

# Test Frequency Selection of Analog Circuit Based on Discrete Particle Swarm Algorithm

Zheng Zhigang

Army Aviation Research Institute  
Beijing, China  
zzg1974@126.com

Wu Liang

Control Engineer Department  
Naval Army Aviation University  
Yantai, China  
wulianghefei@163.com

**Abstract**—A method of optimal test frequency selection for analog circuit fault diagnosis is proposed. The method simulates the circuit faulty states only in the bands with highest fault sensitivity and that reduces part of simulation cost. Test frequency selection is considered as a multi-objective optimization problem, discrete particle swarm algorithm is used as a new approach to get optimal test frequency sets. An example shows that the proposed method is effective in frequency selection and works more efficiently compared with genetic algorithm.

**Keywords**- Test frequency selection; Discrete particle swarm algorithm; Analog circuit; Fitness function

## I. INTRODUCTION

Analog circuit automatic test and fault diagnosis is becoming more important with the rapid growth of mixed analog/digital circuit. Present fault diagnosis methods are generally divided into two categories<sup>[1]</sup>: SBT(Simulation Before Test) and SAT(Simulation After Test); SBT method is mainly based on fault dictionary technique, and SAT method solves the component parameter value via linear or nonlinear equations to locate the faulty components.

Because now analog circuit is becoming more integrated, circuit accessible test nodes are few, so in order to acquire more information about the components' faults in a circuit under test, multi-frequency test is a better choice. Sinusoidal signals of various frequencies are input to the circuit and the response voltage are measured at output nodes, it shows that appropriate test frequencies are helpful to fault diagnosis. R. Saeks<sup>[2]</sup> select test frequencies simply by maximize the rank of Jacobian matrix of fault diagnosis equation, S.Mir<sup>[3]</sup> use maximum sensitivity analysis in the frequency domain. Recently artificial intelligence is introduced in frequency selection, Francesco Grasso<sup>[4]</sup> construct sensitivity matrix with different components, define index signature to optimal frequency selection. Nagi<sup>[5]</sup> use heuristic search method select frequency, but that can't ensure minimal test set. P.Jantos<sup>[6]</sup> use genetic algorithm to select optimal frequency, but need simulation in entire band before test.

The method proposed in this paper has improvements in two aspects: one is reduction of simulation cost. The method needn't simulate in entire circuit work band before test, it only selects the bands in which the fault has maximum sensitivity to simulate; The other improvement is the utilization of discrete particle swarm algorithm in test

frequency selection, the algorithm select the minimal test frequency set by multi-objective optimization and works more efficiently than other algorithms.

## II. THE PRINCIPLE OF TEST FREQUENCY SELECTION

When an analog circuit is tested using multi-frequency method, the main idea of test frequency selection is just to find the frequencies at which the component's faults can be more obvious than other frequencies. The whole test frequency set of the CUT(Circuit Under Test) is assumed to be  $F$ . Test frequency selection process is to find the subset  $F_{sub}$  of set  $F$ , and the subset  $F_{sub}$  is the minimal complete test set. So the size of set  $F_{sub}$  is the minimal, although  $F_{sub}$  consists of less frequency elements than set  $F$ , the frequency elements in set  $F_{sub}$  have the same fault diagnosability as elements in set  $F$ , that is to say,  $F_{sub}$  is the minimal set of test frequencies which guarantees maximum fault coverage and maximal fault diagnosability at the same time. So the test frequency selection is substantially a multi-objective optimization problem, and we can get the best test frequencies by some optimization algorithms such as genetic algorithm and so on, in this paper we try to select the frequencies by using particle swarm algorithm. In the following paragraphs we will introduce the particle swarm algorithm in detail.

Particle swarm optimization is a novel population-based searching technique as an alternative to genetic algorithm, it was proposed by Kennedy J and Eberhart R C<sup>[7]</sup> in 1995. The particle swarm algorithm adopt a velocity-position model, the algorithm initial a swarm of random particles at beginning time, every particles has its own velocity which determine its moving directions and distance, the particle search the best position of its own and among populations, and is evaluated by the fitness function. Suppose in a searching space of  $D$  dimensions, there is a populations consisting of  $N$  particles, among which the position variable of the  $i$ th particle at moment  $k$   $X_i(k)$  is:

$$X_i(k) = (x_{i1}, x_{i2}, \dots, x_{iD}) \quad (1)$$

And its velocity variable  $v_i(k)$  at moment  $k$  is:

$$v_i(k) = (v_{i1}, v_{i2}, \dots, v_{iD}) \quad (2)$$

At the next moment  $k+1$ , the  $i$ th particle's position and velocity are updated by the following formula:

$$v_i(k+1) = v_i(k) + c_1 r_1 (P_i - X_i(k)) + c_2 r_2 (P_g - X_i(k)) \quad (3)$$

$$X_i(k+1) = X_i(k) + v_i(k+1)$$

Where  $c_1$  and  $c_2$  are accelerating constants,  $r_1$  and  $r_2$  are random constants in interval  $[0,1]$ ,  $P_i$  represents the ever best position of the  $i$ th particle,  $P_g$  represents the global best position of all the particles in the population. The position fitness of a particle is evaluated by a fitness function, all the particles in population moves towards the position with highest fitness value. Although the particle swarm algorithm converge quickly, it often falls into local extreme, so to avoid this, Shi Y<sup>[8]</sup> introduced a linear inertia variable  $w$  into the update formula, the improved velocity updating formula is:

$$v_i(k+1) = w(k)v_i(k) + c_1r_1(P_i - X_i(k)) + c_2r_2(P_g - X_i(k)) \quad (4)$$

Variable  $w$  describe the impact of previous moment velocity on current velocity, and the change of its value can control its global searching ability, in this paper we choose  $w$  a linearly changing value

$$w(k+1) = (w_{ini} - w_{end})(K_{max} - k) / K_{max} + w_{end} \quad (5)$$

Where  $w_{ini}$  is the initial inertia value,  $w_{end}$  is the end inertia value,  $K_{max}$  is the maximum iteration number.

The particle swarm algorithm has been successfully used in many problems, but many optimization problem is in discrete space, based on continuous PSO Kennedy and Eberhart R C developed Discrete Particle Swarm Optimization<sup>[9]</sup> in 1997, the PSO is extended to discrete space. In the DPSO every particle's position value is in binary form which is either 0 or 1. The velocity variable represents the change possibility of the position variable, the position of next moment is decided by the sigmoid function value of velocity, since the sigmoid function value is in the interval  $[0,1]$ , when the value exceeds a previously defined threshold, the position variable becomes 1, otherwise it is 0. In DPSO algorithm the position variable updating equation is:

$$X_i(k+1) = \begin{cases} 0, (1 + \exp(-v_i(k+1)))^{-1} < rand \\ 1, otherwise \end{cases} \quad (6)$$

DPSO is an effective algorithm for optimization problem, we try to utilize it in test frequencies selection, in the next section, we focus on the detailed procedure.

### III. TEST FREQUENCY SELECT PROCEDURE

The proposed method of test frequency selection in this paper belongs to SBT category. The frequency selection process mainly includes two steps: firstly we need define fault states in the circuit and simulate the fault states by circuit simulation software at various test frequencies. After simulation we can construct a fault dictionary for all the frequencies; Secondly we set the relevant parameters of the particle swarm algorithm and use it to search a minimal test frequency set based on the derived fault dictionary.

In the stage of fault dictionary construction, we need to define the faulty states of the circuit before simulation, the faulty states may include hard faults and parametric faults, hard faults are due to a catastrophic change in an element value like open circuit or short circuit cases. Soft faults are due to a change in an element value from its nominal value without reaching its extreme bounds. The circuit states set  $S$  is defined as  $S = \{S_0, S_1, \dots, S_L\}$ , in which  $L$  is the number of faulty states,  $S_0$  denotes the fault free state, and the others denote faulty states of the components.

After defining the faulty circuit states, all the states are simulated by the electronic circuit simulation software such as Pspice. The simulation needs to be performed on whole circuit working frequency band, if the frequency of circuit input signal is  $f_i$ , every fault state in  $S$  is simulated at  $f_i$ . For measurement convenience we take RMS(Root Mean Square) voltages at the output nodes as measured variables, and so get a measured vector  $M_i = \{m_{i0}, m_{i1}, \dots, m_{iN}\}$  ( $N$  is the total number of test frequencies). After the simulation of the circuit faulty states at various frequencies just like frequency  $f_i$ , we derive a matrix  $M_{L \times N}$ , whose component  $M_{ij}$  denotes the response voltage of faulty state  $S_i$  at frequency  $f_j$ . During the simulation, there is an important factor needs to be considered, that is the analog components' parameters of the circuit have tolerances. So in the simulation process, every component value tolerance is defined. We should run Monte Carlo simulation several times and get the maximum and minimum value of a faulty state. Thus matrix element  $M_{ij}$  would form a fuzzy interval, the voltage in this interval is the possible value of corresponding circuit states response. If at frequency  $f_j$  response  $M_{ij}$  and  $M_{kj}$  overlap or intersect, that means the test frequency  $f_i$  is unable to isolate the faulty states  $S_i$  and  $S_k$ . In general case the simulation band is from minimal working frequency  $f_{min}$  to maximum frequency  $f_{max}$ . If the frequency interval is  $\Delta f$ , the total simulation number  $f_{num}$  is:

$$f_{num} = \frac{f_{max} - f_{min}}{\Delta f} + 1 \quad (7)$$

In order to reduce the simulation work, we use the method proposed in [10], which compute the amplitude sensitivity of every fault, and choose the frequency band with highest sensitivity to simulate. We can get the sensitivity curve of every fault via the simulation software Pspice during the simulation stage. In the selected band the fault sensitivity is highest, so the change caused by the fault is most obvious, and this is helpful to detect the faults. Through doing this we reduce part of simulation work before test.

After simulation a fault dictionary for the circuit under test is constructed, It is matrix  $M_{L \times N}$  with  $L$  rows and  $N$  columns which reflect  $L$  faulty states at  $N$  frequencies. DPSO is adopted to select the optimal test frequency set. Define a binary  $N$  dimensional vector  $X_i = [a_{i1}, a_{i2}, \dots, a_{iN}]$ ,  $a_{ij}$  denotes the  $j$ th test frequency  $f_j$  is selected or not, when  $f_j$  is selected  $a_{ij} = 1$ , otherwise  $a_{ij} = 0$ , so the  $X_i$  substantially denotes a set of frequencies. If  $X_i$  is looked as the position variable of  $i$ th particle, when the particle reach the best position, the solution of optimal position  $X_i$  corresponds to the optimal selected frequency set.

In the particle swarm algorithm a key issue is designing a proper fitness function. The fitness function value of a particle decides its search direction and whether it is selected, so the fitness function construction is important. The fitness function needs to reflect fault detection rate, fault isolation rate and test cost factors. The smaller a fault testability is, the lower possibility of the fault being detected is. If faulty state  $S_i$  can only be detected at frequency  $f_i$  and  $f_j$ , then its

testability is 2. If we remove  $f_i$  and  $f_j$  from necessary frequency, faulty state  $S_i$  can't be detected any more, so selecting the frequencies with small fault coverage is helpful to detect and isolate faulty states, and assures the fault detection rate and isolation rate of test frequency set. For a single frequency in the frequency set, its fitness can be evaluated by formula (8):

$$h(t_i) = a \frac{N(f_i)}{S(f_i)} \quad (8)$$

where  $N(f_i)$  is total faults' quantity that frequency  $f_i$  can detect;  $S(f_i)$  is the minimum testability of faults that frequency  $f_i$  can detect, the testability of a fault is equal to the frequency quantity which can detect the fault; Coefficient  $a$  is a regulating constant, if we want the optimal test frequency set has high fault detection rate, the value of  $a$  is equal to a constant in interval  $[0,1]$ .

When it comes to evaluate the fitness value of a frequency set, the fitness function involves not only the fault detection rate and isolation rate but also other factors such as test cost or the number of frequencies, so the fitness function is constructed as formula (4):

$$F(T_s) = \eta_{FD} \eta_{FI} \left( \frac{\sum_{t_i \in T_s} h(t_i) / \sum_{t_i \in T_s} c_i}{N(\sum_{t_j \in T} h(t_j) / \sum_{t_j \in T} c_i)} \right) \quad (9)$$

Where  $\eta_{FD}$  is fault detection rate,  $\eta_{FI}$  is fault isolation rate,  $T$  is the original frequency set,  $T_s$  is the selected frequency set,  $N$  is the number of test in  $T_s$ ,  $c_i$  is the test cost of frequency  $f_i$ . The fitness function in formula (9) integrates fault detection rate, fault isolation rate and test cost in one function, and this turns multi-objective optimization problem into single objective optimization.

The detailed procedure to select optimal test frequency set for the circuit under test is performed following next steps:

step 1 Initialize a population consisting of  $p_{num}$  particles, set the maximum iteration number  $i_{max}$ . The beginning position variable  $X_i(0)$  of particle  $i$  is a random binary vector and the particle's velocity variable  $v_i(0)$  takes a random value,

step 2 Compute the fitness function value of all particles in the population according to formula (4),  $P_i$  and  $P_g$  are initialized with the particle position owning highest fitness value.

step 3 In the  $k$ th iteration, the position variable  $X_i(k)$  and velocity variable  $v_i(k)$  of the  $i$ th particle are updated according to formula (3).

step 4 Compute the fitness value of  $i$ th particle  $f(i)$  according to formula (9) and compare with  $f(P_i)$ , records the larger one as  $P_i$  among all particles.

step 5 if  $i \leq p_{num}$ , let  $i=i+1$ , go to step 3, else go to step 6.

step 6 if  $k \leq i_{max}$ , record the position with maximum fitness value as  $P_g$ , let  $k=k+1$ , go to step 3, else end.

In the DPSO algorithm, when several particles has the same fitness value with the best position value  $P_g$ , then these particles are also retained in a set. So there perhaps exist a group of solutions for optimal test frequency selection among the same frequency set.

#### IV. APPLICATION EXAMPLE

In this section we illustrate the proposed algorithm of test frequency selection by an example. Figure 1 is the schematic of a lowpass filter.

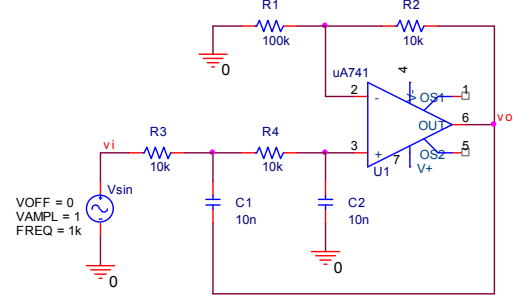


Figure 1 low pass filter

The filter is consist of 6 analog components and a operator amplifier, every component's parameter value is known, the tolerance of every component is  $\pm 5\%$ . We define the faults that the parameter deviate  $\pm 50\%$  from the component's nominal value, and 2 extra parametric faults are considered for the two capacitors,  $S_0$  is the fault free state, so there are 15 states considered in the example circuit. The defined states are simulated in a fault sensitive band which includes 13 various test frequencies. Every circuit state we run 20 Monte Carlo simulations and record the maximum and minimum RMS voltages at the output node, these voltages are divided into 6 fuzzy groups, the simulation results are shown in table 1.

The fault dictionary matrix is input into DPSO program, the particles population number is set to be 20, and maximum iteration number is 50, constant  $a$  is 1.2, suppose test cost at every test frequency is 1, then run the program and the optimization results are:

$$\begin{aligned} X_1 &= [0 \ 1 \ 0 \ 0 \ 0 \ 0 \ 1 \ 1 \ 1 \ 0 \ 0 \ 1 \ 0] \\ X_2 &= [0 \ 0 \ 1 \ 0 \ 0 \ 0 \ 1 \ 1 \ 1 \ 0 \ 0 \ 1 \ 0] \\ X_3 &= [0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 1 \ 1 \ 1 \ 0 \ 1 \ 0] \end{aligned} \quad (10)$$

These three solution have the same fitness value and the corresponding optimal test frequency is  $[f_2, f_7, f_0, f_9, f_{12}]$ ,  $[f_3, f_7, f_8, f_9, f_{12}]$  and  $[f_7, f_8, f_9, f_{10}, f_{12}]$ , the number of test frequencies is reduced from original 13 to 5.

In order to compare the algorithm performance we use both DPSO algorithm and genetic algorithm to select the circuit test frequencies. Both algorithm are Operated on the same computer, the DPSO algorithm spends 5.76 seconds to get the right solution, but the genetic algorithm takes about 12 seconds to derive the solution. The test frequency selection method based on DPSO is also tested by some large fault dictionary with more faulty states, frequencies and fuzzy intervals, experiment results show that the DPSO converge more quickly than genetic algorithm and the solution is globally optimal. In the algorithm performance analysis we also find that the DPSO algorithm using formula (3) converge more slowly than using formula (4), so the inertia weight algorithm is better than the basic DPSO.

## V. SUMMARIES

Test frequency selection is an important step for analog circuit automatic testing. An efficient test frequency selection method based on discrete particle swarm algorithm is proposed. The novelty of the method is the integration of discrete particle swarm algorithm, through designing fitness function the test frequency selection is treated as an optimization problem in discrete space. The proposed method is tested on an example circuit, and simulation results show that the DPSO algorithm has better performance compared with other optimization algorithm, it converges quickly and the final results are globally optimal. So the proposed frequency selection method are applicable in analog circuit automatic testing and fault diagnosis.

## REFERENCES

- [1] J.W.Bandler; A.E.Salama; Fault Diagnosis of Analog Circuits. Proceedings. of the IEEE[C], 1985.8, Vol.73,No.8, PP:1279-1325.
- [2] R. Sacks; N. Sen; Fault diagnosis for linear system via multifrequency measurement. IEEE Transaction on Circuits and System[J], Jul. 1979.7, vol. CAS-26, no. 7, PP:457– 465.
- [3] S. Mir; M. Lubaszewski;V. Kolarik; B.Courtois; Automatic test generation for maximal diagnosis of linear analog circuits. Proceeding. EDTC[C], 1996, PP:254–258.
- [4] Francesco Grasso; Antonio Luchetta; Stefano Manetti; Maria Cristina Piccirilli; A Method for the Automatic Selection of Test Frequencies in Analog Fault Diagnosis. IEEE Transactions on Instrumentation and Measurement[J], 2007.12, Vol.56, No.6, PP:2322-2329.
- [5] N. Nagi; A.Chatterjee; A.Balivada; J.A. Abraham; Fault-based automatic test generator for linear analog circuits. International Conference on Computer-Aided Design[C], 1993.11, PP:88-91.
- [6] P.Jantos; D.Grzechca; T.Golonek; J.Rutkowski; Heuristic methods to test frequencies optimization for analogue circuit diagnosis. Bulletin of the Polish Academy of Sciences[J], 2008., Vol 56, No.1, PP:29-38.
- [7] Kennedy J; Eberhart R C; Particle swarm optimization. Proceedings of IEEE International Conference on Neural Networks[C], 1995, PP:1942-1948.
- [8] Shi Y; Eberhart R C; A modified particle swarm optimizer. Proceedings of IEEE World Congress on Computational Intelligence[C], 1998, Vol.1:69-73.
- [9] Kennedy J; Eberhart R C; A discrete binary version of the particle swarm algorithm. Proceedings of IEEE Conference on Systems, Man, and Cybernetics[C], 1997, Vol.5:4104-4108.
- [10] Constantin Viorel Marin; Doina Marin; Test frequency selection in analog fault diagnosis. part II: results. Revue Roumaine des Sciences Techniques-serie Electrotechnique et Energetique[J], 2008. 53 (1), PP:41-50.

TABLE I. CIRCUIT RESPONSE OF ALL STATES AT DIFFERENT FREQUENCIES

Fault state	f1	f2	f3	f4	f5	f6	f7	f8	f9	f10	f11	f12	f13
S0	1	1	1	4	2	1	5	2	4	3	1	3	2
S1	2	1	2	3	2	2	4	3	1	3	1	3	2
S2	3	1	4	2	1	3	3	4	4	3	1	3	2
S3	1	1	1	4	2	2	3	2	4	3	1	3	2
S4	3	1	4	2	1	3	3	4	4	1	2	2	2
S5	1	1	1	4	2	1	5	2	4	2	1	2	2
S6	1	4	3	4	2	1	5	2	4	5	1	3	2
S7	1	1	1	4	2	1	5	2	5	4	1	3	2
S8	1	1	1	4	2	1	5	4	4	3	1	3	2
S9	1	2	5	4	2	1	5	2	4	4	1	3	2
S10	1	1	1	4	2	1	5	3	1	3	1	3	3
S11	1	1	1	4	2	1	1	2	4	3	1	3	2
S12	1	1	1	4	2	1	5	2	2	4	1	3	1
S13	1	1	1	4	2	1	5	2	4	3	1	1	2
S14	4	3	3	1	2	4	2	1	4	4	1	3	2