent random numbers uniformly distributed in the range [0,1]. $V_{i,d}^t$, $X_{i,d}^t$ and $pbest_{i,d}^t$ are the velocity, position and the personal best of i th particle in d th dimension for the t th iteration, respectively. The $gbest_d^t$ is the d th dimension of best particle in the swarm for the t th iteration.

4. Artificial bee colony algorithm

The foraging bees are classified into three categories; employed bees, onlookers and scout bees. All bees that are currently exploiting a food source are known as *employed*. The *employed* bees exploit the food source (NFS) and they carry the information

about food source back to the hive and share this information with onlooker bees. *Onlookers* bees are waiting in the hive for the information to be shared by the employed bees about their discovered food sources and *scouts* bees will always be searching for new food sources near the hive. Employed bees share information about food sources by dancing in the designated dance area inside the hive. The nature of dance is proportional to the nectar content of food source just exploited by the dancing bee. Onlooker bees watch the dance and choose a food source according to the probability proportional to the quality of that food source. Therefore, good food sources attract more onlooker bees compared to bad ones. Whenever a food source is exploited fully, all the employed bees associated with it abandon the food source, and become scout. Scout bees can be visualized as performing the job of exploration, whereas employed and onlooker bees can be visualized as performing the job of exploitation.

In the ABC algorithm (Karaboga and Basturk, 2008), each food source is a possible solution for the problem under consideration and the nectar amount of a food source represents the quality of the solution represented by the fitness value. The number of food sources is same as the number of employed bees and there is exactly one employed bee for every food source.

This algorithm starts by associating all employed bees with randomly generated food sources (*solution*). In each iteration, every employed bee determines a food source in the neighborhood of its current food source and evaluates its nectar amount (*fitness*). The i th food source position is represented as $X_i = (x_{i1}, x_{i2}, \ldots, x_{id})$. $F(X_i)$ refers to the nectar amount of the food source located at X_i. After watching the dancing of employed bees, an onlooker bee goes to the region of food source at X_i by the probability p_i defined as

$$p_i = \frac{F(X_i)}{\sum_{k=1}^{S} F(X_k)}$$

(8)

where S is total number of food sources. The onlooker finds a neighborhood food source in the vicinity of X_i by using

$$X_i(t+1) = X_i(t) + \delta_{ij} * u$$

(9)

where δ_{ij} is the neighborhood patch size for j th dimension of i th food source defined as

$$\delta_{ij} = x_{ij} - x_{kj}$$

(10)

where k is a random number $\in (1, 2, \ldots, S)$ and $k \neq i$, u is random uniform variate $\in [-1, 1]$.

If its new fitness value is better than the best fitness value achieved so far, then the bee moves to this new food source abandoning the old one, otherwise it remains in its old food source. When all employed bees have finished this process, they share the fitness information with the onlookers, each of which selects a food source according to probability given in Eq. (8). With this scheme, good food sources will get more onlookers than the bad ones. Each bee will search for better food source around neighborhood patch for a certain number of cycle (*limit*), and if the fitness value will not improve then that bee becomes scout bee.

4.1. Pseudocode for ABC algorithm

```
1: Initialize
2: REPEAT.
3: Move the employed bees onto their food source and
   evaluate the fitness
4: Move the onlookers onto the food source and evaluate their
   fitness
5: Move the scouts for searching new food source
6: Memorize the best food source found so far
7: UNTIL (termination criteria satisfied)
```

5. Experimental results and discussions

S-parameter data of a fabricated MESFET with gate length of 0.7 μm and gate width of 600 μm (4 × 150) are measured in the laboratory using network analyzer in the frequency range from 0.5 to 25 GHz. Bias point of $V_{gs} = -1.0$ V and $V_{ds} = 4.25$ V is fixed for the analysis. The bias value is chosen such that the device can be used for amplifier purpose. Small signal model parameter values are obtained by minimizing the difference between measured and modeled S-parameters using basic PSO and ABC algorithm. The algorithms are simulated in an Intel Core 2 duo processor with 2.0 GHz frequency and 1.0 GB RAM using MATLAB 2007a. The algorithmic parameters are tabulated in Table 2.

The search range for model parameters are tabulated in Table 3. Each algorithms are simulated 50 times. Performance indicator parameters such as minimum, maximum, mean and standard deviation values of extracted model parameters are tabulated in Tables 4–7.

Tables 4 and 5 give the summary of the extracted extrinsic elements using basic PSO, and ABC algorithms, respectively. Tables 6 and 7 show the summary of extracted intrinsic element values using basic PSO and ABC algorithms, respectively.

It is clearly evident from Tables 4 to 7 that ABC algorithm performance is better compared to PSO algorithm in terms of both performance evaluation parameters namely standard deviation and relative error between measured and modeled data.

Figs. 2–5 compares the modeled and measured S-parameters for the bias point $V_{gs} = -1.0$ V; $V_{ds} = 4.25$ V. The comparison shows that in ABC algorithm, the parameters are within small standard deviation, and relative error for all the S-parameters are small compared to PSO algorithm. Table 9 provides a comparative results of the amplitude and phase relative error between measured and modeled S-parameters. Convergence time and

Table 2
Algorithmic parameters.

Parameters	Basic PSO	ABC
No. of particles/bees	20	20
No. of iterations	1000	1000
c_1	Varies with iteration	Not required
c_2	Varies with iteration	Not required
w	Varies with iteration	Not required
No. of bees	Not required	20
No. of food sources	Not required	10
Limit number	Not required	30

Table 3
Search range of extracted extrinsic and intrinsic parameters.

Parameter	Search range (min)	Search range (max)
$L_g(pH)$	10	100
$L_d(pH)$	10	100
$L_s(pH)$	10	100
$R_g(\Omega)$	1	4
$R_d(\Omega)$	1	4
$R_s(\Omega)$	1	4
$C_{pg}(fF)$	12	35
$C_{pd}(fF)$	25	75
$C_{gs}(fF)$	200	400
$R_i(\Omega)$	0.1	5.0
$C_{gd}(fF)$	40	80
$R_{gd}(\Omega)$	12	20
$C_{ds}(pF)$	40	80
$R_{ds}(\Omega)$	110	170
$g_m(S)$	0.02	0.06
$\tau(ps)$	1.5	5.5

Table 4
List of extracted extrinsic parameters using PSO.

Parameter	Minimum	Maximum	Mean	Best	Std. deviation
$L_g(pH)$	10.08	19.11	13.98	16.53	3.54
$L_d(pH)$	10.06	22.42	13.25	10.25	4.46
$L_s(pH)$	21.57	45.83	37.55	21.75	7.73
$R_g(\Omega)$	1.00	1.45	1.06	1.01	0.13
$R_d(\Omega)$	1.09	2.75	1.9	2.23	0.60
$R_s(\Omega)$	1.45	3.92	2.78	1.45	0.85
$C_{pg}(fF)$	33.1	34.98	34.51	34.04	0.59
$C_{pd}(fF)$	74.12	74.99	74.75	74.65	0.27

Table 5
List of extracted extrinsic parameters using ABC.

Parameter	Minimum	Maximum	Mean	Best	Std. deviation
$L_g(pH)$	13.07	24.07	18.66	13.11	4.11
$L_d(pH)$	15.91	39.77	26.09	23.77	9.1
$L_s(pH)$	10.00	10.04	10.01	10.04	0.01
$R_g(\Omega)$	1.30	1.45	1.39	1.32	0.06
$R_d(\Omega)$	2.03	3.08	2.44	2.31	0.36
$R_s(\Omega)$	1.00	1.00	1.00	1.00	0.00
$C_{pg}(fF)$	33.1	34.92	34.1	34.5	0.01
$C_{pd}(fF)$	74.1	74.92	74.81	74.8	0.01

Table 6
List of extracted intrinsic parameters using PSO.

Parameter	Minimum	Maximum	Mean	Best	Std. deviation
$C_{gs}(pF)$	0.39	0.39	0.39	0.39	0.00
$R_i(\Omega)$	0.10	0.119	0.104	0.10	0.005
$C_{gd}(fF)$	49.92	76.86	64.25	58.74	7.71
$R_{gd}(\Omega)$	16.86	19.94	19.23	16.96	1.14
$C_{ds}(fF)$	52.46	78.15	69.86	55.22	8.57
$R_{ds}(\Omega)$	142.52	144.24	142.81	144.00	0.61
$g_m(S)$	0.055	0.059	0.058	0.058	0.00
$\tau(ps)$	1.97	2.74	2.29	2.74	0.21

Table 7
List of extracted intrinsic parameters using ABC.

Parameter	Minimum	Maximum	Mean	Best	Std. deviation
$C_{gs}(pF)$	0.3	0.377	0.32	0.354	0.01
$R_i(\Omega)$	1.23	4.18	2.68	4.01	1.10
$C_{gd}(fF)$	56.67	60.55	58.12	60.22	1.18
$R_{gd}(\Omega)$	10.00	10.00	10.00	10.00	0.00
$C_{ds}(fF)$	40.00	44.43	41.62	41.10	1.72
$R_{ds}(\Omega)$	148.92	151.94	150.17	149.32	1.1
$g_m(S)$	0.057	0.058	0.057	0.058	0.00
$\tau(ps)$	1.73	4.01	2.83	3.21	0.7

number of iterations required for PSO and ABC algorithms are tabulated in Table 8. The tabular results of Table 9 indicates the superiority of ABC algorithm in terms of accuracy. Both Tables 8 and 9 indicate the better performance of ABC algorithm in terms of accuracy and convergence time compared to PSO.

The main advantages of the proposed technique is that it is not sensitive to initial parameter values and also not affected by increasing the dimension of problem. Empirical results illustrate
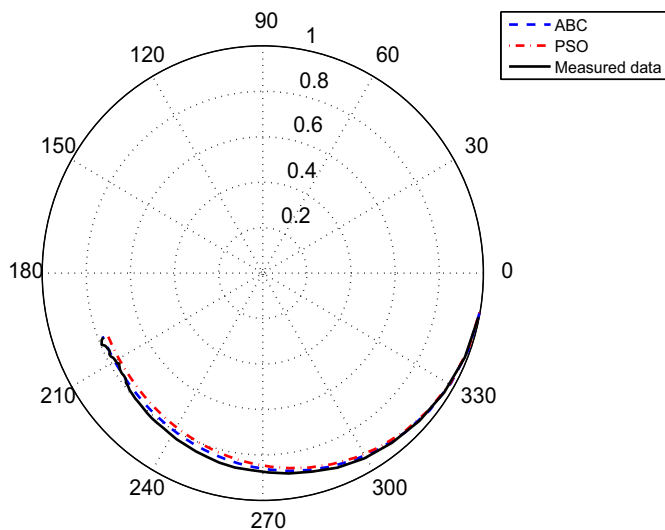
Fig. 2. Comparison of the measured and modeled characteristics of S_{11}-parameters as a function of operating frequency range 500 MHz–25.0 GHz with the step of 500 MHz.
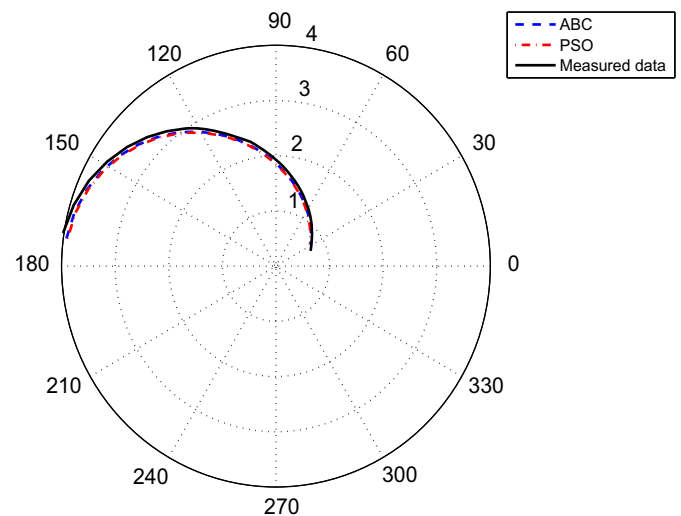

Fig. 4. Comparison of the measured and modeled characteristics of S_{21}-parameters as a function of operating frequency range 500 MHz–25.0 GHz with the step of 500 MHz.
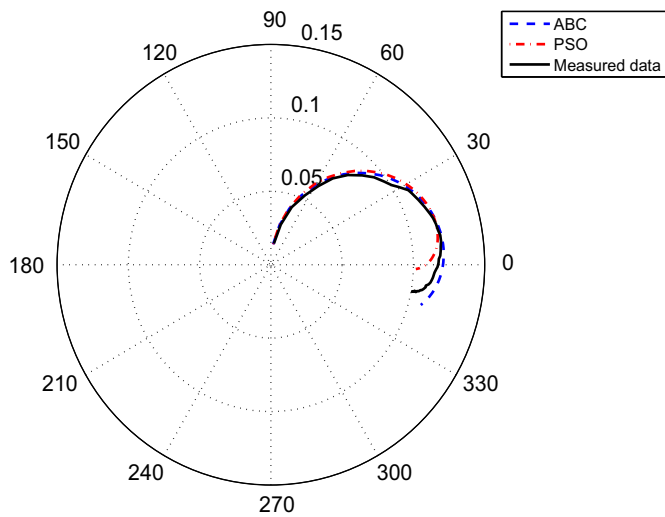

Fig. 3. Comparison of the measured and modeled characteristics of S_{12}-parameters as a function of operating frequency range 500 MHz–25.0 GHz with the step of 500 MHz.
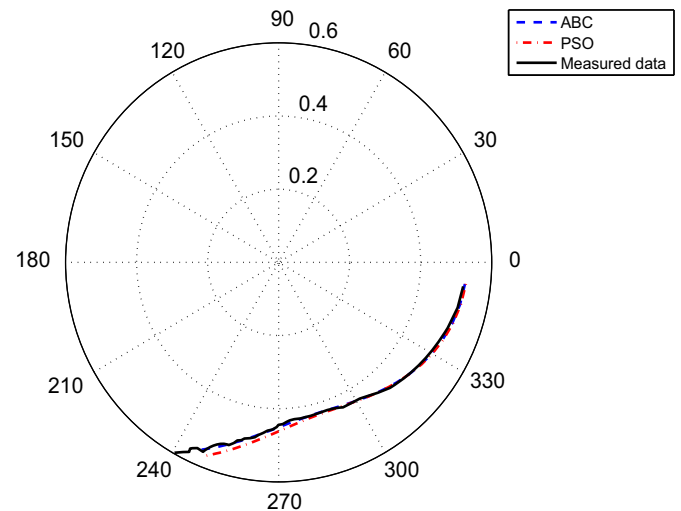

Fig. 5. Comparison of the measured and modeled characteristics of S_{22}-parameters as a function of operating frequency range 500 MHz–25.0 GHz with the step of 500 MHz.

that the proposed algorithm possesses high accuracy and also provides physical significant values of the extracted parameters.

6. Conclusions

In this paper, the artificial bee colony (ABC) algorithm, which is a new, simple and robust optimization algorithm, is used to extract small signal model parameters of a fabricated GaAs MESFET from measured *S*-parameter data. The performance of the proposed algorithm is compared with the particle swarm optimization (PSO) algorithm. These algorithms are able to successively extract the small signal model parameters of MESFET. The results reveal that ABC is more robust and has less relative error between the measured and modeled *S*-parameters compared to PSO algorithm. ABC algorithms also converges faster compared to PSO algorithm for this problem. Moreover, in these swarm intelligence strategy no user intervention is required for

Table 8
Convergence time comparison of algorithms.

Algorithm	Conv. time (iterations, secs)
PSO	1000, 180
ABC	200, 145

Table 9
Relative error performance comparison of different algorithms.

Parameter	PSO	ABC
$\lvert S_{11} \rvert$	0.255	0.011
$\langle S_{11}$	2.133	0.015
$\lvert S_{12} \rvert$	0.023	0.058
$\langle S_{12}$	1.0	0.65
$\lvert S_{21} \rvert$	0.47	0.011
$\langle S_{21}$	1.0	0.232
$\lvert S_{22} \rvert$	0.245	0.042
$\langle S_{22}$	2.50	0.029

model parameter extraction unlike gradient descent approach. In future, we will study the possible variations of ABC and PSO to further improve the performance.

Acknowledgments

References

Eberhart, R., Kenedy, J., 1995a. A new optimizer using particle swarm theory. In: Proceedings of 6th International Symposium on Micro Machine and Human Science (MHS), Cape Cod, MA, November, pp. 39–43.

Eberhart, R., Kenedy, J., 1995b. Particle swarm optimization. In: Proceedings of IEEE International Conference on Neural Networks, Piscataway, NJ, November, pp. 1114–1121.

Gao, Y., 2001. Genetic algorithm based parameter extraction for power gaas mesfet. In: Proceedings of the Fifth International Conference on Telecommunications in Modern Satellite, Cable and Broadcasting Service, USA, November, pp. 317–318.

Karaboga, D., Basturk, B., 2008. On the performance of artificial bee colony (ABC) algorithm. Appl. Soft Comput. 8 (3), 687–697.

Karaboga, N., 2009. A new design method based on artificial bee colony algorithm for digital IIR filters. J. Franklin Institute 346 (4), 328–348.

Leung, Y., Gao, Y., Xu, Z., 1997. Degree of population diversity—a perspective on premature convergence in genetic algorithms and its Markov chain analysis. IEEE Trans. Neural Networks 8, 1165–1176.

Lin, F., Kompa, G., 1994. FET model parameter extraction based on optimization with multiplane data-fitting and bidirectional search—a new concept. IEEE Trans. Microwave Theory 42 (7), 1114–1121.

Menozzi, R., Piazzi, A., Contini, F., 1996. Small signal modeling for microwave FET linear circuits based on a genetic algorithm. IEEE Trans. Circuits Syst. I Fundam. Theory Appl. 43 (10), 839–847.

Ratnaweera, A., Halgamuge, S.K., Watson, H.C., 2004. Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients. IEEE Trans. Evol. Comput. 8 (3), 240–255.

Riley, J., Greggers, U., Smith, A.D., Reynolds, D., Menzel, R., 2005. The flight paths of honeybees recruited by the waggle dance. Nature 435, 205–207.

Sabat, S.L., Ali, L., 2008. The hyperspherical acceleration effect particle swarm optimizer. Appl. Soft Comput. 9 (13), 906–917.

Sabat, S.L., Coelhol, L.S., Abraham, A., 2009. MESFET DC model parameter extraction using quantum particle warm optimization. Microelectron. Reliab. 49 (6), 660–666.

Sabat, S.L., Udgata, S.K., Murthy, K.P.N., 2010. MESFET small signal model parameter extraction using quantum particle swarm optimization. Microelectron. Reliab. 50 (2), 199–206.

Shi, Y., Eberhart, R.C., 1998. Parameter selection in particle swarm optimization. In: EP '98: Proceedings of the 7th International Conference on Evolutionary Programming VII, London, UK, pp. 591–600.

Van Niekerk, C., Meyer, P., Schreurs, D., Winson, P., 2000. A robust integrated multibias parameter extraction method for MESFET and HEMT models. IEEE Trans. Microwave Theory 48 (5), 777–786.

Watts, J., Bittner, C., Heaberlin, D., Huffmen, J., 1999. Extraction of compact model parameters for ULI MOSFETs using a genetic algorithm. In: Proceedings of Second International Conference on Modeling and Simulation of Microsystems, Cambridge, MA, November, pp. 176–179.

Yaser, K., 2000. Systematic optimization technique for mesfet modeling. Ph.D. Thesis, Virginia Polytechnic Institute and State University, Blacksburg, Virginia.