Analog system-level fault diagnosis based on a symbolic method in the frequency domain

Citation for published version (APA):

DOI:
10.1109/19.368107

Document status and date:
Published: 01/01/1995

Document Version:
Publisher’s PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:

• A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
• The final author version and the galley proof are versions of the publication after peer review.
• The final published version features the final layout of the paper including the volume, issue and page numbers.

Link to publication

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

• Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
• You may not further distribute the material or use it for any profit-making activity or commercial gain
• You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the “Taverne” license above, please follow below link for the End User Agreement:
www.tue.nl/taverne

Take down policy
If you believe that this document breaches copyright please contact us at:
openaccess@tue.nl
providing details and we will investigate your claim.
Analog System-Level Fault Diagnosis Based on a Symbolic Method in the Frequency Domain

Zhihong You, E. Sánchez-Sinencio, Fellow, IEEE, and Jose Pineda de Gyvez, Member, IEEE

Abstract—Concurrently, a symbolic approach for analog system-level fault diagnosis and a systematic approach to maximize the fault location capability are proposed. This unified approach is realized as a result of combining the simulation before test (SBT) fault dictionary diagnosis method with a symbolic approach. The traditional SBT fault dictionary method is often costly and inefficient because of a high number of simulations, but it can become very efficient when a symbolic approach is employed. This symbolic approach only requires one analysis for circuit topology to generate the network symbolic approach. The traditional SBT fault dictionary method is employed.

1) Relations between input and output signals in analog circuits are sometimes more complicated as compared to those in digital circuits. These relations in analog circuits are more difficult to model than digital circuit representations, which are based on classical truth tables, and thus, are precise and easy to model.

2) The number of I/O’s in analog circuits is small. This implies that analog circuit fault diagnosis requires a great amount of computer time and computer storage as the number of observable/controllable pins is small.

3) There are some inherent interactions between various circuit parameters, which will impede efficient functional verification and diagnosis.

There are many problem formulations depending on the systems under test, e.g., fault models, measurement conditions and the final objective of the fault diagnosis [5]. The key factors in the classification of diagnostic problems are the following:

1) Systems under test [1], [5]
   a) Elements contained in the system: RLC or RC; controlled sources; operational amplifiers (OP AMP); operational transconductance amplifiers (OTA); switches and nonlinear elements.
   b) Excitation of the systems: DC excitation; sinusoidal excitation, which includes single frequency or multifrequency.
   c) Number of exciting sources (independent sources or inputs): Single exciting sources; multiple exciting sources.

2) Fault models [4], [5]
   By a fault we mean, in general, any change in the value of an element with respect to its nominal value which can cause the failure of the whole circuit. The fault could be catastrophic (HARD fault) when the faulty element produces either a short circuit or an open circuit, or parametric (SOFT fault) when the faulty element just deviates from its nominal value for a specific design tolerance range. These soft faults could result from manufacturing tolerances, aging, or parasitic effects.
   Several fault location techniques only address the cases when just one parameter causes the fault. This is referred to as a single fault. In practice the multiple-fault case, which is caused by several faulty parameters, could occur, and recent techniques address this as a difficult case [1], [2].
3) Measurements

The measurements can be carried out in DC, frequency or time domains. Within the different domains, voltage measurements or current measurements can be chosen.

4) Final objective of fault diagnosis [3], [4], [6]

In general, analog circuit fault diagnosis has three main objectives: fault detection, fault location and fault identification. Fault detection is the minimum requirement for the fault diagnosis; it answers WHETHER an analog circuit's functionality falls outside design specification. Fault location is the necessary requirement for repairing faulty parts at a later time; it addresses WHERE the faults are. Fault identification is necessary for tuning and adjustment; it answers WHAT the exact value of the deviation is. For these purposes, many fault diagnosis ideas have been developed. The major characteristics of these ideas can be classified into the pattern recognition, parameter identification, fault verification, and approximation techniques.

Two basic methods of analog fault diagnosis are the simulation-before-test (SBT) and the simulation-after-test (SAT) methods [2]. The SBT techniques use a pattern recognition concept [2], which is implemented by simulating the circuit with the possible fault conditions before the actual testing; the results are stored in look-up tables. It can effectively detect and locate hard faults. The fault simulation plays a very important role in this approach. The application of SBT techniques is usually limited to circuits of small or moderate size for the detection of only hard faults. The fault dictionary technique is an SBT method.

All parameter identification and fault verification techniques belong to the SAT approach, which simulates the circuit at the testing time or after the testing time. Parameter identification techniques identify all circuit parameters under the assumption that enough independent measurements are available [7], [8]. These techniques are classified as linear or nonlinear according to the nature of the diagnosis equations. Fault verification techniques can only identify a few faulty elements under the assumption that a few elements are faulty and the rest of the network elements are within design tolerances. Based on this idea, several algorithms have been developed such as the substitution theorem-based technique, the failure bound technique and the network decomposition approach [1], [9]. In the SAT approach there have been various methods which may locate single or multiple, hard or soft faults in circuits of small or relatively large size.

The motivation of the research hereby presented is to improve the traditional SBT fault dictionary approach. It is known that the traditional SBT fault dictionary technique is very time consuming and costly because a high number of numerical simulations are required to construct the fault dictionaries of the system under test. But, it might be rendered more useful if an efficient approach could be found to generate fault dictionaries in reasonable amounts of computer time. For this reason, a symbolic approach is introduced during the fault dictionary generation.

The proposed approach uses the symbolic simulator (ASAP) [13] to generate the requested network functions of the circuit under test [15], and then only expression evaluations are required to obtain all the possible faulty responses. Compared to a traditional approach, which uses a numerical simulator (SPICE) to obtain the faulty responses, the proposed approach is more efficient and less costly. It can also deal with single/multiple faults, soft/hard faults at the same time. This reduction in computation time to evaluate the faulty responses is a key contribution of this paper. The other practical contribution is the systematic approach to maximize the fault location capability.

II. PROPOSED FAULT DIAGNOSIS APPROACH

Nowadays, two main classes of simulators exist. One class is the numerical simulator, such as SPICE, which can be used for nonlinear dc, nonlinear transient and linear ac analysis problems. Even though it is very efficient, it only returns a collection of numbers, in tabular or plot forms. The other class is the symbolic simulator. It can yield the network function in which some, or all, of the circuit elements, $p_1, p_2, \ldots, p_m$, along with the complex frequency $s$, are represented by symbolic parameters, as shown in (1)

$$H(s, p_1, p_2, \ldots, p_m) \equiv \frac{N(s, p_1, p_2, \ldots, p_m)}{D(s, p_1, p_2, \ldots, p_m)}. \quad (1)$$

The symbolic simulation has noteworthy advantages with respect to the numerical simulation for those applications which require the repetition of a high number of simulations performed on the same circuit topology with the variation of component values and/or frequency [14]-[15]. In these types of applications, the symbolic approach can be used to generate the requested network functions of the analyzed circuit in parametric form. In this way, the circuit analysis is performed only once and, during the fault dictionary generation phase, only a parameter substitution and an expression evaluation are required to obtain the numerical results. Evidently, the function evaluation is much easier and faster than working repeatedly with a network analysis program to perform a full ac analysis.

The flow diagram of the proposed symbolic fault dictionary diagnosis approach is shown in Fig. 1. The dashed lines are
Fig. 2. OTA-C leapfrog lowpass filter.

Fig. 3. A simple macromodel for an OTA.

related to SPICE simulations. All these parts (dashed arrows) are only used to verify the new symbolic fault dictionary approach during the research stage. In actual use, SPICE will not be used.

For a system under test, it is necessary to have macromodels for dealing with the system-level faults. These macromodels focus on building-block performance parameters of OP AMP’s and OTA’s rather than on transistor-level parameters.

Next, a detailed explanation of Fig. 1 is given. Once the macromodels are generated, the symbolic simulator, ASAP, is used to generate the network transfer function, $H(s)$, in which some, or all, of the circuit elements, along with the complex frequency, are represented by symbolic parameters. At the same time, the system-level faults are determined based on the macromodel circuit, and the initial testing frequencies are chosen according to the Seshu and Waxman theory [2]. Then, a parameter substitution and an expression evaluation are performed to obtain the required numerical frequency responses. The output frequency responses, under all faulty conditions, form a recognition matrix. Each column of the recognition matrix is related to a testing frequency point, and each row of the matrix is related to a specific faulty case. An optimum testing-point selection technique [18] is used to make the new approach self-contained. The testing-point selection is a choice with the minimum testing frequency points to maximize the fault diagnosability. Based on the optimum testing points, the fault dictionary can be generated automatically. At this point, the off-line computation is finished.

During the on-line computation, the nearest neighbor rule is used to locate the faults. This rule is based on the idea that a faulty case which has the minimum Euclidean distance with the testing data will be the most likely fault. For an analog system, there are several accessible nodes which can provide different information for fault diagnosis. This fact leads us to propose a systematic fault location approach, which is developed to maximize the fault location capability.

System-level faults are those faults that do not consider actual transistors.

An OTA-C leapfrog lowpass filter is chosen as an example to illustrate the proposed fault diagnosis symbolic method. The topology of the OTA-C leapfrog lowpass filter is shown in Fig. 2. The real OTA used in the OTA-C filter is a tunable OTA [11], which uses the voltage $Vc$ to control the transconductance $g_m$ of the OTA, for $i = 1, \ldots, 5$.

This approach can address system-level faults, which focus on the behavioral performance of the OP AMP and the OTA. To deal with system-level faults, simple macromodels for the OP AMP and the OTA need to be built. These macromodels should include typical parameters: namely, gain, dominant pole, input impedance and output impedance, to accurately represent the performance of the circuit or subcircuit under test for all faulty conditions.

For the OTA-C leapfrog lowpass filter, the typical one-dominant-pole macromodel is shown in Fig. 3. Here, $G_1(Vc)$ = $G_m$ represents the transconductance gain; the dominant transconductance pole is $\omega_p = 1/(R_{dp} \times C_{dp})$, $G_{buf} = 1$ and; $R_{in}$ and $R_{out}$ characterize the input and output resistances, respectively. When this simple, one-dominant-pole macromodel is included in a closed-loop configuration, it matches well with the real OTA at the transistor level. Fig. 4 shows the comparison between the OTA-C leapfrog lowpass filter with the real OTA and with its macromodel.

Based on the above discussion, it is evident that the proposed symbolic fault diagnosis approach can save the off-line computation time invested in the fault dictionary generation. Undoubtedly, this proposed approach can be more efficient and practical than conventional techniques. Also, fault location capabilities can be maximized using a systematic technique.

### III. Symbolic Fault Diagnosis

#### A. System Under Test

In this presentation, we address continuous-time, linear, time-invariant systems, which are composed of lumped passive elements and active devices. These systems can be characterized by (2) or (3) in the Laplace domain:

$$H(s) = \frac{Y(s)}{X(s)} = \frac{a_m s^m + a_{m-1} s^{m-1} + \cdots + a_0}{b_0 s^n + b_{n-1} s^{n-1} + \cdots + b_0} \quad (2)$$

$$H(s) = \frac{Y(s)}{X(s)} = \frac{a_m (s - Z_1) (s - Z_2) \cdots (s - Z_m)}{b_n (s - p_1) (s - p_2) \cdots (s - p_n)} \quad (3)$$

An OTA-C leapfrog lowpass filter is chosen as an example to illustrate the proposed fault diagnosis symbolic method. The topology of the OTA-C leapfrog lowpass filter is shown in Fig. 2. The real OTA used in the OTA-C filter is a tunable OTA [11], which uses the voltage $Vc$, to control the transconductance $g_m$ of the OTA, for $i = 1, \ldots, 5$.

This approach can address system-level faults, which focus on the behavioral performance of the OP AMP and the OTA. To deal with system-level faults, simple macromodels for the OP AMP and the OTA need to be built. These macromodels should include typical parameters: namely, gain, dominant pole, input impedance and output impedance, to accurately represent the performance of the circuit or subcircuit under test for all faulty conditions.

For the OTA-C leapfrog lowpass filter, the typical one-dominant-pole macromodel is shown in Fig. 3. Here, $G_1(Vc)$ = $G_m$ represents the transconductance gain; the dominant transconductance pole is $\omega_p = 1/(R_{dp} \times C_{dp})$, $G_{buf} = 1$ and; $R_{in}$ and $R_{out}$ characterize the input and output resistances, respectively. When this simple, one-dominant-pole macromodel is included in a closed-loop configuration, it matches well with the real OTA at the transistor level. Fig. 4 shows the comparison between the OTA-C leapfrog lowpass filter with the real OTA and with its macromodel.

A more complex OTA macromodel could be used for higher frequency applications.

---

1 System-level faults are those faults that do not consider actual transistors.

2 A more complex OTA macromodel could be used for higher frequency applications.
Table I

<table>
<thead>
<tr>
<th>Table I</th>
<th>COMPARISON OF THE EXTRACTED POLES/ZEROS FROM ASAP AND SPICE FOR THE OTA-C LEAPFROG LOWPASS FILTER</th>
</tr>
</thead>
<tbody>
<tr>
<td>Poles/Zeros from ASAP (rad/sec)</td>
<td>Poles/Zeros from SPICE (rad/sec)</td>
</tr>
<tr>
<td><strong>Symbolic pole extraction</strong></td>
<td><strong>Symbolic pole extraction</strong></td>
</tr>
<tr>
<td>root[0] = -488.841e+3 + 1.239e+6 i</td>
<td>root[0] = -488.838e+3 + 1.239e+6 i</td>
</tr>
<tr>
<td>root[1] = -488.841e+3 - 1.239e+6 i</td>
<td>root[1] = -488.838e+3 - 1.239e+6 i</td>
</tr>
<tr>
<td><strong>Symbolic zero extraction</strong></td>
<td><strong>Symbolic zero extraction</strong></td>
</tr>
<tr>
<td>root[0] = -666.6667e+6 + 21.838 i</td>
<td>root[0] = -666.6667e+6 + 0 i</td>
</tr>
</tbody>
</table>

with the real OTA subcircuit and with its macromodel. The standard deviation of the difference between the two responses is 0.053. The CPU time used for the simulation with the OTA macromodel is about 10 times faster than for the simulation with the OTA at the transistor level.

B. Feasibility Analysis of Using ASAP to Substitute for SPICE

One of the main advantages of the proposed fault diagnosis approach is to use a symbolic simulator to substitute for the use of numerical simulations during the generation of the fault dictionaries. This will reduce the computational time and make the approach more practical. However, it is necessary to analyze the feasibility of using symbolic simulations to substitute for numerical simulations. We present two sufficient and necessary conditions to do the feasibility analysis.

**Condition 1: Poles and Zeros Comparison**

Each continuous-time linear time-invariant system can be represented by (3), which is the transfer function characterized by the poles and zeros, and concurrently all these poles/zeros are functions of some, or all, circuit parameters. Parameter deviations change the locations of the poles and zeros, and consequently change the magnitude of the transfer function. This fact implies that the feasibility analysis of using a symbolic simulator can be verified by comparing the numerically and symbolically extracted poles and zeros. Table I shows the comparison of poles/zeros from ASAP and HSPICE for the OTA-C leapfrog lowpass filter. The results show that there are some differences between the larger poles and zeros. This is due to the fact that we are using a simple OTA macromodel. However, the critical poles and zeros of this system are matching very well with a standard deviations of less than 1.0e-6, see Fig. 4. Based on the fact that mainly the critical poles and zeros affect the system performance, one can conclude that good matching between the poles and zeros from ASAP and SPICE is obtained.

**Condition 2: Numerical Frequency Response Comparison**

The symbolic transfer function from ASAP can be represented by (2). When s = jω, the magnitude frequency responses can be calculated by (4), shown at the bottom of the page.

Here, “INT” means integer, and all ai and bj are functions of the circuit parameters.

The flow chart for calculating the frequency response of the system under test is shown in Fig. 5. Two libraries are set to automate the calculation. One is the element library, which includes all possible elements of the system under test. They are capacitors, resistors, OP AMP’s and OTA’s. The other library is the faulty case library, which includes several possible faulty cases for each kind of element. For example, each OP AMP has several faulty cases, namely, voltage gain deviations from their nominal value, input impedance reductions, and output impedance changes to open circuits. For a given system under test, the faulty cases can be generated automatically by searching the faulty case library. Using the

\[ |H(j\omega)| = \left( \frac{\sum_{i=1}^{(N/2)} (-1)^i a_{2i} \omega^{2i}}{\sum_{i=1}^{(N/2)} (\omega^{2i-1} a_{2i-1} \omega^{2i-1})} \right)^{1/2} \] (4)
faulty cases and the testing frequency points, the evaluation of the symbolic transfer function can be done to obtain the frequency responses of the system.

The comparison between ASAP calculations and SPICE simulations using the OTA-C leapfrog lowpass filter are shown in Fig. 6. The standard deviation of the difference is about 0.0003. These results verify that the symbolic simulator can be used to substitute for SPICE numerical simulations.

C. Optimum Testing-Point Selection

One of the most important aspects in the design of fault diagnosis schemes is the testing-point selection. With proper and efficient testing-point selection, diagnosability of faulty components can be increased considerably. Improper or ineffective testing points will increase the number of frequency points needed for testing, thereby increasing the testing time and/or computational facilities required. It will also increase confusion and errors in diagnosability [2]. Testing-point selection [23] is the optimal choice of the minimum number of measurement points while maintaining and/or maximizing the probability of correct diagnosis. The initial testing features are chosen according to the Seshu and Waxman theory, which requires that there must be at least one test break point below the "lowest nonzero" break point, and one between the successive break points [2], [18]. The rationale behind this choice is that all poles and zeros of the network are functions of the network parameters. Parameter deviations cause changes in the locations of the poles and zeros, and consequently change the magnitude of the transfer function. Here the "lowest nonzero" and the "highest finite" frequency break points depend on the particular poles/zeros distribution and the range of frequencies over which the system is active.

A heuristic optimum testing-point selection, proposed in [18], is used in this symbolic approach for the sake of completeness. This procedure selects a set of testing points without performing the exhaustive search of all subsets of frequency points.

For a given circuit, the fault dictionaries for some accessible nodes are built and stored in a dictionary library. After obtaining measurement data, the corresponding fault dictionary is chosen. Then, the nearest neighbor rule is used to locate the most likely faulty case based on the minimum "distance" concept. Possible faulty cases which fall within the tolerance window can be identified and grouped together. If the fault family only has one faulty case, the process is finished; if not, a different dictionary for a different node is chosen, and another fault family is obtained. See Fig. 7 for a flow chart of this procedure. The same procedure is carried out k times, where k is the number of accessible nodes. The final fault location result can be determined as follows:

Fault = fault family (node 1) ∩ fault family (node 2) ∩ ... fault family (node k). (5)

This approach is similar to the ambiguity sets concept introduced in [24]. Rather than using fixed constraints to isolate the faults, the method we implemented makes use of tolerance windows. That is to say, the output responses are considered within a certain standard deviation. This allows us to consider the effect of component tolerances in the final result. Obviously, the wider this window is, the more fault cases it can include. But on the other hand, a window allows taking into account the regular variations of the fault-free response. Cases in which an element is considered faulty under the fixed-constraint value approach, may actually not be faulty if they fall within the tolerance window of the fault-free response. Furthermore, our fault diagnosis approach uses a symbolic analysis not used in [24].
TABLE II
INITIAL TESTING-POINT SET FOR THE OTA-C LEAPFROG LOWPASS FILTER (kHz)

<table>
<thead>
<tr>
<th>No.</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>150</td>
<td>200</td>
<td>250</td>
<td>300</td>
<td>350</td>
<td>400</td>
<td></td>
</tr>
<tr>
<td>450</td>
<td>500</td>
<td>550</td>
<td>600</td>
<td>650</td>
<td>700</td>
<td></td>
</tr>
<tr>
<td>750</td>
<td>800</td>
<td>850</td>
<td>900</td>
<td>950</td>
<td>1000</td>
<td></td>
</tr>
<tr>
<td>3250</td>
<td>3500</td>
<td>3750</td>
<td>4000</td>
<td>4250</td>
<td>4500</td>
<td></td>
</tr>
</tbody>
</table>

TABLE III
OPTIMUM TESTING-POINT SET FOR THE OTA-C LEAPFROG LOWPASS FILTER (kHz)

<table>
<thead>
<tr>
<th>Node</th>
<th>Optimum Testing Point Set</th>
<th>No. of Points</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>10 50 100 150 200 250 300 350 400 450</td>
<td>15</td>
</tr>
<tr>
<td>2</td>
<td>4 6 9 150 250 350 500 750 1000 3250</td>
<td>6</td>
</tr>
<tr>
<td>3</td>
<td>2 50 100 150 200 250 300 3250 10000</td>
<td>9</td>
</tr>
</tbody>
</table>

D. Diagnosis Results

Based on the rationale presented before, the fault dictionary should be built for each accessible node of the circuit. After the fault dictionaries of all accessible nodes in the circuit are built, the off-line computation has been finished. Then, the on-line computation of testing data can be carried out to locate the faults.

Step 1: Off-Line Computation Results: Based on Seshu and Waxman’s theory, the initial testing-point set for the OTA-C leapfrog lowpass filter is shown in Table II, which totally includes 34 frequency points. Table III shows the optimum frequency testing-point set for each accessible node in the OTA-C leapfrog lowpass filter, which is shown in Fig. 2. After the optimum testing-point sets are determined, the fault dictionary for each node is generated automatically.

Step 2: On-Line Computation Results: In this presentation, two kinds of testing data are discussed. One of them, generated from SPICE simulations using the real OTA, is used to test the theoretical feasibility of the proposed solution.

In the data set, 38 faults were generated consisting of short circuits, opens, and excessive parameter value variations. The other testing data was obtained from actual measurements of a test chip to verify the practicality.

1) Using SPICE Simulation Data: For the OTA-C leapfrog lowpass filter, the simulation data is generated from the filter with the real OTA subcircuit. Some diagnosis results are shown in Table IV.

In an analog system, the output response has its specification tolerance range because of the circuit parameter tolerance and the inaccuracies in the measurement [19]. The tolerance column in Table IV shows the specification tolerance range (also called tolerance window).

2) Using Measurement Data of OTA-C Leapfrog Lowpass Filter: In order to verify the practicality of this proposed symbolic fault diagnosis approach, the actual testing data of the OTA-C leapfrog lowpass filter is used. During the on-line testing, several possible faulty cases are produced. For a capacitor, a fault can be produced by adding or reducing one or more capacitors to make its value change. For an OTA, the control voltage can be used to adjust the gain to a faulty case. After a certain faulty case is generated, the magnitude frequency responses under the optimum feature points are tested for each accessible node, and then diagnosed by using the corresponding dictionary.

The fault case F1, corresponding to a 50% variation of C1 from its nominal value, is chosen to illustrate the approach. Fig. 8 shows the magnitude frequency response for node 1 under the optimum feature set, which has 15 frequency points. By using the fault location approach with the +40% specification tolerance window, the most likely faulty case and its fault family are obtained. Here, the diagnosis result shows that the most likely faulty case is F1, and the fault family includes F1, F5, F9, and F32. Diagnosis results for the OTA-C leapfrog lowpass filter from actual measurement data are shown in Table V. These results verify the feasibility of the proposed approach.

IV. CONCLUSIONS

A new symbolic method based on a conventional analog system fault diagnosis approach has been presented. This approach combines the SBT fault dictionary diagnosis method with a symbolic simulator, ASAP, to locate system-level faults
It becomes very efficient when a symbolic approach is used. This approach permits us to automatically obtain the optimum conventional SBT fault dictionary method can be costly and in continuous-time, linear, and time-invariant systems. The systematic approach has been developed to maximize the fault recognition matrix, respectively. It is clear that the new symbolic approach can deal with hardsoft faults at the same time editing the ASAP symbolic results and to generate the initial line computation compared to the conventional SBT fault dictionary method.

Table VI shows that the symbolic approach is more than 20 times faster than the conventional method. In Table VI, \( t_1 \) is the time needed for editing SPICE results to generate the recognition matrix. A general comparison between the common SBT fault dictionary method and this new approach is shown in Table VII.

**ACKNOWLEDGMENT**

The authors want to thank the Analog Microelectronics Group from the Centro Nacional Microelectronica, Seville, and in particular, Dr. Angel Rodríguez-Vázquez for giving us access to a preliminary version of ASAP.

**REFERENCES**


**TABLE V**

<table>
<thead>
<tr>
<th>Faulty case of testing data</th>
<th>Program diagnosis results</th>
<th>Tolerance</th>
<th>Node</th>
<th>Fault families</th>
<th>Final result</th>
</tr>
</thead>
<tbody>
<tr>
<td>F0</td>
<td>F0, F16, F25</td>
<td>±30%</td>
<td>1</td>
<td>F0, F9, F16, F22, F30, F31</td>
<td>F0</td>
</tr>
<tr>
<td></td>
<td>F1, F5, F14</td>
<td>±40%</td>
<td>2</td>
<td>F1, F9, F14</td>
<td>F1</td>
</tr>
<tr>
<td></td>
<td>F14</td>
<td>±20%</td>
<td>3</td>
<td>F14</td>
<td>F14</td>
</tr>
<tr>
<td></td>
<td>F18</td>
<td>±20%</td>
<td>4</td>
<td>F18, F16, F24, F16, F37, F12, F16, F35</td>
<td>F18</td>
</tr>
<tr>
<td></td>
<td>F27</td>
<td>±30%</td>
<td>5</td>
<td>F27, F26, F27</td>
<td>F27</td>
</tr>
</tbody>
</table>

**TABLE VI**

<table>
<thead>
<tr>
<th>Symbolic method</th>
<th>On-line computation</th>
<th>Off-line computation</th>
<th>Test nodes</th>
<th>Types of faults</th>
<th>Network type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASAP</td>
<td>minimal</td>
<td>high</td>
<td>limited</td>
<td>single</td>
<td>linear/</td>
</tr>
<tr>
<td>HSPICE</td>
<td>moderate</td>
<td>limited</td>
<td>limited</td>
<td>multiple</td>
<td>linear/</td>
</tr>
</tbody>
</table>

**TABLE VII**

<table>
<thead>
<tr>
<th>General fault dictionary</th>
<th>New symbolic approach</th>
</tr>
</thead>
<tbody>
<tr>
<td>minimal</td>
<td>minimal</td>
</tr>
<tr>
<td>high</td>
<td>moderate</td>
</tr>
<tr>
<td>limited</td>
<td>limited</td>
</tr>
<tr>
<td>single</td>
<td>multiple</td>
</tr>
<tr>
<td>linear</td>
<td>linear</td>
</tr>
<tr>
<td>nonlinear</td>
<td>general</td>
</tr>
</tbody>
</table>

The CPU time for off-line computation (OTA-C leapfrog lowpass filter) is shown in Table VI. The above CPU times are measured for the OTA-C leapfrog lowpass filter. Its initial recognition matrix includes 38 faulty cases, which means that 38 SPICE simulations have to be done when using the conventional SBT fault dictionary method. It becomes very efficient when a symbolic approach is used. The above CPU times are measured for the OTA-C leapfrog lowpass filter. Its initial recognition matrix includes 38 faulty cases, which means that 38 SPICE simulations have to be done when using the conventional SBT fault dictionary method. Table VI shows that the symbolic approach is more than 20 times faster than the conventional method. In Table VI, \( t_1 \) is the time needed for editing SPICE results to generate the recognition matrix. A general comparison between the common SBT fault dictionary method and this new approach is shown in Table VII.

**ACKNOWLEDGMENT**

The authors want to thank the Analog Microelectronics Group from the Centro Nacional Microelectronica, Seville, and in particular, Dr. Angel Rodríguez-Vázquez for giving us access to a preliminary version of ASAP.

**REFERENCES**


Zhiyong You was born in Beijing, People’s Republic of China. She received the B.S. and M.S. degrees in electrical engineering from Tsinghua University, in 1987 and 1989, respectively. She received the M.Sc. degree from Texas A&M University, in 1993. From 1989 to 1990, she worked in China Technology Company, as an Engineer. She then worked in China-American Bausch-Lomb Contact Lens Inc., as a Manager Assistant. She is currently employed by Texas Instruments, Inc., Dallas, TX.

Edgar Sánchez-Sinencio (S’72–M’74–SM’83–F’92) was born October 27, 1944, in Mexico City, Mexico. He received the B.Sc. degree in communications and electronic engineering from the National Polytechnic Institute of Mexico, the M.Sc. degree from Stanford University, Stanford, CA, and the Ph.D. degree from the University of Illinois at Champaign-Urbana, in 1966, 1970, and 1973, respectively.

He was a member of the Board of Governors, IEEE Circuits and Systems Society from 1990-1992, serving as Chairman in 1991 and Chairman Elect. in 1990; the IEEE CAS Technical Committee on Neural Systems and Applications; Associate Editor of the IEEE TRANS. ON CIRCUITS AND SYSTEMS from 1985 to 1987, and the IEEE TRANS. ON NEURAL NETWORKS from 1990–1993; Associate Editor of the IEEE CIRCUITS AND DEVICES MAGAZINE (1985–1988); General Chairman of the 26th Midwest Symposium CAS in Puebla, Mexico (1983); a member IEEE CAS Technical Committee on Analog Signal Processing; Associate Editor, IEEE CAS MAGAZINE (1983–1984); and Guest Editor, Special Issues on Neural Network Circuit Implementation, IEEE TRANS. ON NEURAL NETWORKS, March 1991 and May 1992.

Jose Pineda de Gyvez (S’88–M’90–S’91–M’91) received the degree in electronic systems engineering from the Technological Institute of Monterrey, Monterrey, Mexico, the M.Sc. degree from the National Institute of Astrophysics Optics and Electronics, Mexico, and the Ph.D. from Eindhoven University of Technology, Eindhoven, The Netherlands, in 1982, 1984, and 1991, respectively.

He was a Junior Scientist with the Foundation for Fundamental Research on Matter, The Netherlands, from 1986 to 1991, working on CAD for yield, defect, and fault modeling. He is currently an Assistant Professor in the Department of Electrica Engineering and holds a joint Faculty appointment with the Department of Computer Science, Texas A&M University, College Station. He is the author of *Integrated Circuit Defect Sensitivity: Theory and Computational Models* (Kluwer).

Dr. Pineda is a member of the ACM and IEEE Circuits and Systems Society.