J Electron Test (2010) 26:523–534
DOI 10.1007/s10836-010-5169-4

A Novel Test Point Selection Method for Analog Fault Dictionary Techniques

ChengLin Yang · ShuLin Tian · Bing Long · Fang Chen

Received: 9 July 2009 / Accepted: 29 July 2010 / Published online: 1 September 2010

Abstract Most of the recently reported test point selection algorithms for analog fault dictionary techniques are based on integer-coded table (ICT) technique. Hence, the accuracy of these algorithms is closely related to the accuracy of the ICT technique. Unfortunately, this technique is not accurate, especially when the size of fault dictionary is large. This paper proposes an accurate fault-pair Boolean table technique for the test point selection problem. First, the approach to transform the fault dictionary into a fault-pair Boolean table is introduced. Then, a test point selection algorithm based on the fault-pair Boolean table is proposed. Thirdly, several example circuits are used to illustrate the proposed algorithm. Simulated results indicate that the proposed method is more accurate than the other methods. Therefore, it is a good solution for minimizing the size of the test point set.

Keywords Analog fault dictionary · Integer-coded table · Fault-pair Boolean table · Test points selection · Fault diagnosis

1 Introduction

Methods for the fault diagnosis of analog circuits are usually classified into two main categories [3]: simulation before test (SBT) and simulation after test (SAT). The probabilistic method and fault dictionary techniques fall into SBT, and optimization, fault verification and parameter identification techniques fall into SAT. Among the various fault diagnosis methods, one of the most widely used SBT methods is the fault dictionary method [6]. This method is applicable to both hard and soft fault detection and consists of three important phases [12]. First, in the construction of a fault dictionary, all potential faults are listed and the stimuli type (i.e., dc, ac, or time domain), shape, frequency, and amplitude are selected. A circuit under test (CUT) is then simulated for the fault-free case and all fault cases. The fault signatures (i.e., output response or a set of responses or other parameters that are derived from output responses) are generated when a test program is executed on a unit having a fault. The fault features of the responses are stored and organized in the dictionary for fault diagnosis. The second phase is test point selection. In this phase, an optimum test point set is required to achieve the desired degree of fault diagnosis. The third phase is fault isolation. At this stage, the same stimuli as those used in constructing the dictionary are applied to the fault CUT. The measured values at selected test points or calculated signatures are compared with those stored in the dictionary to match the fault(s) to one of the predefined faults or to a set of faults according to preset criteria [12]. This paper focuses on the second phase.

The problem of test point selection for the analog fault dictionary technique has been studied extensively [1, 2, 4–6, 8–13, 15–17]. Varghese et al. [15] proposed a heuristic method for test point selection based on the concept of confidence levels determined by using distance concepts. The concept of ambiguity groups and developed logical rules to select test points was proposed by Hochwald and Bastian [5]. Stenbakken and Souders proposed a method of QR factorization of a circuit sensitivity matrix [13].

Responsible Editor: J. W. Sheppard

C. Yang (✉) · S. Tian · B. Long
University of Electronic science and technology of China,
Chengdu 610054, China
e-mail: yangclin@yahoo.com.cn

F. Chen
Chengdu College of Sichuan Normal University,
Chengdu 610054, China
e-mail: pingfan9896@sina.com

Abderrahman et al. [13] used sequential quadratic programming and constraint logic programming to generate the test set. Lin and Elcherif [6] proposed two heuristic methods based on the two criteria proposed by Hochward and Bastian. Spaandonk and Kevenaar [11] proposed looking for a set of test points by combining the decomposition method of a system sensitivity matrix and an iterative algorithm. Prasad and Babu [12] proposed four algorithms for inclusive approaches and three strategies for exclusive approaches. Based on the fault detection table and computing the information content of every node, Pinjala and Kim [8] proposed an approach to get the near-minimum test sets. An entropy-based approach was proposed by Starzyk, et al. [12]. Golonek and Rutkowsk [4] proposed a genetic-algorithm-based test point selection method. Most test point selection algorithms and strategies [4, 6, 8–10, 12, 16, 17] mentioned above are based on the ICT technique. Hence, the accuracy of these algorithms is closely related to the accuracy of the ICT technique.

Since a single-fault situation is mostly encountered in fault diagnosis, only if the method works for this situation can it be generalized to solving multiple soft fault cases. Hence, this paper mainly focuses on tolerance handling in the presence of single faults. Section 2 illustrates the ICT technique and its shortcoming, and a new fault-pair Boolean table technique is proposed to overcome this shortcoming. A test point selection algorithm based on the fault-pair Boolean table is given in this section as well. Section 3 presents several simulated results and demonstrates the superiority in the computational efficiency and solution quality of the proposed method by comparing it with the other reported methods. Section 4 concludes this work.

The nomenclature of this paper is as follows:

n_j	The *jth* test point.
$I(n_j)$	Count of the number of 1′ s associated with test point n_j.
N_T	Number of candidate test points.
f_i	The *ith* fault.
N_F	Number of all potential faults (including the nominal case).
S_{opt}	Desired test point set.
S_C	Candidate test point set.
NT_i	Number of test points that can isolate the *ith* fault pair.

Table 1 Fault dictionary

Faults	n_1	n_2	n_3
f_0	6.1v	9.0v	2.3v
f_1	6.6v	8.1v	2.4v
f_2	7.2v	9.3v	2.8v
f_3	7.8v	8.5v	3.4v

Table 2 Integer-coded table

Faults	n_1	n_2	n_3
f_0	0	0	0
f_1	0	0	0
f_2	0	0	0
f_3	0	0	0

2 Fault-Pair Boolean Table

Most of the reported test point selection algorithms and strategies [4, 6, 8–10, 12, 16, 17] are based on the ICT technique. This technique was first proposed by Lin and Elcherif [6]. According to this technique [6, 9, 12], an ambiguity group is defined as any two faulty conditions that fall into the same ambiguity group if the gap between the voltage values produced by them is less than 0.7 volts. If the two fault components belong to one ambiguous group, they are undistinguishable. Faults that belong to the same ambiguity group in a given column are coded as the same integer. In this table, the same integer number represents all the faults that belong to the same ambiguity group in a given column. Since each test point represents an independent measurement, ambiguity groups of different test points are independent and can be numbered using the same integers without confusion. The detailed description of this technique can be found in the literature [6, 9, 12].

2.1 The Shortage of ICT Technique

Assume that Table 1 shows faulty (including fault free) voltage values of a network. Faults f_0 and f_2 are isolated by test point n_1 because the voltage gap produced by them is larger than 0.7 volts. Hence, the fault pair (f_0, f_2) is isolated by n_1. Similarly, the fault pairs (f_0, f_3) and (f_1, f_3) are isolated by n_1. The residual fault pairs (f_0, f_1), (f_1, f_2) and (f_2, f_3) are isolated by test point n_2. Thus, test point set $\{n_1, n_2\}$ can isolate all faults.

If the ICT technique is adopted, we will conclude that not all of the faults can be isolated. For example, on test point n_1 of

Table 3 Fault-pair Boolean table

Sequence number of fault pair	Fault pair	n_1	n_2	n_3	NT_i
1	(f_0, f_1)	0	1	0	1
2	(f_0, f_2)	1	0	0	1
3	(f_0, f_3)	1	0	1	2
4	(f_1, f_2)	0	1	0	1
5	(f_1, f_3)	1	0	1	2
6	(f_2, f_3)	0	1	0	1
$I(n_j)$		3	3	2	–

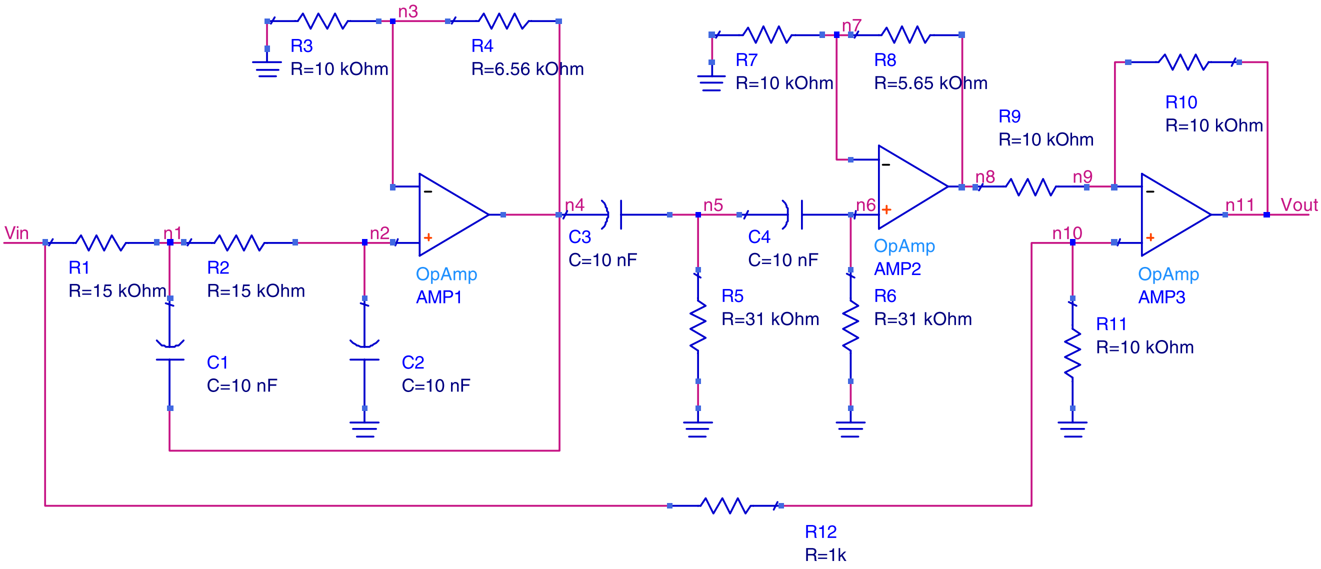

Fig. 1 A band pass filter circuit

Table 1, faults f_0 and f_1 are coded as 0. The voltage gap produced by faults f_1 and f_2 is 0.6 volts (smaller than the 0.7-volt criterion). Hence, they fall into one ambiguous group. Since f_1 is coded as 0, f_2 should also be coded as 0 according to the ICT technique [6, 12]. Similarly, f_3 is coded as 0. Table 2 shows the results. In this table, all the faults are coded as 000. Therefore, all the faults are undistinguishable. Based on this table, regardless of which test point selection algorithm is used, we cannot get any solution.

Apparently, the traditional ICT technique induces a wrong conclusion in this example. The fault-pair Boolean

Table 4 AC voltage fault dictionary (Unit: V)

	n_1	n_2	n_3	n_4	n_5	n_6	n_7	n_8	n_9	n_{10}	n_{11}
f_0(NOM)	4.33	3.16	3.16	5.24	3.39	2.99	2.99	4.67	3.64	3.64	3.20
f_1(R1Short)	4.00	2.92	2.92	4.84	3.12	2.76	2.76	4.31	3.64	3.64	3.70
f_2(R1Open)	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	3.64	3.64	7.27
f_3(R2Short)	3.81	3.81	3.81	6.31	4.07	3.59	3.59	5.62	3.64	3.64	5.47
f_4(R2Open)	2.92	0.00	0.00	0.00	0.00	0.00	0.00	0.00	3.64	3.64	7.27
f_5(R3Open)	2.92	2.13	2.13	2.13	1.38	1.22	1.22	1.90	3.64	3.64	5.53
f_6(R4Open)	0.00	0.00	0.00	4.27	2.76	2.44	2.44	3.81	3.64	3.64	10.86
f_7(R5Short)	4.33	3.16	3.16	5.24	0.00	0.00	0.00	0.00	3.64	3.64	7.27
f_8(R5Open)	4.33	3.16	3.16	5.24	4.06	3.58	3.58	5.60	3.64	3.64	4.43
f_9(R6Short)	4.33	3.16	3.16	5.24	2.53	0.00	0.00	0.00	3.64	3.64	7.27
f_{10}(R6Open)	4.33	3.16	3.16	5.24	4.62	4.62	4.62	7.23	3.64	3.64	6.84
f_{11}(R7Open)	4.33	3.16	3.16	5.24	3.39	2.99	2.99	2.99	3.64	3.64	4.54
f_{12}(R8Open)	4.33	3.16	3.16	5.24	3.39	2.99	0.27	15.00	6.27	3.64	15.00
f_{13}(R9Open)	4.33	3.16	3.16	5.24	3.39	2.99	2.99	4.67	3.64	3.64	3.64
f_{14}(R10Open)	4.33	3.16	3.16	5.24	3.39	2.99	2.99	4.67	4.55	3.64	15.00
f_{15}(R11Open)	4.33	3.16	3.16	5.24	3.39	2.99	2.99	4.67	4.00	4.00	3.86
f_{16}(R12Open)	4.33	3.16	3.16	5.24	3.39	2.99	2.99	4.67	0.00	0.00	4.67
f_{17}(C1Open)	2.58	1.88	1.88	3.12	2.02	1.78	1.78	2.78	3.64	3.64	4.50
f_{18}(C2Open)	3.41	3.41	3.41	5.64	3.65	3.22	3.22	5.04	3.64	3.64	9.37
f_{19}(C2Short)	1.81	0.00	0.00	0.00	0.00	0.00	0.00	0.00	3.64	3.64	7.27
f_{20}(C3Short)	4.33	3.16	3.16	5.24	5.24	4.62	4.62	7.23	3.64	3.64	6.84
f_{21}(C4Open)	4.33	3.16	3.16	5.24	4.62	0.00	0.00	0.00	3.64	3.64	7.27
f_{22}(C4Short)	4.33	3.16	3.16	5.24	3.58	3.58	3.58	5.60	3.64	3.64	4.43

Table 5 Fault-pair Boolean table

	Fault pair	n_1	n_2	n_3	n_4	n_5	n_6	n_7	n_8	n_9	n_{10}	n_{11}	NT_i
1	(f_0,f_1)	1	1	1	1	1	1	1	1	0	0	1	9
2	(f_0,f_2)	1	1	1	1	1	1	1	1	0	0	1	9
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
13	(f_0,f_{13})	0	0	0	0	0	0	0	0	0	0	1	1
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
45	(f_2,f_4)	1	0	0	0	0	0	0	0	0	0	0	1
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
135	(f_7,f_9)	0	0	0	0	1	0	0	0	0	0	0	1
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

table technique proposed in the following section can overcome this shortcoming.

2.2 Fault-Pair Boolean Table Technique

In the proposed fault-pair Boolean table, rows represent any possible fault pairs, and columns show the available test points. A fault dictionary that consists of N_F faults has $C^2_{N_F} = \frac{N_F\times(N_F-1)}{2}$ fault pairs. Each test point n_j is represented by a binary column vector d of dimension $\frac{N_f\times\left(N_f-1\right)}{2}$. Test point n_j isolates the ith $\left(1 \leq i \leq \frac{N_f\times\left(N_f-1\right)}{2}\right)$ fault pair if the ith component of the test vector d_{ij} is 1.

According to the 0.7-volt criterion, f_0 and f_1 cannot be isolated by n_1 in Table 1; hence, the corresponding component d_{11} of Table 3 is 0. Similarly, two other fault pairs (f_1, f_2) and (f_2, f_3) cannot be isolated by n_1, and d_{14} and d_{16} are 0. These three fault pairs can be isolated by test point n_2; therefore, d_{21}=d_{24}=d_{26}=1. The other components of Table 5 are derived in the same way. Since all six fault pairs can be isolated by n_1 and n_2 together, test point set $\{n_1, n_2\}$ is the optimum solution.

It is important to note that in the present era of low-voltage analog circuits, the 0.7-volt criterion seems to be excessive. Supposedly, 0.2 volts gap should be enough to distinguish two faults; therefore, this gap is adopted in some literature [14, 16]. Hence, the 0.2-volt criterion is adopted in the following text. The problem of whether this criterion is the best criterion or not falls into the first phase of fault dictionary techniques, and its discussion is beyond the scope of this paper.

2.3 Test Point Set Selection Algorithm

The global minimum set of test points can only be guaranteed by an exhaustive search method [12], which is computationally expensive and impractical. This paper handle the near optimum test point set selection problem. The popular choice is the greedy search algorithm [4, 8–10, 12, 17]. The main idea of the greedy search algorithm for the test point selection problem can be illustrated as follows. The optimum test point set (S_{opt}) is initialized to null. Then, the best test point among the candidate test point set (S_C) is drawn from S_C, and added into S_{opt}. If the test points in S_{opt} are not sufficient to diagnose the fault pairs, the best test point among S_C is drawn from S_C and added into S_{opt}. The above steps are repeatedly executed until all fault pairs are isolated. Whether a test point is the best one or not is evaluated by a test point evaluation criterion [16]. The test point evaluation criterion used in this paper is a measure known as information content $I(n_j)$. Test point n_1 in Table 3 can distinguish three fault pairs (f_0, f_2), (f_0, f_3) and (f_1, f_3). Hence, $I(n_1)$=3. Similarly, $I(n_2)$= 3 and $I(n_3)$=2. The $I(n_j)$ value is easy to calculate, and $\max\limits_{j}(I(n_j))$ is chosen as the test point evaluation criterion in this paper.

Only one test point n_2 can isolate the 4th fault pair (f_1, f_2) in Table 3; hence, NT_4=1. Therefore, this test point is necessary for fault isolation, and it should be added into the desired optimum test point set (S_{opt}). According to the above discussion, the proposed test point selection algorithm based on the fault-pair Boolean table technique can be illustrated as follows.

Table 6 Residual fault pairs

	Fault pair	n_2	n_3	n_4	n_6	n_7	n_8	n_9	n_{10}	NT_i
192	(f_{11},f_{16})	0	0	0	0	0	1	1	1	3
198	(f_{11},f_{22})	0	0	0	1	1	1	0	0	3
200	(f_{12},f_{14})	0	0	0	0	1	1	1	0	3
$I(n_j)$		0	0	0	1	2	3	2	1	–

Step 1) The optimum test point set S_{opt} is initialized to a null set.

Step 2) For each fault pair that corresponds to NT_i=1, the test point that can uniquely isolate the ith fault pair is added into S_{opt}. The rows (fault pairs) that can be isolated by this test point are deleted from the fault-pair Boolean table, and the test point is removed from the fault-pair Boolean table. Go to Step 4.

Step 3) $I(n_j)$ of every test point in the fault-pair Boolean table is calculated, and the test point with $\max\limits_{j}(I(n_j))$ is added into S_{opt}. The rows (fault pairs) isolated by this test point are deleted. The point is also deleted.

Step 4) If all the fault pairs are isolated, exit. Otherwise, go to Step 3.

It is important to note that NT_i used in Step 2 and the $I(n_j)$ value for the first iteration were obtained after the fault-pair table was constructed.

Usually, in an actual analog circuit, some faults can be diagnosed only by a certain test point. In this case, Step 2 helps the algorithm more accurately and efficiently finds the final solution. If there is only one optimum solution, the proposed algorithm can find the global optimum solution by using just one step, viz. Step 2. Even there are more than one optimum, Step 2 dramatically decreases the size of the fault-pair Boolean table. For example, in Table 2 from [12], f_1 and f_3 can be distinguished only by test point n_1, f_8 and f_{10} are diagnosed by only one test point n_5, f_{15} and f_{16} are isolated by exclusive test point n_9, and f_0 and f_{14} are discerned by n_{11}. Therefore, n_1, n_5, n_9, and n_{11} should be added into S_{opt} according to Step 2. These four test points isolate all faults. In this example, only one step (Step 2) can find the optimum solution.

Theorem 1 *The time complexity of the proposed algorithm is less than* $O\left(N_F^2 N_T m\right)$.

Proof As discussed in Section 2.2, the fault-pair table consists of $\frac{N_F\times(N_F-1)}{2}$ rows. Hence, in Step 2, deleting the rows (fault pairs) isolated by the selected test points has the time complexity of $O\left(\frac{N_F\times(N_F-1)}{2}\right)=O\left(N_F^2\right)$. In Step 3, the time complexity of calculating $I(n_j)$ is $O\left(N_F^2\times N_T\right)$,

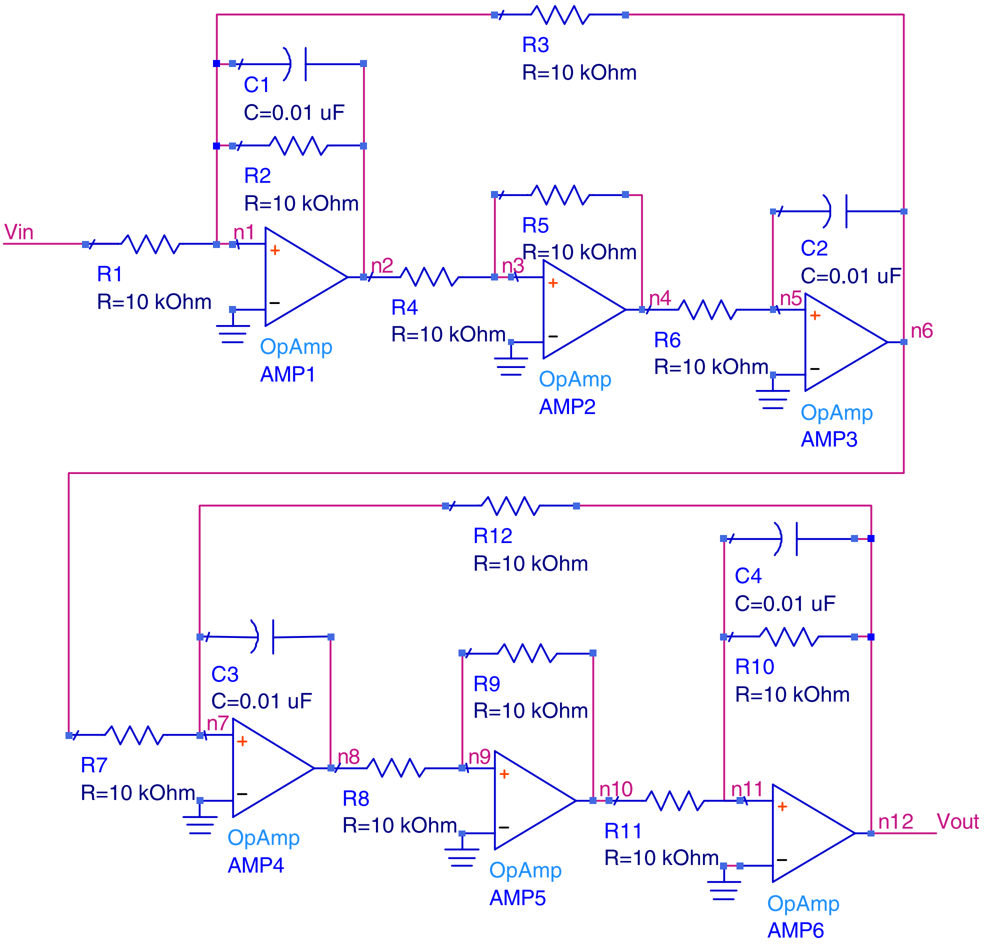

Fig. 2 A Leapfrog filter circuit

Table 7 Fault dictionary (Unit:V)

	n_1		n_2		n_3		n_4		n_5		n_6		n_7		n_8		n_9		n_{10}		n_{11}		n_{12}	
	H	L	H	L	H	L	H	L	H	L	H	L	H	L	H	L	H	L	H	L	H	L	H	L
Normal	0	0	4.92	3.48	0	0	5.41	3.21	0	0	4.62	2.37	0	0	4.95	1.88	0	0	5.22	1.81	0	0	4.79	1.40
R1Open	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R2Open	0	0	15	15	0	0	15	15	0	0	15	15	0	0	15	15	0	0	15	15	0	0	15	15
R3Open	0	0	4.17	3.09	0	0	4.61	2.80	0	0	4.07	2.02	0	0	4.36	1.60	0	0	4.60	1.54	0	0	4.22	1.19
R4Open	0	0	4.17	3.09	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R5Open	0	0	0	0	0	0	6.82	4.14	0	0	4.92	3.65	0	0	5.28	2.89	0	0	5.57	2.78	0	0	5.11	2.15
R6Open	0	0	4.17	3.09	0	0	4.61	2.80	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R7Open	0	0	4.92	3.48	0	0	5.41	3.21	0	0	4.62	2.37	0	0	0	0	0	0	0	0	0	0	0	0
R8Open	0	0	4.92	3.48	0	0	5.41	3.21	0	0	4.62	2.37	0	0	4.07	1.71	0	0	2.32	0.84	0	0	0	0
R9Open	0	0	4.92	3.48	0	0	5.41	3.21	0	0	4.62	2.37	0	0	0	0	0	0	5.64	1.94	0	0	5.11	2.15
R10Open	0	0	4.92	3.48	0	0	5.41	3.21	0	0	4.62	2.37	0	0	15	15	0	0	15	15	0	0	15	15
R11Open	0	0	4.92	3.48	0	0	5.41	3.21	0	0	4.62	2.37	0	0	4.07	1.71	0	0	4.40	1.60	0	0	0	0
R12Open	0	0	4.92	3.48	0	0	5.41	3.21	0	0	4.62	2.37	0	0	4.07	1.71	0	0	4.40	1.60	0	0	4.22	1.19
C1Open	0	0	4.12	2.61	0	0	4.25	2.58	0	0	3.44	1.99	0	0	3.68	1.58	0	0	3.88	1.52	0	0	3.56	1.17
C2Open	0	0	0	0	0	0	0	0	0	0	4.92	3.65	0	0	5.28	2.89	0	0	5.56	2.78	0	0	5.10	2.15
C3Open	0	0	4.92	3.48	0	0	5.41	3.21	0	0	4.62	2.37	0	0	7.36	2.07	0	0	6.86	2.23	0	0	5.10	2.15
C4Open	0	0	4.92	3.48	0	0	5.41	3.21	0	0	4.62	2.37	0	0	3.41	1.23	0	0	3.52	1.21	0	0	3.56	1.17
C1short	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
C2short	0	0	4.17	3.09	0	0	4.61	2.80	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
C3short	0	0	4.92	3.48	0	0	5.41	3.21	0	0	4.62	2.37	0	0	0	0	0	0	0	0	0	0	0	0
C4short	0	0	4.92	3.48	0	0	5.41	3.21	0	0	4.62	2.37	0	0	4.07	1.71	0	0	4.07	1.71	0	0	0	0

H- High faulty voltage limits

L- Low faulty voltage limits

Table 8 Faults number

f_0	Normal, R3Open, R12Open, C1Open, C3Open, C4Open
f_1	R1Open, C1Short
f_2	R2Open
f_3	R4Open
f_4	R5Open
f_5	R6Open, C2Short
f_6	R7Open, R3Short
f_7	R8Open, R11Open, C4Short
f_8	R9Open
f_9	R10Open
f_{10}	C2Open

and deleting rows has the time complexity of $O\left(N_F^2\right)$. Suppose that Step 3 is executed for m iterations, viz. $|S_{opt}| = m$, the total complexity of the proposed algorithm is $O\left(N_F^2 + N_F^2 \times N_T \times m\right) = O\left(N_F^2 N_T m\right)$. Because the isolated faults are deleted in each iteration, the size of the fault-pair table decreases gradually. Therefore, the total complexity is less than $O\left(N_F^2 + N_F^2 \times N_T \times m\right) = O\left(N_F^2 N_T m\right)$.

3 Simulation Examples

3.1 Simulations on Circuits

Example 1 A band pass filter circuit example

The filter circuit shown in Fig. 1 is stimulated by a 1 kHz, 4 V sinusoidal wave. A total of 23 faulty conditions f_0–f_{22} and eleven test points n_1–n_{11} are examined. Voltage values at all the test points for different faulty conditions are simulated by using PSPICE. The results are shown in Table 4. The fault-pair Boolean table, Table 5, is constructed by using the procedures introduced in Section 2.2

In Step 1 of the proposed algorithm, the optimum test point set S_{opt} is initialized to a null set. In Step 2, NT_{13}=1, and n_{11} is the only one test point that isolates the 13*th* fault pair (f_0, f_{13}). Hence, n_{11} is added into S_{opt}. The rows isolated by n_{11} are deleted from Table 5. For example, the 1*st* fault pair (f_0, f_1) and the 2*nd* fault pair (f_0, f_2), isolated by n_{11}, are removed. Test point n_{11} is also removed from Table 5. Similarly, n_1 and n_5 should be added into S_{opt}, and then $S_{opt} = \{n_1, n_5, n_{11}\}$. After all the fault pairs isolated by the test point set S_{opt} are removed from Table 5, three rows shown in Table 6 remain.

In Step 3, the $I(n_j)$ value is calculated, and the results are shown in the final row of this table. Test point n_8, which has the maximum $I(n_j)$ value among the candidate test points, is added into S_{opt}, and then $S_{opt} = \{n_1, n_5, n_8, n_{11}\}$. So far, all fault pairs are isolated. The optimum solution is found when the proposed algorithm is executed for only one iteration. If the integer-coded technique is adopted, the size of the test point set found by other algorithms, including the exhaustive method, is larger than 4. The results are shown below.

Graph search algorithm [17]:

$$S_{opt} = \{n_1, n_5, n_6, n_9, n_{11}\}$$

Pinjala's algorithm [8]:

$$S_{opt} = \{n_1, n_5, n_8, n_9, n_{11}\}$$

Starzyk's algorithm [12]:

$$S_{opt} = \{n_1, n_5, n_8, n_9, n_{11}\}$$

The exhaustive algorithm [17]:

Table 9 A part of fault-pair Boolean table

Fault pair	n_1	n_2	n_3	n_4	n_5	n_6	n_7	n_8	n_9	n_{10}	n_{11}	n_{12}	NT_i
(f_0,f_1)	0	1	0	1	0	1	0	1	0	1	0	1	6
(f_0,f_2)	0	1	0	1	0	1	0	1	0	1	0	1	6
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
(f_0,f_4)	0	1	0	0	0	0	0	0	0	0	0	0	1
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
(f_0,f_7)	0	0	0	0	0	0	0	0	0	0	0	1	1
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
(f_0,f_8)	0	0	0	0	0	0	0	1	0	0	0	0	1
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
(f_3,f_5)	0	0	0	1	0	0	0	0	0	0	0	0	1
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
(f_5,f_6)	0	0	0	0	0	1	0	0	0	0	0	0	1
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

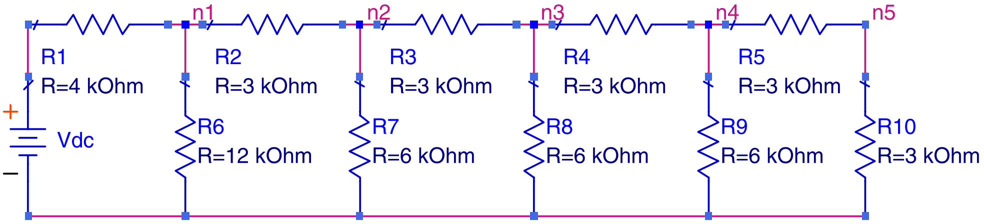

Fig. 3 A linear voltage-dividing circuit

$S_{opt} = \{n_1, n_5, n_6, n_9, n_{11}\}$

Obviously, the proposed method more accurately finds the solution than the other algorithms.

Example 2 Leapfrog filter circuit example

Figure 2 shows a Leapfrog filter circuit. A total of 20 potential faults are examined. All 12 test points are assumed to be accessible for the purpose of demonstration. The excitation signal is a f=1 kHz, V_p=6 V sinusoidal wave. The tolerance of resistors and capacitors is 5% and 10%, respectively. High and Low faulty voltage limits on every test point of each individual fault are calculated by using a worst-case analysis. The measurement accuracy is ±5%. Table 7 shows the results of worst-case analysis. The voltage signature of every fault varies between its high and low faulty voltage limit. For example, the signature area of fault condition *R3Open* on test point n_2 is (4.17, 3.09). If two faults have overlapping signature areas, these two faults are undistinguishable. On test point n_2, the nominal case and *R3Open* cannot be diagnosed because their signature areas have the common intersection (4.17, 3.48). The undistinguished faults are listed in Table 8. Based on Tables 7, 8, and 9 is obtained by using the proposed fault-pair technique.

It is easy to see from Table 9 that test points n_2, n_4, n_6, n_8, and n_{12} are necessary to isolate (f_0, f_4), (f_3, f_5), (f_5, f_6), (f_0, f_8) and (f_0, f_7), respectively. Hence, S_{opt} = {n_2, n_4, n_6, n_8, n_{12}}. So far the test points in S_{opt} isolate all the fault pairs. It means that when the proposed algorithm runs to Step 2 in the first iteration, the final solution is found. The computation time is only about 0.9 ms.

In this example, all the other algorithms, including the graph search algorithm [17], Pinjala's algorithm [8], and the entropy-based algorithm [12] also find the optimum solution. However, these algorithms run for 5 iterations, and the computation time is about 2.1 ms. This result justifies the former deduction that Step 2 helps the proposed algorithm more accurate and efficient than the other ICT-based algorithms.

Example 3 A linear net

The fault dictionary technique is traditionally used to diagnose catastrophic faults; however, it can also be used to diagnose the soft fault of linear circuits. Hence, the proposed fault-pair technique and test point selection algorithm are also applicable to the soft fault diagnosis problem.

Assume n_1 and n_2 are two different test points (except for the reference point) in a linear circuit N, the fault set is

Table 10 Slope value

Fault component	P_1 n_1 $\mid$ n_2	P_2 n_1 $\mid$ n_3	P_3 n_1 $\mid$ n_4	P_4 n_1 $\mid$ n_5	P_5 n_1 $\mid$ n_3	P_6 n_2 $\mid$ n_4	P_7 n_2 $\mid$ n_5	P_8 n_3 $\mid$ n_4	P_9 n_3 $\mid$ n_5	P_{10} n_4 $\mid$ n_5
f_1(R2)	−1.00	−2.00	−4.00	−8.00	2.00	4.00	8.00	2.00	4.00	2.00
f_2(R3)	0.50	−0.50	−1.00	−2.00	−1.00	−2.00	−4.00	2.00	4.00	2.00
f_3(R4)	0.50	0.25	−0.25	−0.50	0.50	−0.50	−1.00	−1.00	−2.00	2.00
f_4(R5)	0.50	0.25	0.13	−0.13	0.50	0.25	−0.25	0.50	−0.50	−1.00
f_5(R6)	2.00	4.00	8.00	16.00	2.00	4.00	8.00	2.00	4.00	2.00
f_6(R7)	0.50	1.00	2.00	4.00	2.00	4.00	8.00	2.00	4.00	2.00
f_7(R8)	0.50	0.25	0.50	1.00	0.50	1.00	2.00	2.00	4.00	2.00
f_8(R9)	0.50	0.25	0.13	0.25	0.50	0.25	0.50	0.50	1.00	2.00
f_9(R10)	0.50	0.25	0.13	0.06	0.50	0.25	0.13	0.50	0.25	0.50

Table 11 θ values

Fault component	P_1 n_1 $\vert$ n_2	P_2 n_1 $\vert$ n_3	P_3 n_1 $\vert$ n_4	P_4 n_1 $\vert$ n_5	P_5 n_1 $\vert$ n_3	P_6 n_2 $\vert$ n_4	P_7 n_2 $\vert$ n_5	P_8 n_3 $\vert$ n_4	P_9 n_3 $\vert$ n_5	P_{10} n_4 $\vert$ n_5
f_1(R2)	−45.0	−63.4	−76.0	−82.9	63.4	76.0	82.9	63.4	76.0	63.4
f_2(R3)	26.6	−26.6	−45.0	−63.4	−45.0	−63.4	−76.0	63.4	76.0	63.4
f_3(R4)	26.6	14.0	−14.0	−26.6	26.6	−26.6	−45.0	−45.0	−63.4	63.4
f_4(R5)	26.6	14.0	7.1	−7.1	26.6	14.0	−14.0	26.6	−26.6	−45.0
f_5(R6)	63.4	76.0	82.9	86.4	63.4	76.0	82.9	63.4	76.0	63.4
f_6(R7)	26.6	45.0	63.4	76.0	63.4	76.0	82.9	63.4	76.0	63.4
f_7(R8)	26.6	14.0	26.6	45.0	26.6	45.0	63.4	63.4	76.0	63.4
f_8(R9)	26.6	14.0	7.1	14.0	26.6	14.0	26.6	26.6	45.0	63.4
f_9(R10)	26.6	14.0	7.1	3.6	26.6	14.0	7.1	26.6	14.0	26.6

Table 12 Fault-pair Boolean table

	Fault component pair	P_1 n_1 $\vert$ n_2	P_2 n_1 $\vert$ n_3	P_3 n_1 $\vert$ n_4	P_4 n_1 $\vert$ n_5	P_5 n_1 $\vert$ n_3	P_6 n_2 $\vert$ n_4	P_7 n_2 $\vert$ n_5	P_8 n_3 $\vert$ n_4	P_9 n_3 $\vert$ n_5	P_{10} n_4 $\vert$ n_5	NT_i
1	(f_1,f_2)	1	1	1	1	1	1	1	0	0	0	7
2	(f_1,f_3)	1	1	1	1	1	1	1	1	1	0	9
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
36	(f_8,f_9)	0	0	0	1	0	0	1	0	1	1	4
–	$I(P_j)$	15	26	33	36	23	30	33	23	26	15	–

Table 13 Statistical simulation results

Size of the min. set	A_1	A_2	A_3	A_4	A_5	A_6
3	–	–	–	–	–	100
4	–	–	–	–	–	–
5	–	–	–	–	–	–
6	–	–	–	–	–	–
7	23	9	–	30	30	–
8	39	47	2	33	33	–
9	14	13	47	19	19	–
10	13	–	16	10	10	–
11	3	23	27	–	–	–
∞	8	8	8	8	8	–
Computation time per dictionary (Unit: ms)	33	40	35	124.9	320	214.9

A_1-Graph search algorithm [17]
A_2- Entropy method [12]
A_3- Pinjala's algorithm [8]
A_4- Genetic-algorithm [4]
A_5- Exhaustive algorithm [17]
A_6- Proposed algorithm

$F=\{f_0,f_1,\cdots f_n\}$,where f_0 is the nominal case, and f_i denotes a faulty condition of the i^{th} component. In the nominal case, the voltage values measured on n_1 and n_2 are V_{01} and V_{02}, respectively. For any other faulty condition f_i, the node voltage function must cross (V_{01},V_{02}). The node voltage function for f_i is expressed as follows [7]:

$$V_1-V_{10}=A_i(V_2-V_{20}), \tag{1}$$

where A_i is the slope of formula (1) on the V_1–V_2 plane. $\theta_i=\arctan A_i\left(-90^0\leq\theta_i\leq 90^0\right)$ is selected as the fault model.

Figure 3 shows a linear voltage-dividing circuit. The potential fault components are listed in Table 10. The A_i value for every component on each plane is obtained by simulation. In this table, P_j denotes the j^{th} plane. For example, P_1 is the V_{n1}–V_{n2} plane, P_2 denotes the V_{n1}–V_{n3} plane, P_{10} is the V_{n4}–V_{n5} plane, and so on. The components in Table 11 are the θ values of all potential fault components.

Assume that any two faulty components are distinguishable if the angle degree difference they produce is larger than 10°. The problem of whether 10° is the best criterion or not falls into the first phase of fault dictionary techniques. This problem is beyond the scope of this paper. The fault-pair Boolean table, Table 12, is constructed by using the procedures introduced in Section 2.2

The proposed algorithm is used to select the minimum plane set from Table 13. Because no fault pair has NT_i=1, Step 2 is overleaped. In Step 3, P_4, which has the maximum $I(P_4)$ value 36, is added into S_{opt}. Because all 36 fault pairs are isolated by $S_{opt}=\{P_4\}$, $S_{opt}=\{n_1,\, n_5\}$ is the final solution.

The component parameter randomly varies within its tolerance range (x_i±x_i 10%). Assume that R1=4.4 k, R2=3.1 k, R3=2.8 k, R4=2.9 k, R5=3.3 k, R6=13.1 k, R7=6.2 k, R8=5.4 k, R9=5.8 k, and R10=3.2 k. When the circuit works steadily, the voltages on the two selected test points $t_1$and t_5 are $V_{t_10}=7.756\text{V}$, $V_{t_50}=481.4\text{mV}$, respectively. When R4 fails, R4=2 K, for example, the faulty voltage values are $V_{t_10}=7.728\text{V}$, $V_{t_50}=536.3\text{mV}$. According to expression (1), the fault signature is $A=\frac{0.5363-0.4814}{7.728-7.756}\approx -1.96=\theta=-11.1^0$. Table 11 shows that on the selected plane P_4, $\theta=-11.1^0$ falls into the signature area of R4, viz. -7.1±5°. Hence, faulty component R4 is diagnosed accurately.

In fact, regardless of what type of measurement is carried out; and what kind of failure needs to be diagnosed, if only the fault dictionary can be constructed, then the fault-pair Boolean table is applicable, and the proposed algorithm can be used to select the minimum columns to discriminate the rows.

Table 14 Time complexity

A_1 A_2 A_3 [16]	A_4	A_5[16]	A_6
$O(N_F m N_T \log N_F)$	–	$O(2^{N_T} N_F \log N_F)$	$O\left({N_F}^2 m N_T\right)$

Table 15 Average computation time

	A_1	A_2	A_3	A_4	A_5
Computation time per dictionary (Unit: ms)	60	62	59	327	>1 h

3.2 Statistical Simulation Examples

In the former three examples, the proposed algorithm more accurately finds the minimum test point set than the other methods. In this subsection, its accuracy is also examined on extensive randomly computer-generated fault dictionaries. Besides, for comparison, five latest test point selection algorithms are also examined on the same dictionaries. All the simulations are done by using MATLAB codes in an Intel 1.6 GHz processor, 512 M RAM and the Microsoft Windows XP operating system.

Statistical Simulation Example 1 A total of 100 fault dictionaries are examined. Every dictionary consists of 100

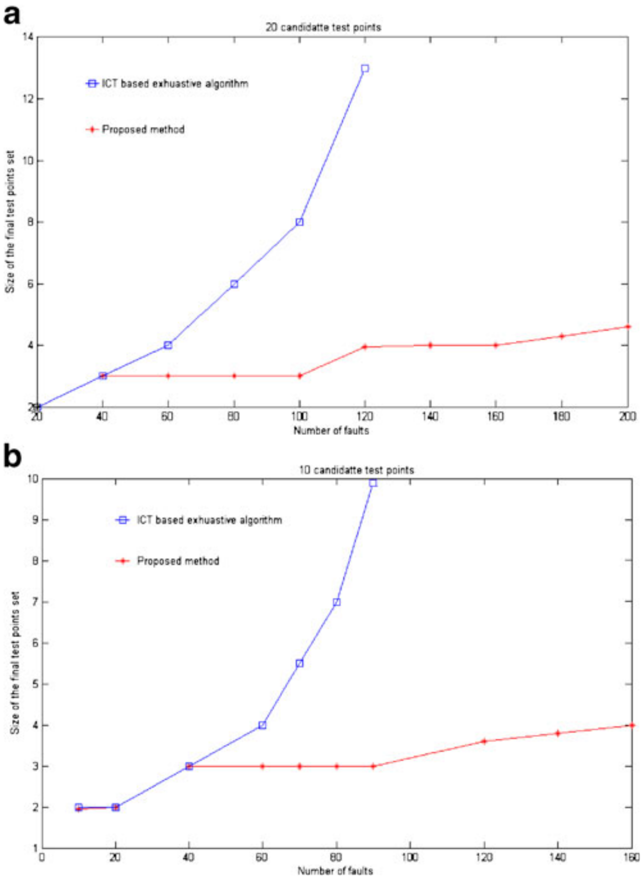

Fig. 4 Statistical results of the proposed method and the exhaustive algorithm based on Integer-Coded technique

simulated faults and 20 test points. Every component of these dictionaries is randomly generated by MATLAB codes and varies between 0.00 and 5.00 volts. First, these dictionaries are transformed into integer-coded and fault-pair tables. Then, the proposed algorithm is carried out on the fault-pair tables, and the other algorithms are carried out on the integer-coded tables. Table 13 shows the simulation results. It can be seen that if the proposed method A_6 is adopted, three test points that isolate all the faults in all 100 simulated fault dictionaries (100% of the simulated cases) are sufficient, however, if the other algorithms (A_1~A_5) including the exhaustive algorithm are used, the sizes of final solutions are larger than 7, which indicates that these solutions contain at least four redundant test points. Besides, in eight simulated cases, the integer-coded technique leads to an erroneous conclusion that faults cannot be isolated fully.

Table 14 shows the time complexity of every algorithm, cited in [16] except for A_6. The time complexity of algorithm A_4 is hard to obtain and can be examined only by statistical simulations. It can be seen that the time complexity of proposed algorithm A_6 is larger than that of A_1, A_2, and A_3; however, it is much smaller than that of A_5. In this example, the complexity of A_6 is $O\left(\frac{{N_F}^2 mN_T}{N_F mN_T \log_2(N_F)}\right) = O\left(\frac{100}{\log_2(100)}\right) \approx 6.7$ times more than that of A_1, A_2 and A_3; however, it is much smaller than that of A_5. For the eight no-solution cases, the average computation time of which is listed in Table 15. In these cases, the time complexity of A_5 is $o(2^{20} \cdot 100\log100)$; therefore, its executing time is larger than 1 h.

Although the execution time of the proposed algorithm is greater than that of other near optimum algorithms, since we do not execute this algorithm in real-time but rather do it offline, the proposed algorithm is still a computationally viable algorithm. For example, the execution time for a 100×20 size fault dictionary is only about 0.2 s.

Statistical Simulation Example 2 Figure 4 illustrates the performance of the proposed algorithm and the ICT-based exhaustive method. This figure shows the relationship between the number of faults and the size of the final solution. In Fig. 4(a), the total number of candidate test points is 20. If the fault number is smaller than 40, no difference exists between the proposed algorithm and the ICT-based exhaustive algorithm. When the fault number is larger than 40, the size of the final solution found by the exhaustive algorithm increases dramatically; however, if the proposed method is adopted, the size of the final solution increases slowly. Besides, when the fault number is larger than 120, the ICT technique makes all the faults undistinguishable. Hence, with an increase in the dictionary size, the ICT technique cannot accurately find the final solution whereas the proposed algorithm exhibits an excellent accuracy. The same conclusion is drawn from Fig. 4(b).

4 Conclusion

Modern densely-loaded circuit boards have posed problems for fault diagnosis with in-circuit testers because only limited physical access to the boards is allowed. Therefore, a method that can select a minimum test point set is badly needed to reduce the number of test points. The recently reported test point selection algorithms that based on the ICT technique are claimed to be applicable to a large fault dictionary. Unfortunately, as pointed out in Section 2, the ICT technique leads to an erroneous conclusion when the fault dictionary is large. Therefore, the integer-coded technique is not applicable to large size fault dictionary. Based on these considerations, the main focus and contributions of this paper are listed as follows:

- A new fault-pair Boolean table technique that can fully take advantage of the fault discrimination power of the test points is proposed.
- A test point selection algorithm that fits for the fault-pair Boolean table is introduced.
- This paper proves that the faulty dictionary technique can be used to diagnose not only catastrophic faults but also parameter faults.

The proposed algorithm is succinct, easy to program and execute. Its accuracy has is proved by theoretical analysis and statistical simulations. Thus, it is particularly applicable to actual circuits and engineering applications.

References

1. Abderrahman, Cerny E, Kaminska B (1996) Optimization-based multifrequency test generation for analog circuits. J Electron Test Theory Application 9:59–73
2. Augusto JS, Almeida C (2006) A tool for test point selection and single fault diagnosis in linear analog circuits. In Proc. XXI Intl. Conf. On Design of Systems and Integrated Systems DCIS'06, Barcelona, Spain
3. Bandler JW, Salama AE (1985) Fault diagnosis of analog circuits. Proc IEEE 73:1279–1325
4. Golonek T, Rutkowski J (2007) Genetic-Algorithm-Based Method for Optimal analog test points selection. IEEE Trans Circuits Syst II, Exp Briefs 54(2):117–121
5. Hochwald W, Bastian JD (1979) A dc approach for analog fault dictionary determination. IEEE Trans Circuits Syst CAS-26:523–529
6. Lin PM, Elcherif YS (1985) Analogue circuits fault dictionary—New approaches and implementation. Inl J Circuit Theory Appl 13:149–172

7. Peng W, Shiyuan Y (2006) A soft fault dictionary method for analog circuit diagnosis based on slope fault mode. Control Autom 22(6):1–23
8. Pinjala KK, Kim BC (2003) An Approach for Selection of Test nodes for Analog Fault diagnosis. Proceedings of the 18th IEEE International Symposium on Defect and Fault Tolerance in VLSI Systems
9. Prasad VC, Babu NSC (2000) Selection of test nodes for analog fault diagnosis in dictionary approach. IEEE Trans Instrum Meas 49:1289–1297
10. Prasad VC, Rao Pinjala SN (1995) Fast algorithms for selection of test nodes of an analog circuit using a generalized fault dictionary approach. Circ Syst Signal Process 14(6):707–724
11. Spaandonk J, Kevenaar T (1996) Iterative test-point selection for analog circuits. In Proc 14th VLSI Test Symp 66–71
12. Starzyk JA, Liu D, Liu Z-H, Nelson DE, Rutkowski JO (2004) Entropy-based optimum test nodes selection for analog fault dictionary techniques. IEEE Trans Instrum Meas 53:754–761
13. Stenbakken GN, Souders TM (1987) Test point selection and testability measure via QR factorization of linear models. IEEE Trans Instrum Meas IM-36(6):406–410
14. Varaprasad KSVL, Patnaik LM, Jamadagni HS, Agrawal VK (2007) A new ATPG technique (ExpoTan) for testiong analog circuits. IEEE Trans Computer-Aided Design 26 (1):189–196
15. Varghese X, Williams JH, Towill DR (1978) Computer-aided feature selection for enhanced analog system fault location. Pattern Recognit 10:265–280
16. Yang C, Tian S, Binglong (2009) Test point selection for analog fault dictionary techniques. J Electron Test Theory Application 25 (2):157–168
17. Yang CL, Tian SL, Long B (2009) Application of heuristic graph search to test point selection for analog fault dictionary techniques. IEEE Trans Instrum Meas 58(7):2145–2158

Chenglin Yang received his B.S. and M.S degrees in computer science from Nanchang Institute of Aeronautic Technology in 2001 and 2006 respectively. He is now a Ph.D. student in the University of Electronic Science and Technology of China (UESTC) and a visiting research student in the Georgia Institute of Technology. His research interests include design for testability, test generation, and analog fault diagnosis.

Shulin Tian received his B.S. and M.S degrees in automation engineering science from the University of Electronic Science and Technology of China in 1989 and 1991 respectively. Currently, he is a professor and the Dean of School of Automation Engineering, UESTC. Majored in Measuring and Testing Technology & Instruments, he has conducted a number of projects in high speed, high precision data acquisition and processing, high speed waveform generation, network/communication testing, testing bus technology and testing system integration. He is a member of National Electronic Measuring Instrument Standardization Technology Committee, and Chief Committeeman of Sichuan Automation Control Committee of Instrument Institute of Sichuan Province, P.R. China.

Bing Long was born in July 1974 in Chengdu, Sichuan, P.R. China. He is currently an associate-professor in School of Automation, UESTC. His research interests include testability analysis and fault diagnosis of electronic device system.

Fang Chen received her BS degree from the Nanchang Hangkong University in 2002 and MS degree from the UESTC in 2009. She is now a teacher of Sichuan Normal University Chengdu College. Her research interests include laser communication and circuit design.