A Method to Diagnose Faults in Linear Analog Circuits Using an Adaptive Tester

Érika F. Cota Luigi Carro Marcelo Lubaszewski

Departamento de Engenharia Elétrica
Universidade Federal do Rio Grande do Sul
Av. Oswaldo Aranha 103 - CEP 90035-190 - Porto Alegre - BRASIL
{erika, carro, luba }@iee.ufrgs.br

Abstract

This work presents a new diagnosis method for use in an adaptive analog tester. The tester is able to detect faults in any linear circuit by learning a reference behavior in a first step, and comparing this behavior against the output of the circuit under test in a second step. Considering the same basic structure, the diagnosis method consists on injecting probable faults in a mathematical model of the circuit, and later comparing its output with the output of the real faulty circuit. This method has been successfully applied to a case study, a biquad filter. Component soft, large, and hard deviations, and faults in operational amplifiers were considered. The results obtained from practical experiments with this analog circuit are discussed in the paper.

1. Introduction

In the last few years, many works have addressed the problems of fault detection and fault diagnosis in analog circuits. Different searching processes exist today that aim at identifying those test stimuli that can minimize testing time, maximize fault coverage, or both. For example, sensitivity analysis is used in [6] in order to find the input frequencies that maximize the error output for every kind of fault in a component (soft, large and hard faults). If diagnosis is not a concern, the individual frequency subsets can be merged together, in such a way that a smaller set of test stimuli is obtained. [2] also applies sensitivity analysis to analog circuits, but generates a minimal subset of input-parameter pairs that simultaneously maximizes fault detection and fault diagnosis.

In [1] a test method is proposed that is based on two steps (figure 1). The first step consists on a *learning phase*, where a digital *adaptive filter* learns the behavior of a fault-free circuit. The second step consists on a *test phase*, where the output of the filter is compared with the output of the circuit under test (*CUT*). When this difference exceeds a given threshold, then a fault is detected. This method uses white noise as input stimulus for both the adaptive filter and the CUT. The error output obtained is in general very sensitive, even for deviations of component values as small as 1%. This is a low cost method, suitable for production testing, that achieves very high fault coverage. However, since the obtained error output does not bring much information about the fault location, fault diagnosis is yet to provide in this test environment. This work attempts to bring a solution for that problem.

This paper is structured as follows: Session 2 briefly discusses existing approaches for analog fault diagnosis and introduces the diagnosis strategy used in the analog tester. Session 3 gives an overview of the conceptual model proposed for the diagnosis procedure. Session 4 describes the first practical implementation of the method, shows up the preliminary results obtained with a biquad filter, and discusses the limitations of the approach. Finally, session 5 concludes the paper.

2 Analog fault diagnosis

In terms of analog diagnosis, existing tools follow two basic approaches [3]: Simulation Before Test (SBT) and Simulation After Test (SAT). In the SBT method, the design is analyzed before testing, and the output of a faulty circuit is stored in a fault dictionary. Thus, if a fault is detected, it can be located by matching the actual result against the dictionary. For instance, [4] proposes a tool based on the SBT method. In the SAT approach, on the other hand, a mathematical analysis is only performed after a fault is detected. Conventional SAT approaches make this mathematical analysis based on diagnosis equations [3].

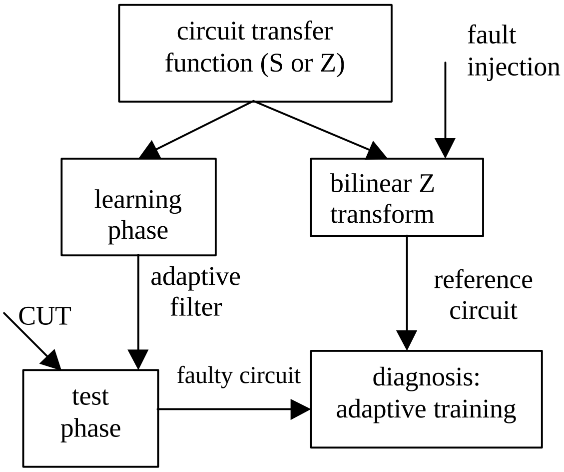


Figure 1: Complete test and diagnosis system

This work presents a new SAT method to diagnose faults in analog circuits checked according to the test approach in [1]. As denoted in figure 1, after a fault is detected, a third step is added to the adaptive tester that aims at identifying the faulty component (*adaptive training*). The basic diagnosis mechanism consists on injecting faults into a *reference circuit* (model) and on comparing its output against the output of the circuit under test. When a match is found, then the component which caused the failure is identified. The tester proposed in this paper implements this mechanism in the digital domain, where fault injection can be easily achieved in a fast and economical way.

Some works using similar learning techniques have already appeared in the literature. [8] shows that Robust Heteroscedastic Probabilistic Neural Networks can be efficient to detect faults in analog circuits. The convergence and testing time are both very small. [7] also uses neural networks to provide automatic test generation for robust diagnosis. A test and diagnosis scheme is proposed that includes a white noise generator at the circuit inputs and a neural network at the outputs. Both techniques offer a high fault coverage and high diagnosis resolution, but they also imply a big effort in simulation and training of the neural network. The reason is that learning is based on exercising the neural network with too many different deviations of each individual component in the circuit. Besides that, they may need measuring too many parameters to achieve high figures for fault detection and fault classification. In this work, an alternative diagnosis system is proposed that measures a single test parameter and requires much less time for training. However, as it will be discussed later, in the adaptive tester a 100% fault classification cannot be ensured yet.

3. Fundamentals of the diagnosis procedure

The basic approach of the diagnosis method is to compare the circuit under test against a mathematical model of the fault-free circuit, into which faults are injected. The mathematical model is a bilinear Z-transform, obtained automatically from the circuit transfer function [9]. The Z-transform brings the analog circuit to the digital domain, where fault injection can be easily accomplished. For example, changing the nominal value of a capacitor by ±10% can be achieved by simply modifying the coefficients of a digital filter. The idea is then to inject faults into the Z-transform domain and evaluate the difference to the behavior of the circuit under test (figure 2). When the fault injected into the mathematical model is the same fault that affects the real circuit, then the two circuits will present a very close behavior. Therefore, when the error output in figure 2 is minimal, the fault location is identified, since the algorithm is tracking the components into which faults are injected.

Faults are injected into the model one-by-one. As opposed to [14], the mathematical model is not modified to implement a new circuit topology due to a hard fault in a component. Only component deviations are injected into the mathematical model, changing every part of the model where the faulty component appears.

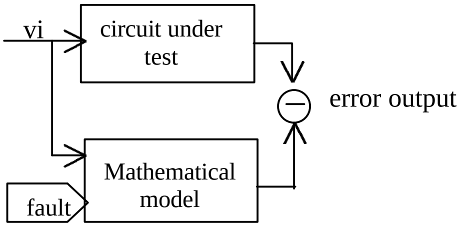


Figure 2: Theoretical diagnosis method

The initial fault can be arbitrarily defined by the test engineer. Usually, positive and negative relative component deviations are considered (for example, +5% and –5% of the nominal value). The output is evaluated for each fault. If the error for the positive is smaller than for the negative deviation, then the following faults to inject will be given by +50%, and next by -50% of the previous deviation considered (+2.5% and +7.5%, for the example of +5%). Similarly, if the error for the negative is smaller than for the positive deviation, then, considering the example of 5%, the faults to inject next will be –2.5% and –7.5%.

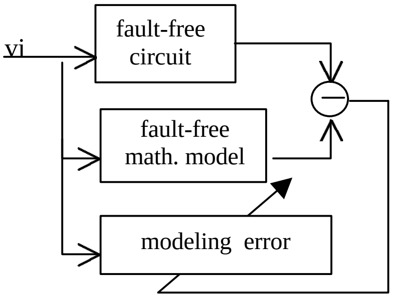
The MATLAB tool has been used to implement the sensitivity-guided dichotomy-based search process described above. This kind of procedure has been previously used for test generation in analog circuits [12] and microsystems [13].

4. Practical experiments

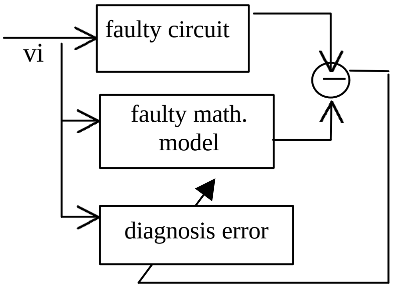
In the experiments made according to the diagnosis method presented in the previous sessions, the *circuit under test* of figure 2 was in fact replaced by some practical data that had been previously obtained from the adaptive testing of the real circuit.

It was noticed that there is an intrinsic difference between the real circuit and its mathematical model. This difference varies according to the fault, and it exists even for a fault-free circuit and the ideal model. One of the reasons for this disagreement is that the physical circuit uses non-ideal components (resistors with 5% tolerance, capacitors with 10% tolerance, for example). Moreover, the A/D converter used to bring information from the analog to the digital world also introduces some phase distortion. Because of this, it is not possible to directly compare these two models, as originally intended. Therefore, a new step is necessary before the diagnosis phase takes place. It consists on learning the difference between the real and the ideal circuits by using an adaptive filter [5,10] (figure 3(a)). This learning phase must also be performed when comparing the faulty circuit with the model into which a fault was injected, as it can be seen from figure 3(b).

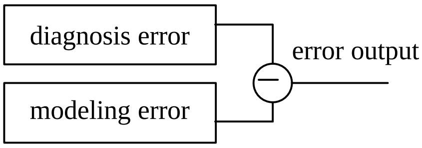
Notice that *modeling error* in figure 3(a) is just the difference between the models. However, *diagnosis error* in figure 3(b) embodies both the *modeling error* and the difference between the faulty circuit and the mathematical model into which the fault was injected. Then, when the model has the same fault of the faulty circuit, *diagnosis error* must be very close to *modeling error*. The new error output is thus the difference between the modeling error and the error associated to the fault injected into the model. When this value is minimal, then the fault is located (figure 3(c)). This approach was firstly used to diagnose faults in the analog filter shown in figure 4.



(a) learning step



(b) diagnosis phase



(c) error signal

Figure 3: Practical diagnosis scheme

Table 1 – type of faults and diagnosis success ratio

Component deviations	**diagnosis success ratio**
Soft (1%,5%,10%)	70.83%
Large (50%)	100%
Hard (short, open)	50%
Altogether	71.43%

Table 1 summarizes the results obtained from MATLAB simulations considering data extracted from a real circuit. This table shows the diagnosis success ratio, that is, the percentage of fault experiments for which the diagnosis procedure has correctly found the faulty component. 42 different faults, among soft, large and hard deviations, were considered: 1%, 5%, 10% and 50% resistor and capacitor deviations, and all those component shorts and opens that did not saturate the CUT. Catastrophic faults leading to saturation make diagnosis impossible using the procedure described in the previous session. Some faults affecting components R2, R3 and C2 have proven equivalent. In fact, this just confirms the conclusion drawn in [2] that these faults lead to similar faulty behaviors in a broad range of frequencies.

The results in table 1 can be analysed as follows: Similarly to previous approaches based on neural networks [7,8], a very high success ratio was obtained for large component deviations. Unlike [8], the success ratios for soft and hard fault diagnosis have drastically dropped. This is mainly due to the fact that we are measuring a single test parameter: the output voltage. In [8], four different test parameters are considered, thus a straight comparison would be unfair.

From the results above, it is clear that the main problem arises when hard faults come into play. These faults cause strong changes in behavior since circuits are moved out of the linear region of operation. Diagnosis experiments based on sensitivity analysis [2] had already shown in the past that the detection regions for those faults may be very similar for many components of the CUT. That is the reason why we are now focusing our research work on finding a model that is capable of dealing with this situation. For the time being, we are experiencing the replacement of the linear by a non-linear adaptive filter in the analog tester. Hopefully, effects like saturation will be correctly modeled in the adaptive tester, leading to an increasing success ratio for the diagnosis of hard faults.

Additionally, the biquad transfer function was modified to make it possible to change the gain of the operational amplifiers. This way, faults that affect this parameter could be injected into the circuit model. Three large gain deviations were injected into the second operational amplifier of the CUT. Gains of 50, 100 and 1000 were considered, instead of an infinity gain (ideal case). All of them were successfully detected and located by the test and diagnosis system. This is another important contribution of this work, since operational amplifiers, although hard to test, are in general considered as black-boxes in existing test systems.

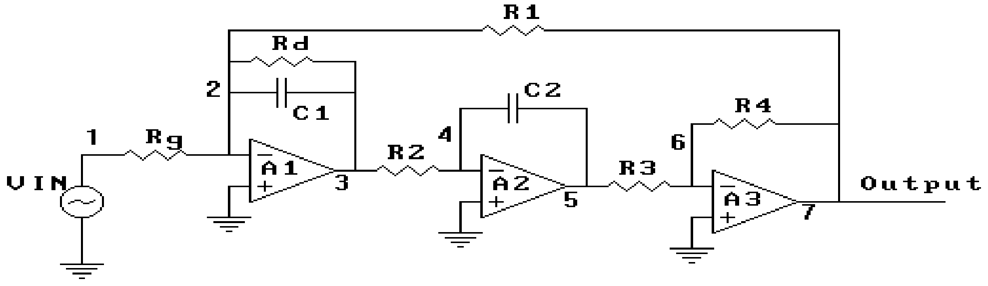


Figure 4: Biquad filter

5. Concluding remarks

This paper has presented an alternative diagnosis method for analog circuit testing. The method is based on adaptive testing and on fault injection into a mathematical model of the CUT. Its ability of classifying component faults is comparable to existing methods based on neural networks. Its main advantages are the extremely low time required for learning and diagnosing, and the location of faults even in operational amplifiers.

The first obtained results are very promising, although diagnosis of hard faults needs further investigation. Additionally, experimental results performed using a prototype for the tester have demonstrated that a low cost and time-efficient implementation can be achieved for the diagnosis procedure. This technique can also be applied to non-linear circuits by modifying the adaptive filter as shown in [11].

6. References

[1] L. Carro, M. Negreiros. "Efficient Analog Test Methodology Based on Adaptive Algorithms". *Proc. ACM/IEEE Design Automation Conference*, pp. 32-37, 1998.

[2] E.F. Cota, M. Lubaszewski, E. Domenico. "A Frequency-domain Tool for Test and Diagnosis on Analog Circuits". *Journal of the Brazilian Computer Society*, Special Issue on Fault-tolerant Computing, vol. 4, n.2, pp.5-15, November 1997.

[3] R. W. Liu (Editor). *Testing and Diagnosis of Analog Circuits and Systems*. Van Nostrand Reinhold, New York, 1991.

[4] S. Mir, M. Lubaszewski, B. Courtois. "Fault-Based ATPG for Linear Analog Circuits with Minimal Size Multifrequency Test Sets". *Journal of Electronic Testing: Theory and Applications*, vol. 9, pp. 43-57, 1996.

[5] J. G. Proakis, D. G. Manolakis. *Introduction to Digital Signal Processing.* Macmillan Publishing Company, New York, 1988.

[6] M. Slamani, B. Kaminska. "Multifrequency Analysis of Faults in Analog Circuits". *IEEE Design & Test of Computers*, pp.70-80, Summer 95.

[7] R. Spina, S. Upadhyaya. "Linear Circuit Fault Diagnosis Using Neuromorphic Analysers". *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 44, n. 3, pp.188-196, March 1997.

[8] Z.R. Yang, M.Z. Zwolinski, C.D. Chalk. "Fault Detection and Classification in Analog Integrated Circuits using Robust Heteroscedastic Probabilistic Neural Networks"**.** *Proc. 4th International Mixed-Signal Testing Workshop*, pp. 41-46, June 1998.

[9] K. Ogata, *Modern Control Engineering,* Prentice Hall, 1970.

[10] B. Widrow, S.D. Stearns. *Adaptive signal processing*. Englewood Cliffs, N.J.: Prentice-Hall, 1985.

[11] G. Jahn, L. Carro. "Sensor and amplifier non linear characteristics compensation." *Proc. Intl.Conference on Microelectronics and Packaging*, pp.275-282, August 1998.

[12] N. Naveena A. Chatterjee, A. Balivada, J. A. Abraham. "Fault-based Automatic Test Generator for Linear Analog Circuits", *Proc. IEEE International Conference on Computer-Aided Design*, pp.88-91, 1993.

[13] M. Lubaszewski, E. F. Cota, B. Courtois. "Microsystems testing: an Approach and Open Problems". *Proc. Design, Automation and Test in Europe (DATE'98***),** February 1998.

[14] N. Naveena A. Chatterjee, A. Balivada, J. A. Abraham. "DRAFTS: Discretized Analog Circuit Fault Simulator", *Proc. 30th ACM/IEEE design Automation Conference,* pp.509-514, 1993.