J Electron Test (2009) 25:157–168
DOI 10.1007/s10836-008-5097-8

Test Points Selection for Analog Fault Dictionary Techniques

Chenglin Yang · Shulin Tian · Bing Long

Received: 28 June 2008 / Accepted: 19 December 2008 / Published online: 30 January 2009

Abstract The test points selection problem for analog fault dictionary is researched extensively in many literatures. Various test points selection strategies and criteria for Integer-Coded fault wise table are described and compared in this paper. Firstly, the construction method of Integer-Coded fault wise table for analog fault dictionary is described. Secondly, theory and algorithms associated with these strategies and criteria are reviewed. Thirdly, the time complexity and solution accuracy of existing algorithms are analyzed and compared. Then, a more accurate test points selection strategy is proposed based on the existing strategies. Finally, statistical experiments are carried out and the accuracy and efficiency of different strategies and criteria are compared in a set of comparative tables and figures. Theoretical analysis and statistical experimental results given in this paper can provide an instruction for coding an efficient and accurate test points selection algorithm easily.

Keywords Test points selection · Analog fault dictionary · Integer-Coded fault wise table · Test points selection strategy · Test points selection criterion

Nomenclature

S_{opt}	Desired test points set.
S_c	Candidate test points set.
n_j	Test point j.
N_T	Number of candidate test points.
f_i	Fault i.
N_F	Number of all potential faults (including the nominal case).
k_i	Number of test point (or graph node) that can isolate fault f_i.
q_j	Number of faults that can be isolated by n_j.
S_i^j	Ambiguity set i in test point n_j, defined as:
$S_i^j = \{f_m \mid a_{mj} = i, 0 < m < N_F\}$, where a_{mj}	is element of fault dictionary corresponding to the m th fault (row) and j th test point (column).
F_{ij}	Number of faults in ambiguity set S_i^j.
N_{Aj}	Number of ambiguity sets in n_j or in graph node j.

Responsible Editor: J. W. Sheppard

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

1 Introduction

On the one hand, test points selection is very important task for design for testability (DFT). On the other hand, fault dictionary is an important method of simulation before test (SBT) and this technique is a popular choice for fault diagnosis [8]. So the test points selection problem for fault dictionary are researched extensively in many literatures [1, 3, 5, 6, 8–11, 13, 14–16, 18]. The global minimum set of test points can only be guaranteed by an exhaustive search which is computationally expensive [11, 14]. As pointed out by Augusto et al. [3], the exhaustive search algorithm is limited to small or medium size analog systems. For large or medium systems, such as the dictionaries with more than 40 faults and more than 40 test points, the exhaustive search is impractical. The tradeoff between the desired degree of fault diagnosis and computation cost is to select a near-minimum set (or a local minimum set).

Lin and Elcherif [8] proposed an Integer-Coded fault wise table (or dictionary) technique. This technique is proven to be an effective tool for the optimum test points selection. The later test points selection algorithms [3, 5, 9–11, 14, 18] are all based on the Integer-Coded fault wise table technique. This technique was used to solve test points selection problem for analog fault dictionary technique first. But this technique can be used not only in circuit level fault diagnosis but also in system level fault diagnosis. In fact, the dependency matrix proposed in Deb et al. [4] is a system level binary fault dictionary. Binary fault dictionary is just a special Integer-Coded fault wise table (dictionary) [12, 17]. Don't take the possibility of fault happening and test cost into consideration; sequential fault diagnosis problem is just a test points selection problem. So the test points selection algorithms proposed in literatures [3, 5, 9–11, 14, 18] could be used not only in circuit level but also in board level and system level. Each of these algorithms consists of a test points selection strategy and a test points selection criterion. Although the latest literatures [9, 14, 18] have provide statistical experiments to illustrate their algorithms' superiority, these new test points selection criteria do not always overmatch the previous test points selection criteria as discussed later.

The construction method of Integer-Coded fault wise table for analog fault dictionary is illustrated in this section. In Section 2, existing test points selection algorithms are summarized at first. The accuracy and efficiency of every test points selection strategy and test points selection criterion is summarized and compared in Section 3. A new easy-execute algorithm, which can efficiently and accurately find the minimum test points set, is proposed in this section also. Statistical experiments are carried out in Section 4. Finally, conclusions are given in Section 5.

In fault dictionary, rows represent different faults (including the nominal case) and columns show the available test points. Assume that the voltages shown in Table 1 are obtained when a given net work is analyzed for the nominal condition (f_0) as well as for eight faulty conditions (f_1 to f_8). Ambiguity group is defined that any ***two*** faulty conditions fall into the same ambiguity set if the gap between the voltage values produced by them is less than 0.7 V [6]. But it is necessary to point out that the voltage value span of an ambiguity group may be larger than 0.7 V. In column 2 of Table 1 for example, the voltage value gap produced by f_6 and f_4 is 0.4 V so they belong to one ambiguity set. Similarly, f_4 and f_5 belong to one ambiguity set also. Because f_6 f_4 are not distinguishable and f_4 f_5 are not distinguishable, so f_6 f_4 f_5 are not distinguishable and belong to one ambiguity set although the voltage value gap produced by f_6 and f_5 is 0.9 V, which is larger than 0.7 V. The reader is referred to references [6, 10] for details.

Table 1 Fault dictionary

Faults	n_1	n_2	n_3	n_4
f_0	7.8 v	9.3 v	2.3 v	3.7 v
f_1	6.3 v	6.3 v	4.4 v	7.1 v
f_2	7.9 v	7.5 v	3.4 v	2.6 v
f_3	11.0 v	5.5 v	7.2 v	9.6 v
f_4	5.0 v	4.0 v	1.9 v	5.3 v
f_5	6.0 v	4.5 v	4.8 v	5.6 v
f_6	6.8 v	3.6 v	1.8 v	3.9 v
f_7	9.2 v	4.2 v	5.3 v	4.8 v
f_8	9.0 v	3.9 v	2.1 v	5.0 v

Table 2 indicates the ambiguity sets of Table 1. Lin and Elcherif [8] proposed an Integer-Coded fault wise table (or dictionary) for fault isolation phase. For each fault, a code is generated from the numbers of ambiguity sets of each test point. Considering test point n_1 in Table 2, fault f_4 is coded as "0", faults $\{f_1, f_5, f_6\}$ are coded as "1", faults $\{f_0, f_2\}$ are coded as "2", $\{f_7, f_8\}$ are coded as "3" and f_3 are coded as "4". Then, Table 3 is derived.

In this Integer-Coded fault wise 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 each test point are independent and can be numbered using the same integers without confusion. The Integer-Coded table can also be regarded as an array of "N_F" elements. If all the elements of this array are different, it means that each fault has unique integer number. Hence all faults can be diagnosed. If an entry of this array repeats, it indicates that these faults cannot be separated, with the set of test points used for making this array.

2 Test Points Selection Method

Test points selection methods for faulty dictionary technique can be classified into two categories: inclusive and exclusive [16]. In the inclusive approaches, the optimum set

Table 2 Ambiguity set of Table 1

Ambiguity sets	n_1	n_2	n_3	n_4
0	f_4 (5.0 v)	f_4, f_5, f_6, f_7, f_8 (3.6 v~4.5 v)	f_0, f_4, f_6, f_8 (1.5 v~2.4 v)	f_2 (2.6 v)
1	f_1, f_5, f_6 (6.0 v~6.8 v)	f_3 (5.5 v)	f_2 (3.4 v)	f_0, f_6 (3.7 v~3.9 v)
2	f_0, f_2 (7.8 v~7.9 v)	f_1 (6.3 v)	f_1, f_5, f_7 (4.4 v~5.3 v)	f_4, f_5, f_7, f_8 (4.8 v~5.6 v)
3	f_7, f_8 (9.0 v~9.2 v)	f_2 (7.5 v)	f_3 (7.2 v)	f_1 (7.1 v)
4	f_3 (11.0 v)	f_0 (9.3 v)		f_3 (9.6 v)

of test points (S_{opt}) is initialized to be null, then an "ideal" test point based on a given criterion is added to it. The methods proposed in [6, 8–11, 14, 18] fall into inclusive category. While in exclusive methods, the desired optimum set is initialized to include all available test points at first. Then every test point will be checked one by one and will be deleted if it is redundant. The method proposed by Prasad and Babu [11] fall into exclusive method.

2.1 Inclusive Category

2.1.1 Algorithm 1[10]

Step 1: Initialize S_{opt} as null set and let S_c consist of all candidate test points.
Step 2: Draw out a test point n_j with $\underset{j}{Max}(N_{Aj})$ from S_c and add to S_{opt}.
Step 3: Check whether the selected test points in S_{opt} are enough to diagnose the given set of faults. If yes, stop.
Step 4: If the size of any one ambiguity set determined by S_{opt} is decreased after the inclusion of test point n_j, go to Step 2. Else, discard n_j from S_{opt} and go to Step 2.

Table 3 Integer-coded fault wise table

Faults	n_1	n_2	n_3	n_4
f_0	2	4	0	1
f_1	1	2	2	3
f_2	2	3	1	0
f_3	4	1	3	4
f_4	0	0	0	2
f_5	1	0	2	2
f_6	1	0	0	1
f_7	3	0	2	2
f_8	3	0	0	2

2.1.2 Algorithm 2[3]

Step 1: Initialize S_{opt} as null set and let S_c consist of all candidate test points.
Step 2: Draw out a test point n_j with $\underset{j}{Min}\left(\underset{i}{Max}(F_{ij})\right)$ from S_c and add to S_{opt}.
Step 3: Check whether the selected test points in S_{opt} are enough to diagnose the given set of faults. If yes, stop.
Step 4: If the size of any one ambiguity set determined by S_{opt} is decreased after the inclusion of test point n_j, go to Step 2. Else, discard n_j from S_{opt} and go to Step 2.

2.1.3 Algorithm 3[3]

Step 1: Initialize S_{opt} as null set and let S_c consist of all candidate test points.
Step 2: Draw out a test point n_j with $\underset{j}{Min}(s_j)$ from S_c and add to S_{opt}, where

$$s_j = \sum_{i=1}^{N_{Aj}} \left(F_{ij} - N_F / N_{Aj}\right)^2 / N_{Aj} \tag{1}$$

Step 3: Check whether the selected test points in S_{opt} are enough to diagnose the given set of faults. If yes, stop.
Step 4: If the size of any one ambiguity set determined by S_{opt} is decreased after the inclusion of test point n_j, go to Step 2. Else, discard n_j from S_{opt} and go to Step 2.

2.1.4 Algorithm 4[9]

Step 1: Initialize S_{opt} as null set and let S_c consist of all candidate test points.
Step 2: Draw out a test point n_j with $\underset{j}{Max}(q_j)$ from S_c and add to S_{opt}.

Step 3: Check whether the selected test points in S_{opt} are enough to diagnose the given set of faults. If yes, stop.

Step 4: In fault wise table (dictionary), eliminate rows that can be isolated by test points in S_{opt} together.

Step 5: Calculate q_j for every test point n_j in S_c. Herein, q_j is the number of faults that can be isolated by n_j and test points in S_{opt} together. Go to Step 2.

The second sentence in this Step 5 may be confusing. The following example is used to explain this operation.

Example 1: Considering the test points selection problem of Table 11.

Suppose that n_1 have already added to S_{opt} viz. $S_{opt}=\{n_1\}$ and $S_c=\{n_2,n_3,n_4,n_5\}$. Obviously, all faults can't be isolated by n_1, so additional test points are needed. In Step 5, calculate q_j for every test point n_j in S_c. There is no fault can be isolated by n_2 while n_3 can isolate f_3, n_4 can isolate f_2 and n_5 can isolate f_1. So one may get $q_2=0$, $q_3=q_4=q_5=1$ and draw that n_3,n_4 or n_5 is better than n_2. But when combined with test points in S_{opt}(viz. n_1), all four faults can be isolated by n_2, n_3 can isolate $\{f_2\, f_3\}$, n_4 can isolate $\{f_2\, f_3\}$, also, n_5 can isolate $\{f_0\, f_1\}$. So $q'_2 = 4$, $q'_3 = q'_4 = q'_5 = 2$. Obviously n_2 should be added to S_{opt} according to *Step 2*. In *Step 5*, the q_j value is q' value.

2.1.5 Algorithm 5[2]

Step 1: Initialize S_{opt} as null set and let S_c consist of all candidate test points.

Step 2: Draw out a test point n_j with $\underset{j}{Max}(I_j)$ from S_c and add to S_{opt}, where

$$\begin{aligned} I_j &= -\left(\frac{F_{1j}}{N_F}\ln\frac{F_{1j}}{N_F} + \frac{F_{2j}}{N_F}\ln\frac{F_{2j}}{N_F} + \cdots \frac{F_{N_{Aj}j}}{N_F}\ln\frac{F_{N_{Aj}j}}{N_F}\right) \\ &= \frac{\ln N_F}{N_F}\sum_{i=1}^{N_{Aj}} F_{ij} - \frac{1}{N_F}\sum_{i=1}^{N_{Aj}} F_{ij}\ln F_{ij} \\ &= \ln N_F - \frac{1}{N_F}\sum_{i=1}^{N_{Aj}} F_{ij}\ln F_{ij} \end{aligned} \tag{2}$$

Step 3: Check whether the selected test points in S_{opt} are enough to diagnose the given set of faults. If yes, stop.

Step 4: Partition the rows of the dictionary according to the ambiguity sets of $S_{opt}$and removing the rows that can be isolated by test points in S_{opt} together.

Step 5: Calculate I_j for every test point in S_c. Go to Step 2.

2.1.6 Algorithm 6[13]

In Yang et al. [18], a heuristic Depth-first graph search algorithm is developed to solve the test points selection problem. Root node S contain all faults that to be diagnosed. First, use all candidate test points to expand S and the new generated graph nodes belong to Level 1. Second, on Level 1, select a graph node with maximum fault discrimination power as the one to be expanded for the second iteration. Finally, expand the selected graph node with all the test points except those used on the path from this node to the root node S. The new generated graph nodes belong to Level 2. Execute these steps iteratively until all faults are isolated. An example is used to illustrate this method.

Figure 1 is the expanded graph for Table 3. In this figure, x_i represents graph node and n_j is the test point used on the graph path. Every intermediate graph node x_i consists of two faults group. One group consists of isolated faults and the other consists of residual faults. Residual faults are subdivided into several ambiguity sets as shown in Fig. 1.

Initially let the graph G consist of the root node S. All potential faults (f_0~f_8) are contained in node S and belong to one ambiguity set (viz. N_{AS}=1). The root node S is expanded with all test points and the new four graph nodes (x_1~x_4) are shown on Level 1 in Fig. 1. Based on Level 1, Table 4 list the k_i (number of graph node that can isolate fault f_i) value and information content of every fault, where information content is defined as:

$$I(f_i) = -\ln\left(\frac{k_i}{N_T}\right) \tag{3}$$

Faults f_3 and f_4 are isolated by graph node x_1, so information content contained in x_1 is $I(x_1)=I(f_3)+I(f_4)$. Information content for every graph node on Level 1 is shown in Table 5. Node x_2 has the maximum information content, so it should be expanded in next step. Because test point n_2 has been used on the path from root node to x_2, the residual three test points $\{n_1,n_3,n_4\}$ are used to expand node x_2 on the second level of Fig. 1.

Test points used on the path from goal node x_8 to root node S make up of the final solution.

Obviously, each algorithm has its own test points selection criterion. One can call these criteria as fault discrimination power.

Criterion1 $\underset{j}{Max}(N_{Aj})$

Criterion2 $\underset{j}{Min}\left(\underset{i}{Max}(F_{ij})\right)$

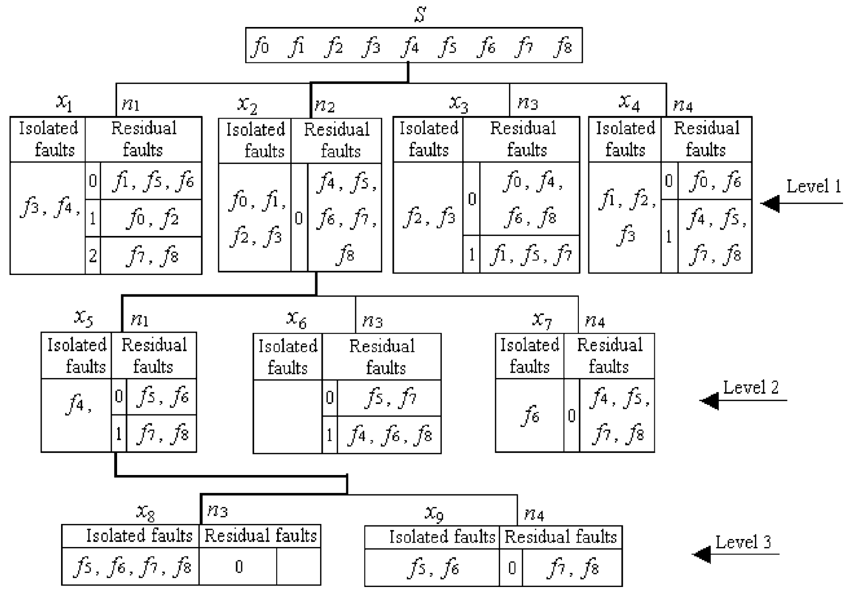
Criterion3 $\underset{j}{Min}(s_j)$

Criterion4 $\underset{j}{Max}(q_j)$

Criterion5 $\underset{j}{Max}(I_j)$

Criterion6 $\underset{j}{Max}(I(x_j))$

Fig. 1 Expanded graph



2.2 Exclusive Category [3]

There are six candidate test points selection criteria for exclusive category also.

Criterion7 $\underset{j}{Min}\left(N_{Aj}\right)$

Criterion8 $\underset{j}{Max}\left(\underset{i}{Max}\left(F_{ij}\right)\right)$

Criterion9 $\underset{j}{Max}\left(s_j\right)$

Criterion10 $\underset{j}{Min}\left(q_j\right)$

Criterion11 $\underset{j}{Min}\left(I_j\right)$

Criterion12 $\underset{j}{Min}\left(I\left(x_j\right)\right)$

Step 1: Let S_{opt} consist of all candidate test points.

Step 2: Draw out a test point n_j from S_{opt} by using any one of the exclusive criteria.

Step 3: Check whether the residual test points in S_{opt} are enough to diagnose the given set of faults. If yes, go to Step 2. Else, add n_j to S_{opt}.

Step 4: Repeat Step 2 to Step 3, till all test points are examined.

3 Algorithm Analysis and Statistical Experiments

3.1 Accuracy Analysis

Before the accuracy analysis, an example is considered at first.

Example 2: A band pass filter circuit example

Figure 2 is a band pass filter circuit, which is cited by literatures [3, 5, 9–11, 14, 18] as a benchmark circuit. The excitation signal is a 1 kHz, 4 V sinusoidal wave. Totally, there are eighteen potential catastrophic faults f_1~f_{18} and eleven test points n_1~n_{11}. Voltage values at all nodes for different faulty conditions are obtained by simulation and the Integer-Coded fault wise table is constructed by procedures introduced in Section 1. The results are shown in Table 6. All seven strategies mentioned in Section 2 (six inclusive strategies and an exclusive strategy) are used to select optimum test points respectively. All the simulations are done by using Visual C++6.0 codes. The results are listed in Table 7.

The global minimum test points set is found by Algorithm 5, Algorithm 6 and Exclusive category. By using Algorithm 4, the final solution contains a redundant test point n_7. Algorithm 1, Algorithm 2 and Algorithm 3 are even worse, since about three redundant test points are contained in the final solution. Obviously, the latter three inclusive strategies (and exclusive category) can more accurately find the optimum solution than the former three strategies. This conclusion was also proved by statistical experiments in literatures [9, 14, 18].

Because each algorithm (except for Exclusive category) has its own test points selection criterion, one may naturally conclude that the later three criteria (4~6) is better than the former criteria (1~3) based on the statistical

Table 4 k_i value and information content of every fault (N_T=4)

Faults	f_0	f_1	f_2	f_3	f_4	f_5~f_8
k_i	1 (x_2)	2 (x_2, x_4)	3 (x_2~x_4)	4 (x_1~x_4)	1 (x_1)	0
$I(f_i)$	1.386	0.693	0.288	0	1.386	–

Table 5 Fault discrimination power of every test point

Test points	x_1	x_2	x_3	x_4
$I(n_i)$	1.386	2.367	0.288	0.981

experiment results. But this is *not* the fact as discussed in the following text.

3.1.1 Inclusive Category

As pointed out in Section 2, in the inclusive approaches, the optimum set of test points (S_{opt}) is initialized to be null, then an "ideal" test points based on a given criterion is added to it. So test points selection criterion is an essential ingredient of inclusive algorithm. All the inclusive algorithms are greedy heuristic algorithms. Although each criterion evaluates the fault discrimination power of a test point from different point of view, but they are all just greedy heuristic functions and there is no theoretical proof can be offered to demonstrate a specific criterion's optimality.

Besides test points selection criterion, test points selection strategy is another important ingredient of inclusive algorithm. A good test points selection strategy can improve the accuracy of a searching algorithm. For Algorithms 1, 2 and 3 mentioned in Section 2, the most important strategy lies in Step 4 and is called as Strategy 1 in this paper. According to Strategy 1, although a test point is evaluated as the best one in S_c, it will not be added to S_{opt} if it can not subdivide ambiguity sets determined by S_{opt}. In Example 2, after n_8, n_{11}, n_7 and n_5 are added to S_{opt}, n_6 has the maximum fault discrimination power among the test points in S_c. By using Criterion 1, n_6 is concluded as the best one and added to S_{opt}. Table 8 is the ambiguity sets information decided by the test points n_8, n_{11}, n_7 and n_5 together. Table 9 is the ambiguity sets information decided by test points n_8, n_{11}, n_7, n_5 and n_6 together. Obviously, Table 8 and Table 9 have the same ambiguity sets information, say the new selected test point n_6 has no contribution to S_{opt}. In other words, n_6 is redundant test point for S_{opt}. By using Strategy 1, n_6 will be eliminated from S_{opt}, so Strategy 1 can reduce the number of redundant test points of final solution set.

Fault discrimination power of S_{opt} is determined by the all test points contained in it together. So simply adding a test point with best C_i to S_{opt} a maximum information increment of S_{opt} is not guaranteed and then global minimum test points set is less likely to be obtained. To overcome this shortcoming, Algorithm 4 provides another strategy in Step5 called as Strategy 2 in this paper. By using Strategy 2, fault discrimination power of a test point n_j is determined not by itself but by n_j and all test points in S_{opt} together. So if a test point n_j is to be added to S_{opt}, a maximum fault discrimination power increase of S_{opt} is guaranteed. In fact, Strategy 2 is also included in Algorithm 5 and Algorithm 6. In Algorithm 5, step 4 is the embodiment of Strategy 2. In Algorithm 6, graph node expanding process is based on the ambiguity sets information determined by S_{opt}, where S_{opt} consists of test points used on the path from current graph node to root node. So the selected node guarantees the maximum information increase to S_{opt}. Besides Strategy 2, Algorithms 4, 5 and 6 remove the faults that can be isolated by S_{opt} in each iteration. This can reduce these algorithms' time and space complexities and is called as Strategy 3. Strategy 2 can further reduce redundant test points and in turn improve the solution accuracy. If Algorithms 1, 2 and 3 adopt Strategy 2 and 3, they can also more accurately find the optimum test point set. In Example 2, the new results are listed in Table 10. Compared to Table 7, improved algorithms (with Strategy 2 and Strategy 3) can more accurately find the optimum test point set.

Although Strategy 2 can reduce the number of redundant test points, but the final solution may still have redundant test points. For example, there are still one or two redundant test points in final solutions in Table 10. As mentioned above, in the inclusive approaches, the optimum set of test points (S_{opt}) is initialized to be null, then test point is added to it one by one. Although the new selected

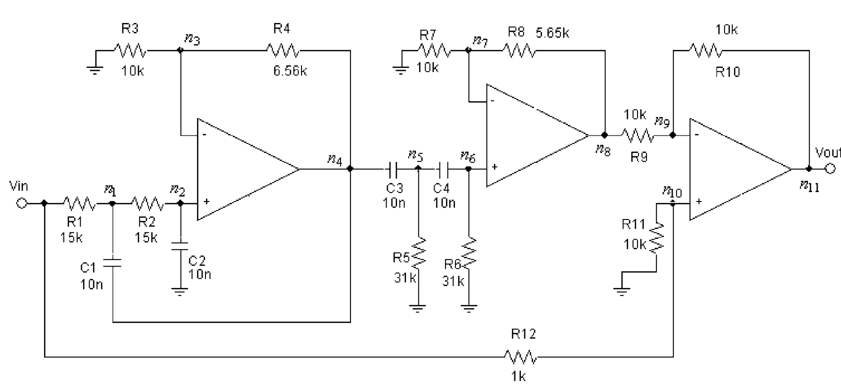


Fig. 2 A band pass filter circuit

Table 6 Integer-coded fault wise table for analog filter

Faults	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)	3	2	2	3	3	3	4	4	1	1	1
f_1(R1 open)	0	0	0	0	0	0	0	0	1	1	7
f_2(R1 short)	3	2	2	3	4	3	4	4	1	1	0
f_3(R2 open)	1	0	0	0	0	0	0	0	1	1	7
f_4(R2 short)	2	3	3	4	6	5	6	6	1	1	2
f_5(R3 open)	1	1	1	1	1	1	1	1	1	1	5
f_6(R4 open)	0	0	0	2	3	2	3	3	1	1	8
f_7(R5 open)	3	2	2	3	5	4	5	5	1	1	4
f_8(R5 short)	3	2	2	3	0	0	0	0	1	1	7
f_9(R6 open)	3	2	2	3	6	6	7	7	1	1	6
f_{10}(R6 short)	3	2	2	3	2	0	0	0	1	1	7
f_{11}(R7 open)	3	2	2	3	3	3	4	2	1	1	4
f_{12}(R7 short)	3	2	2	3	3	2	0	8	5	1	8
f_{13}(R8 open)	3	2	2	3	3	2	2	8	4	1	8
f_{14}(R9 open)	3	2	2	3	3	3	4	4	1	1	0
f_{15}(R9 short)	3	2	2	3	3	3	4	4	2	1	8
f_{16}(R10 open)	3	2	2	3	3	3	4	4	3	1	8
f_{17}(R11 open)	3	2	2	3	3	3	4	4	2	2	3
f_{18}(R12 open)	3	2	2	3	3	3	4	4	0	0	1

test point is not redundant for those selected test points, it cannot be guaranteed that one or more former selected test points is not redundant for the new set S_{opt}. In Table 10 for example, the final solution found by improved Algorithm 1 is $\{n_1,n_5,n_8,n_9,n_{11}\}$. The selected order is $n_8 \rightarrow n_9 \rightarrow n_5 \rightarrow n_1 \rightarrow n_{11}$. At first S_{opt} is initialized to be null, then test point n_8 is added to S_{opt}. Obviously n_8 is not redundant. But after n_9,n_5,n_1,n_{11} are added to S_{opt}, n_8 is a redundant test point for S_{opt} because that all faults can be isolated by n_9,n_5, n_1,n_{11} without n_8. So if only an algorithm belong to inclusive category, it cannot guarantee that there is no redundant test point contained in the final solution.

3.1.2 Exclusive Category

As mentioned in Section 2, exclusive algorithms examine all test points. Anyone redundant will be eliminated from S_{opt}. So the final solution found by exclusive category must have no redundant test points. It is needed to point out that a set without redundant test points does not mean that this set is a minimum set. An example is used to illustrate this viewpoint. Table 11 is a simple fault wise table. Both set $S_{opt1}=\{n_1,n_2\}$ and set $S_{opt2}=\{n_3,n_4,n_5\}$ can isolate all faults and each set contains no redundant test points. Obviously, S_{opt2} is not the minimum set.

For a practical complex system, there are many candidate test points (N_T) in the design stage, but only a few physical test points (m) can be set, normally $N_T > 2\,m$.

Table 7 Final solution found by algorithms

Algorithm	Final solutions
Algorithm 1	$n_1,n_4,n_5,n_7,n_8,n_9,n_{11}$
Algorithm 2	$n_1,n_5,n_7,n_8,n_9,n_{11}$
Algorithm 3	$n_1,n_4,n_5,n_7,n_8,n_9,n_{11}$
Algorithm 4	n_1,n_5,n_7,n_9,n_{11}
Algorithm 5	n_1,n_5,n_9,n_{11}
Algorithm 6	n_1,n_5,n_9,n_{11}
Exclusive category	n_1,n_5,n_9,n_{11}

Table 8 Ambiguity sets information decided by $\{n_8,n_{11},n_7,n_5\}$

Ambiguity set	Faults	n_8	n_{11}	n_7	n_5
0	f_1(R1 open)	0	7	0	0
	f_3(R2 open)	0	7	0	0
	f_8(R5 short)	0	7	0	0
1	f_{10}(R6 short)	0	7	0	2
2	f_5(R3 open)	1	5	1	1
3	f_{11}(R7 open)	2	4	4	3
4	f_6(R4 open)	3	8	3	3
5	f_{14}(R9 open)	4	0	4	3
6	f_2(R1 short)	4	0	4	4
7	f_0 (NOM)	4	1	4	3
	f_{18}(R12 open)	4	1	4	3
8	f_{17}(R11 open)	4	3	4	3
9	f_{15}(R9 short)	4	8	4	3
	f_{16}(R10 open)	4	8	4	3
10	F_7(R5 open)	5	4	5	5
11	f_4(R2 short)	6	2	6	6
12	f_9(R6 open)	7	6	7	6
13	f_2(R7 short)	8	8	0	3
14	f_{13}(R8 open)	8	8	2	3

Table 9 Ambiguity sets information decided by $\{n_8,n_{11},n_7,n_5,n_6\}$

Ambiguity Set	Faults	n_8	n_{11}	n_7	n_5	n_6
0	f_1(R1 open)	0	7	0	0	0
	f_3(R2 open)	0	7	0	0	0
	f_8(R5 short)	0	7	0	0	0
1	f_{10}(R6 short)	0	7	0	2	0
2	f_5(R3 open)	1	5	1	1	1
3	f_{11}(R7 open)	2	4	4	3	3
4	f_6(R4 open)	3	8	3	3	2
5	f_{14}(R9 open)	4	0	4	3	3
6	f_2(R1 short)	4	0	4	4	3
7	f_0 (NOM)	4	1	4	3	3
	f_{18}(R12 open)	4	1	4	3	3
8	f_{17}(R11 open)	4	3	4	3	3
9	f_{15}(R9 short)	4	8	4	3	3
	f_{16}(R10 open)	4	8	4	3	3
10	f_7(R5 open)	5	4	5	5	4
11	f_4(R2 short)	6	2	6	6	5
12	f_9(R6 open)	7	6	7	6	6
13	f_{12}(R7 short)	8	8	0	3	2
14	f_{13}(R8 open)	8	8	2	3	2

Table 10 final solution found by improved algorithms

Improved algorithms	Final solutions
Improved algorithm 1	n_1,n_5,n_8,n_9,n_{11}
Improved algorithm 2	n_1,n_5,n_6,n_9,n_{11}
Improved algorithm 3	n_1,n_2,n_5,n_6,n_9,n_{11}

As discussed above, exclusive method takes more time when $N_T > 2m$.

3.1.3 Combined Category

Based on the former discussion, considering test points set selection problem for a large-scale system, inclusive method is more efficient than exclusive method while exclusive method is more accurate than inclusive method.

To efficiently and accurately find the optimum test points set, an inclusive algorithm is executed at first, then exclusive algorithm is used to eliminate redundant test points in S_{opt}. This method is called as combined category. Table 12 lists the final solutions found by inclusive algorithms (Strategy 2 and Strategy 3 is adopted by every algorithm) and combined algorithms.

From Table 12, it can be seen that no matter which inclusive algorithm is executed, if only exclusive method is appended, the final solution contains no redundant test points. The combined method found the minimum set in this example.

3.2 Time Complexity

The complexities of Algorithm 1, Algorithm 2 and Algorithm 3 are dominated by Strategy 1. They all have the complexity of $O(N_F p \log N_F)$ [11], where p is the number of test points examined and $m \leq p \leq N_T$ $\left(m = \left|S_{opt}\right|\right)$.

The complexity of Algorithms 4, 5 and 6 are dominated by Strategy 2. The process of calculating fault discriminate power q_j for every test point in S_c is the same with the process of checking whether a number repeats or not. There exist several efficient sequential algorithms such as quick sort, heap sort etc. [2][7] with time complexity of $O(N_F \log N_F)$ for calculating faulty discriminate power for one test point. Algorithms 4, 5 and 6 examine every test point in S_c for each iteration, so these algorithms have the time complexity of $O(N_F p' \log N_F)$, where $p' = N_T + (N_T - 1) + \cdots (N_T - m + 1)$. Because the case of $N_T > 2m$ is considered in this paper, there will be $p' >> m$. So the total time complexity of Algorithms 4, 5 and 6 is $O(N_F m N_T \log N_F)$.

Exclusive category has the complexity of $O(N_F(p+m) \log N_F)$, where m is the number of test points contained in S_{opt}. As pointed out in Prasad et al. [11],

$$N_T > 2m; \text{ exclusive method takes more time.} \quad (1)$$

$$N_T > 2m; \text{ exclusive method takes less time.} \quad (1)$$

As discussed above, the time complexity of inclusive algorithm is $O(N_F p \log N_F)$ or $O(N_F p' \log N_F)$. Suppose that the final solution found by an inclusive algorithm contains m test points, using exclusive method to examine all test points in S_{opt} will take $O(N_F m \log N_F)$ complexity. So the time complexity of combined method is

$$O(N_F p \log N_F) \ + \ O(N_F m \log N_F)$$

$$\text{or } O(N_F p' \log N_F) \ + \ O(N_F m \log N_F)$$

Because $p' >> m$ and $p >> m$, the total time complexity of combined algorithm is still $O(N_F p \log N_F)$ or $O(N_F p' \log N_F)$.

The following three conclusions can be drawn according to the theoretical analysis in this Section.

Conclusion 1: Every test points selection criterion is just greedy heuristic function and there is no theoretical proof can be offered to demonstrate a specific criterion's optimality. Any specific criterion has no superior in the final solution to others.

Conclusion 2: Strategy 2 is superior in the accuracy to Strategy 1 while Strategy 1 is more efficient than Strategy 2.

Conclusion 3: When combined with exclusive strategy, either Strategy 1 or Strategy 2 can improve

Table 11 A simple fault wise table

Faults	n_1	n_2	n_3	n_4	n_5
f_0 (NOM)	0	0	0	0	0
f_1(R1 open)	0	1	0	0	1
f_2(R1 short)	1	0	0	1	0
f_3(R2 open)	1	1	1	0	0

Table 12 Solutions found by original inclusive and combined algorithms

Algorithms	Solutions found by inclusive algorithms	Solutions found by combined algorithms
Algorithm 1	n_1,n_5,n_8,n_9,n_{11}	n_1,n_5,n_9,n_{11}
Algorithm 2	n_1,n_5,n_6,n_9,n_{11}	n_1,n_5,n_9,n_{11}
Algorithm 3	n_1,n_2,n_5,n_6,n_9,n_{11}	n_1,n_5,n_9,n_{11}
Algorithm 4	n_1,n_5,n_7,n_9,n_{11}	n_1,n_5,n_9,n_{11}
Algorithm 5	n_1,n_5,n_9,n_{11}	n_1,n_5,n_9,n_{11}
Algorithm 6	n_1,n_5,n_9,n_{11}	n_1,n_5,n_9,n_{11}

the accuracy of the final solution and do not increase the time complexity.

Besides the theoretical proof, the statistical experiments are also provided to prove the above conclusions in Section 4.

4 Statistical Experiments

Statistical experiments are carried out on the randomly computer-generated Integer-Coded dictionaries by using the inclusive category (adopting six criteria, Strategy 1 and Strategy 2 respectively) and combined category respectively. All the simulations are done by using Visual C++ 6.0 codes. All the execution times in this paper are obtained using an Intel 1.6 GHz processor, 512 M RAM and Microsoft Windows XP operating system.

Experiment 1: Totally, there are 100 Integer-Coded fault wise tables and each table has 1,000 simulated faults, 100 test points, and six ambiguity sets per test point.

Table 13 lists the results of Strategy 1 by using six different criteria respectively. It can be seen that by using Strategy 1 and Criterion 1 (the second column of Table 13), 17 simulated fault wise tables (17% of the simulated cases) need seven test points to isolated all faults, 61 simulated fault wise tables need eight test points to isolated all faults, 22 simulated fault wise tables need nine test points to isolated all faults. Comparing different columns of Table 13, one can see that the size of final solutions found by six different criteria are all varying between seven and nine. In about 14–19% of the simulated fault wise tables, the size of final solutions is seven. In most (about 60–71%) of the simulated cases, the final solutions consist of eight test points. In the rest (about 15–22%) of the simulated cases, the size of the final solutions is nine. So it can be drawn that if the same strategy is adopted, the accuracy of final solutions found by different criteria have no dramatic differences (viz. in the accuracy of finding final solution, no one specific criterion is superior to others). Besides the accuracy, the computation time of different criterion has no notable difference also. This gives that in the time efficiency, no one specific criterion is superior to the others. This conclusion can also be drawn according to Tables 14, 15 and 16.

Table 13 Statistical results of inclusive category (by using strategy 1)

Size of the min. set	Inclusive category (by using strategy 1)					
	C_1	C_2	C_3	C_4	C_5	C_6
7	17	19	17	17	14	14
8	61	60	66	61	71	71
9	22	21	17	22	15	15
Computation time per dictionary (unit: ms)	187.5	188.1	199.9	193.1	190.0	207.0

C_i represent i-th criterion

Table 14 Statistical results of combined category (by using strategy 1 and exclusive category)

Size of the min. set	Combined category (strategy 1+exclusive category)					
	C_1	C_2	C_3	C_4	C_5	C_6
7	83	88	90	84	89	90
8	17	12	10	16	11	10
Computation time per dictionary (unit: ms)	211.4	214.6	228.0	217.7	218.9	220.5

By comparing column 2 of Table 14 to column 2 of Table 13, it can be seen that the combined category improves the accuracy of final solutions substantially. By using combined category and Criterion 1, the size of final solutions is no more than 8. While if inclusive category and the same criterion are adopted, one will draw that there are 22 simulated cases need nine test points to isolate all faults as shown in Table 13. Additionally, to isolate all faults, 83% of the simulated dictionaries only need seven test points by using combined category while 83% (61%+22%) of the simulated dictionaries need eight or nine test points by using inclusive category. In Fig. 3, Criterion 3 is the common criterion adopted by Strategy 1 and Combined category. In 66% of the simulated cases, it is concluded that eight test points are necessary by using strategy 1. However in most (90%) of the simulated cases, it is concluded that only seven test points are enough to isolate all faults. Additionally, in 17% of the simulated cases the size of final solution found by Strategy 1 is 9 while the size of final solution is no more than 8 if the Combined category is used.

So it can be drawn that combined category has superiority in the final solutions' accuracy to inclusive category. Comparing the other corresponding columns of

Table 15 Statistical results of inclusive category (by using strategy 2)

Size of the min. set	Inclusive category (By using Strategy 2)					
	C_1	C_2	C_3	C_4	C_5	C_6
6	5	2	5	3	5	6
7	95	98	95	97	95	94
Computation time per dictionary (unit: s)	1.03	1.34	1.32	1.30	1.32	1.05

Table 16 Statistical results of combined category (by using strategy 2 and exclusive category)

Size of the min. set	Combined category (strategy 2+exclusive category)					
	C_1	C_2	C_3	C_4	C_5	C_6
6	5	3	6	3	6	6
7	95	97	94	97	94	94
Computation time per dictionary (unit: s)	1.04	1.34	1.33	1.32	1.33	1.10

Tables 13 and 14 (viz. the other five criteria are adopted respectively), the same conclusion can be drawn.

Computation time per dictionary is averagely about 195 *ms* by using inclusive category and 218 *ms* by using combined category. So the combined category need little more time than inclusive category. This conclusion is consistent with the former time complexity analysis given in Section 3.2. So it can be drawn that the combined category is superior to the inclusive categories. This conclusion can also be drawn by comparing Table 15 to Table 16.

From Table 15 it can be seen that the size of final solutions found by Strategy 2 is no more than seven while more than 80% of simulated cases need eight or nine test points by using Strategy 1 in Table 13. So Strategy 2 can improve the solutions' accuracy dramatically and this conclusion is consistent with the former accuracy analysis given in Section 3.1.2. Figure 4 is used to illustrate this conclusion.

The size (number of selected test points) of the final solutions found by Strategy 2 is no more than 7 while the the size of the final solutions found by Strategy 1 is no less

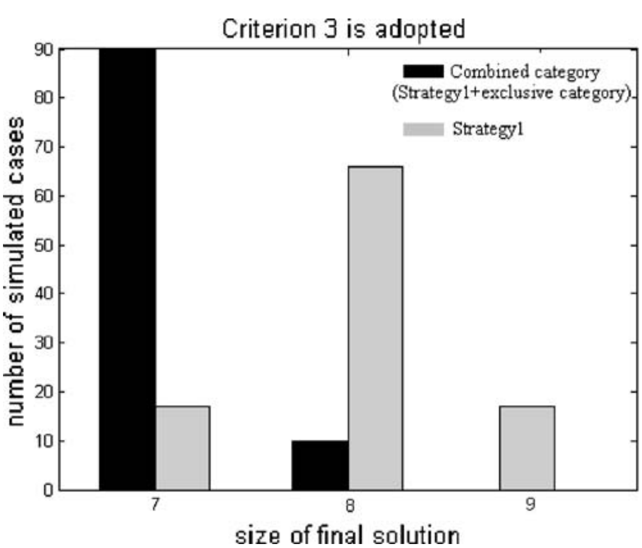


Fig. 3 Statistical results of strategy 1 and combined category

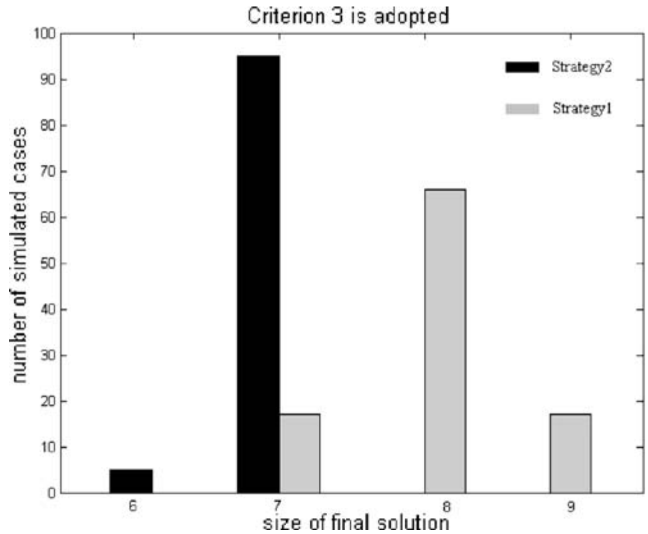


Fig. 4 Statistical results of strategy 1 and strategy 2

than 7. Obviously, Strategy 2 can improve the accuracy of final solution substantially. The same conclusion can be drawn when the other criteria are adopted respectively (by comparing other corresponding columns of Tables 13 and 15).

At the same time the computation time of Strategy 2 is about six times as much as that of Strategy 1. The same conclusion can also be drawn by comparing Table 14 to Table 15. This is consistent with former theoretical analysis. So Strategy 2 is superior in solution accuracy to Strategy 1 while Strategy 1 is superior in time efficiency to Strategy 2.

It can be argued that the ambiguity sets number of a fault wise table is seldom a constant. For example, the ambiguity sets contained in each test points in Table 6 vary in number from 3 to 9. So to validate former conclusions, another set of simulations are carried out in this paper.

Experiment 2: Totally, there are 100 Integer-Coded fault wise tables and each table has 1,000 simulated faults, 40 test points. Ambiguity

Table 17 Statistical results of inclusive category (by using strategy 1)

Size of the min. set	Inclusive category (by using strategy 1)					
	C_1	C_2	C_3	C_4	C_5	C_6
6	11	35	34	7	37	23
7	61	59	54	66	53	64
8	27	6	12	27	10	12
9	1					1
Computation time per dictionary (unit: ms)	125.6	116.0	120.5	124.9	114.4	149.0

Table 18 Statistical results of combined category (by using strategy 1 and exclusive category)

Size of the min. set	Combined category (strategy 1+exclusive category)					
	C_1	C_2	C_3	C_4	C_5	C_6
5			1			
6	55	95	96	43	95	87
7	45	5	3	57	5	13
Computation time per dictionary (unit: ms)	146.8	132.8	140.1	150.2	132.5	169.4

sets number per test point can vary form 1 to 10 randomly.

Table 17 lists the results of inclusive category by using six different criteria respectively. Comparing the different columns of this table, one cannot say which specific criterion is better than the others. This proves the conclusion that no one specific criterion is superior to the others again.

By comparing Table 17 with Table 18 or by comparing Table 19 with Table 20, it can be drawn that the combined category can improve the solution accuracy without taking too much additional computation time. Similarly, conclusion that Strategy 2 is superior to Strategy 1 in solution accuracy can also be drawn by comparing Table 17 with Table 19 or by comparing Table 18 with Table 20.

5 Conclusion

Heuristic test points selection methods for faulty dictionary technique can be classified into two categories: inclusive and exclusive. Either in inclusive category or in exclusive category, every greedy search algorithm consists of a test points selection strategy and a test points selection criterion. The time complexity and solution accuracy of these algorithms, strategies and criteria are analyzed and compared extensively in this paper. Theoretical analysis and statistical experimental results give the following three conclusions.

Conclusion 1: In the accuracy and efficiency of finding final solution, no one specific criterion is superior to the others.

Conclusion 2: Strategy 2 is superior in the accuracy to Strategy 1 while Strategy 1 is more time efficient than Strategy 2.

Conclusion 3: When combined with exclusive strategy, either Strategy 1 or Strategy 2 can improve the accuracy of the final solution and does not increase the time complexity.

Above conclusions provide an instruction for coding a test points selection algorithm. *Conclusion* 1 shows that any criterion can be adopted without affecting the accuracy and the efficiency of an algorithm. So a criterion that can be calculated easily, such as Criterion 1, is suggested to be chosen. Based on *Conclusion* 2 and *Conclusion* 3, if an efficient algorithm is required, Strategy 1 plus exclusive strategy is a good choice. If the solution's accuracy is required, Strategy 2 plus exclusive strategy should be adopted. No matter Strategy 1 or Strategy 2 are adopted, the combined strategy proposed in this paper can guaranty that the final solution contain no redundant test points.

Additionally, conclusions drawn in this paper show that a new strategy rather than a new criterion should be focused on in future work.

Table 19 Statistical results of inclusive category (by using strategy 2)

Size of the min. set	Inclusive category (by using strategy 2)					
	C_1	C_2	C_3	C_4	C_5	C_6
5	1		1			
6	99	73	95	100	100	97
7		24	4			3
8		3	1			
Computation time per dictionary (unit: ms)	511.9	537.3	520.4	507.2	528.9	483.4

Table 20 Statistical results of combined category (by using strategy 2 and exclusive category)

Size of the min. set	Combined category (strategy 2+exclusive category)					
	C_1	C_2	C_3	C_4	C_5	C_6
5	1		1			
6	99	90	99	100	100	97
7		10				3
Computation time per dictionary (unit: ms)	569.5	590.5	581.6	563.1	572.7	520.5

Acknowledgments This work was supported in part by Commission of Science Technology and Industry for National Defense of China (No.A1420061264), and by RFDP(The Research Fund for the Doctoral Program of Higher Education)(No. 20070614018).

References

1. Abderrahman A, Cerny E, Kaminska B (1996) Optimization-based multifrequency test generation for analog circuits. J. Journal of Electronic Testing: Theory and Applications 9:59–73 doi:10.1007/BF00137565
2. Aho A, Hopcroft J, Ullman J (1974) The design and analysis of computer algorithms. Reading, MA, Addison-Wesley
3. Augusto JS, Almeida C (2006) A tool for test points 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
4. Deb S, Pattipati KR, Raghavan V, Shakeri M, & Shrestha R (1995) Multi-signal flow graphs : A novel approach for system testability analysis and fault diagnosis. IEEE Aerospace and Electronics Magazine
5. Golonek T, Rutkowski J (2007) Genetic-algorithm-based method for optimal analog test points selection. IEEE Transactions on Circuits and Systems—II. Express Briefs 54(2):117–121 doi:10.1109/TCSII.2006.884112
6. Hochwald W, Bastian JD (1979) A dc approach for analog fault dictionary determination. IEEE Transactions on Circuits and Systems CAS-26:523–529 doi:10.1109/TCS.1979.1084665
7. Horowitz E, Sahni S (1978) Fundamentals of computer algorithms. Potomac, MD, Computer Science
8. Lin PM, Elcherif YS (1985) Analogue circuits fault dictionary—New approaches and implementation. Inl J Circuit Theory Appl 13:149–172 doi:10.1002/cta.4490130205
9. 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 3:278–294
10. Prasad VC, Rao Pinjala SN (1995) Fast algorithms for selection of test nodes of an analog circuit using a generalized fault dictionary approach. Circuits Syst Signal Process 14(6):707–724 doi:10.1007/BF01204680
11. Prasad VC, Babu NSC (2000) Selection of test nodes for analog fault diagnosis in dictionary approach. IEEE Transactions on Instrumentation and Measurement 49:1289–1297 doi:10.1109/19.893273
12. Shakeri M, Raghavan V, Pattipati KR, Patterson-Hine A (2000) Sequential testing algorithms for multiple fault diagnosis. IEEE Trans SMC Part A: systems humans 30:1–14 doi:10.1109/3468.823474
13. Spaandonk J, & Kevenaar T (1996) Iterative test-point selection for analog circuits. in Proc. 14th VLSI Test Symp 66–71
14. Starzyk JA, Liu D, Liu Z-H, Nelson DE, Rutkowski JO (2004) Entropy-based optimum test nodes selection for analog fault dictionary techniques. IEEE Transactions on Instrumentation and Measurement 53:754–761 doi:10.1109/TIM.2004.827085
15. Stenbakken GN, Souders TM (1987) Test points selection and testability measure via QR factorization of linear models. IEEE Transactions on Instrumentation and Measurement 36(6):406–410
16. Varghese X, Williams JH, Towill DR (1978) Computer-aided feature selection for enhanced analog system fault location. Pattern Recognition 10:265–280 doi:10.1016/0031-3203(78)90036-5
17. Wang W, Hu QH, & Yu D (2007) Application of Multivalued Test Sequencing to fault diagnosis. ICEMI
18. Yang CL, Tian SL, & Long B Application of Heuristic Graph Search to Test Points Selection for Analog Fault Dictionary Techniques. To be published in IEEE Trans Instrum Meas in June 2009

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). His research interests include design for testability, test generation, and dependable computing.

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.