OPTIMIZATION OF PRODUCTION TESTS FOR ANALOG CIRCUITS UNDER PARAMETER VARIATIONS

A. Khouas and A. Derieux
LIP6, Université Pierre et Marie Curie, FRANCE

KEYWORDS: Analog, Test, Optimization, Fault Simulation

Abstract: Analog integrated circuit testing and diagnosis is a very challenging problem. The inaccuracy of measurements, the infinite domain of possible values and the parameter deviations are among the major difficulties. During the process of optimizing production tests, Monte Carlo simulation is often needed due to parameter variations, but because of its expensive computational cost, it becomes the bottleneck of such a process. This paper describes a new technique to reduce the number of simulations required during analog fault simulation. This leads to the optimization of production tests subjected to parameter variations. In section I a review of the state of the art is presented, section II introduces the algorithm and describes the methodology of our approach. The results on CMOS 2-stage opamp and conclusions are given in sections III and IV.

1. INTRODUCTION

Due to the continuous expansion of the market of telecomunications, multimedia, portable systems and systems on-chip which all need an analog interface, mixed signal and analog integrated circuit design becomes more and more important. During the production phase, test is an important factor contributing significantly to the manufacturing cost [1]. Tools for optimizing production tests aim to reduce the test time that a given circuit spends on the expensive ATE (Automatic Test Equipement).

Faults in analog circuits are classified into two categories : catastrophic faults and parametric faults [2]. Catastrophic faults include opens and shorts between nodes, they result from local defect mechanisms during processing. They usually change the topology of the circuit and produce global variations on the circuit behaviour. Parametric faults refer to small changes in the circuit parameters that do not affect its functionality, however they produce variations on the circuit outputs. The set of possible faults can be extracted from the circuit layout by using IFA (Inductive Fault Analysis) methods [3], [4] and [5], or can be constructed manually using fault models.

To build a tool for optimizing production tests, one has to be able to inject faults into the fault-free circuit, to run fault simulations and to compare the obtained results of all simulations. Test generation and optimization have been lately considered for linear analog circuits [6], [7] and [8], for mixed signal circuits [9], [10] and for static test generation [11] and [12]. To take into account parameter deviations we can use either Monte Carlo simulations as in [13] and [1] or interval analysis, see [14].

To have a high accuracy and an exact fault coverage in the optimization phase, we need to use Monte Carlo simulations for each faulty circuit in the fault simulation module. In this paper we propose a new technique to reduce the number of Monte Carlo simulations without degrading the accuracy of the results.

2. TEST OPTIMIZATION UNDER PARAMETER VARIATIONS

2.1 Formulation

For test optimization we need to introduce the following definitions :

- A set of tests : for test optimization we need to define a set $\mathbf{T} = \{T_1, T_2, ...T_n\}$ of n possible tests to be optimized. Each test T_i is defined by an input value and a measured output. The test T_i detects a fault if and only if the measured output is different for the fault-free and the faulty circuits.

- A set of faults : we also need to define a set $\mathbf{F} = \{F_1, F_2, ...F_m\}$ of m possible faults to be detected. The set of possible faults can be extracted from the circuit layout using IFA (Inductive Fault Analysis) methods.

- For each test $T_i \in \mathbf{T}$, Vg_i is the simulated value of the output of the fault-free circuit.

- For each test $T_i \in \mathbf{T}$ and each fault $F_j \in \mathbf{F}$, Vf_{ij} is the simulated value of the output of the faulty circuit having the fault F_j.

Before optimizing the set of production tests we have to run several fault simulations using a spice-like simulator in order to obtain the possible faults

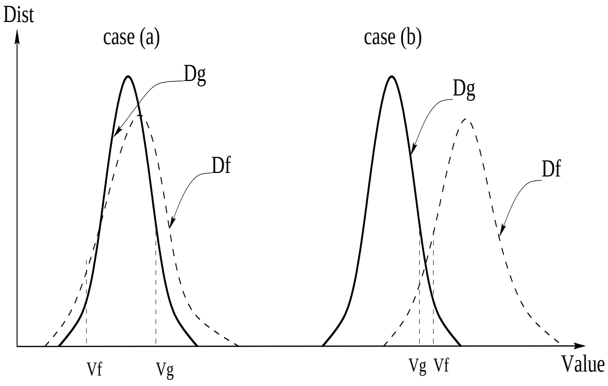

Figure 1: *Comparison of simulated values of good and faulty circuits.*

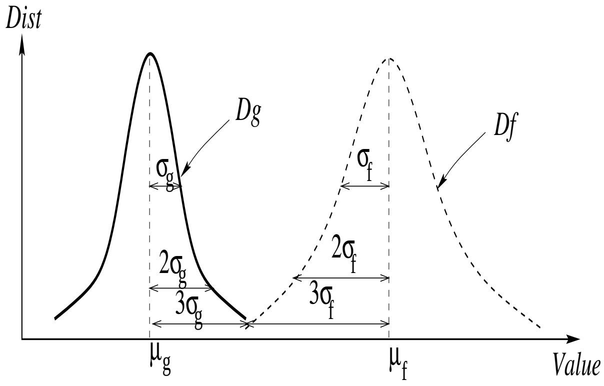

Figure 2: *Detectability of faults according to the distributions of a good and faulty values.*

which can be detected by each test. The fault simulator runs multiple simulations to determine the set $\{Vg_i/i \in [1..n]\}$ of simulation results of the fault-free circuit and the set of $\{Vf_{ij}/i \in [1..n], j \in [1..m]\}$ simulation results of all faulty circuits.
A direct use of the simulation results prove to be practically inadequate. Some of the difficulties include :

- The measurement is generally inaccurate both in the fault-free and the faulty circuits because of the continuous nature of analog signals.
- The inaccurate transistor models used in the simulator.
- Analog circuit parameters are usually considered with a certain tolerance range.

For example, in figure 1 V_g and V_f represent the simulated values of the output of the fault-free and the faulty circuits respectively, D_g and D_f are their corresponding probability density function. If we compare only the simulated values we can say that in case (a) the fault is detected because V_g is far from V_f, and in case (b) the fault is not detected. In fact if we now compare the probability density of the output values the conclusion is completely different, in case (a) the fault is not detected and in case (b) the fault is detected. So we need to compute the distribution for each faulty circuit if we want to get exact results.

Because of the high computational cost of Monte Carlo analysis, we propose a new technique to reduce the number of simulations without degrading the accuracy of the results.

2.2 Our Approach

The aim of our approach is to compute a large number of simulations only when it is really necessary (i.e. when the simulation results of the fault-free and the faulty circuits are close). For many faults, especially the catastrophic ones we don't need to make such a large number of Monte Carlo simulations to get an accurate result.
Let Dg and Df be the probability density of a given output of the fault-free and faulty circuits respectively (see figure 2). Each distribution is defined by its mean value μ and its standard deviation σ.
Depending on the required accuracy, we can define a threshold S such that :

$$(|\mu_g - \mu_f| > S) \Longrightarrow Fault\ is\ detectable \quad (1)$$

$$\text{with } S = k * (\sigma_g + \sigma_f)\ ,\ k \in [1, 3]$$

The factor k determines the effectiveness of the optimal test set. If k is low the probability that a good circuit will fail undergoing the test is high and the probability that a bad circuit will pass the test is low and vice-versa.

For the normal distribution, the probability that a continuous random variable x has a value within a certain interval is given by the area under the corresponding distribution curve between the two limits of the interval. For the interval $[\mu - k\sigma, \mu + k\sigma]$ with different values of k, the probability is shown in the following table :

k	probability
1	68.26 %
2	95.44 %
3	99,74 %

We can see that for k within one, two and three standard deviations of the mean, the probability is more than 68%, 95% and 99.5% respectively. In this paper we choose $k = 2$ and $S = 2(\sigma_g + \sigma_f)$, such that according to (1) we have :

$$(|\mu_g - \mu_f| > 2(\sigma_g + \sigma_f)) \Longrightarrow Fault\ is\ detectable \quad (2)$$

and if we suppose that $(\sigma_g = \sigma_f)$ the assumption (2) becomes :

$$(|\mu_g - \mu_f| > 4\sigma_g) \Longrightarrow Fault\ is\ detectable \quad (3)$$

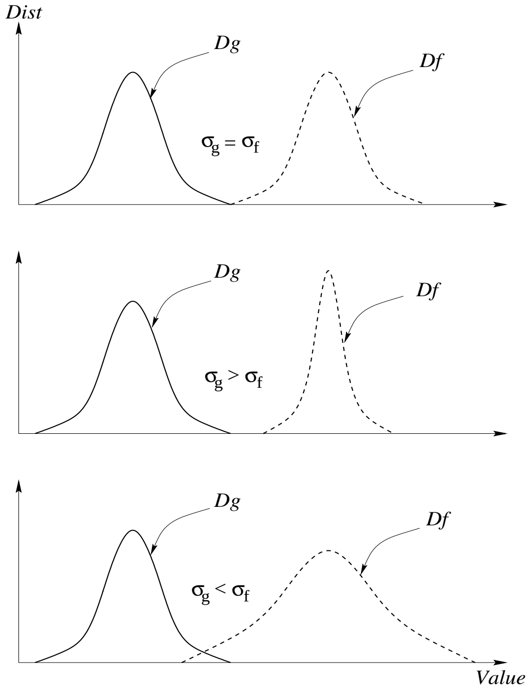

Figure 3: *Comparison of distributions in case of* $(\sigma_g = \sigma_f)$, $(\sigma_g > \sigma_f)$ *and* $(\sigma_g < \sigma_f)$.

We can see from figure 3 that if $(\sigma_g > \sigma_f)$ the assumption (3) is true, and if $(\sigma_g < \sigma_f)$ and $|\mu_g - \mu_f| > 4\sigma_g$ then we can prove that the area $(Dg \cap Df)$ is less than $0.5 * area\ of\ Dg$, so that the fault is detectable. In conclusion, we can consider that the assumption (3) is always true for every σ_f.

Let's consider that $V_f \in [\mu - 2\sigma_f, \mu + 2\sigma_f]$ is one of the simulated values of the faulty circuit, so according to assumption (2) we deduce that :

$$(|(\mu_g - V_f)| > 2\sigma_g + 4\sigma_f) \Longrightarrow Fault\ is\ detectable \quad (4)$$

With the same reasoning, we can deduce that for $(\sigma_g \geq \sigma_f)$ Assumption (4) becomes :

$$\boxed{(|\mu_g - V_f| > 6\sigma_g) \Longrightarrow Fault\ is\ detectable} \quad (5)$$

and for $(\sigma_g < \sigma_f)$ assumption (5) is still valid.

Assumption (5) shows that if the simulated value V_f of the faulty circuit satisfies the condition $(|\mu_g - V_f| > 6\sigma_g)$, we can deduce without computing the distribution of the faulty value and without using Monte Carlo simulation that the fault is detectable by the corresponding test.

2.3 Test Optimazation Algorithm

```
for each test T_i, i ∈ [1..n] do
    compute μ_g et σ_g for the fault-free circuit
    for each fault F_j, j ∈ [1..m] do
        Nstat = 0 ;
        StopSim = 0 ;
        DD_ij = 0 ;          // initialize fault not detected
        while ( (FinSim == 0) and (Nstat < MAXstat) ) do
            make random simulation ;
            compute the value Vf_ij of faulty circuit ;
            if (|μ_g − Vf_ij| > 6σ_g) then
                DD_ij = 1 ;          // the fault is detected
                StopSim = 1 ;
            else
                Nstat ++
                save Vf_ij          // needed to compute μ_f et σ_f
            end if
        end while
        if (DD_ij == 0) then          // if fault not detected
            compute μ_f et σ_f of faulty circuit ;
            I_g = [μ_g − 2σ_g , μ_g + 2σ_g] ;
            I_f = [μ_f − 2σ_f , μ_f + 2σ_f] ;
            DD_ij = (I_f − I_g) / ((I_f ∩ I_g) + (I_f − I_g)) ;
        end if
    end for
end for
```

This algorithm is based on assumption (5). For each test T_i, we compute the mean and the standard deviation μ_g *and* σ_g of the fault-free circuit, and for each fault F_j we make the first simulation to evaluate Vf_{ij}. If Vf_{ij} satisfies assumption (5), then the fault is detected $(DD_{ij} = 1)$ and we proceed to the next fault, else we save the value and we choose another simulation and repeat the same process. If after $MAXstat$ simulations $(MAXstat \in [10 \cdots 50])$, assumption (5) is not reached, then we compute the mean μ_f and the standard deviation σ_f of the distribution of the MAXstat simulated values of the faulty circuit. At the end, we compare the two distributions with the following feasibility function :

$$\boxed{DD(I_g, I_f) = \frac{(I_f - I_g)}{(I_f \cap I_g) + (I_f - I_g)}} \quad (6)$$

With $I_g = [\mu_g - 2\sigma_g\ ,\ \mu_g + 2\sigma_g]$ and $I_f = [\mu_f - 2\sigma_f\ ,\ \mu_f + 2\sigma_f]$

The function $DD(I_g, I_f)$ allows us to quantify by a number $\in [0\ ,\ 1]$ how much the distribution of the faulty circuit, represented by the interval I_f, is close to the distribution of the fault-free circuit, represented by the interval I_g. The coefficient DD_{ij} defines the detectability degree of the fault F_j by the test T_i, it can be considered as the probability that a test T_i detects the fault F_j. When we have $D_f \approx D_g$

(the symbol $\approx$ means close to), that means $I_f \approx I_g$, $(I_f - I_g) \approx 0$ and $DD_{ij} \approx 0$. In the opposite case, if $D_f \not\approx D_g$ then $(I_f - I_g) \approx 1$, $(I_f \cap I_g) \approx 0$ and $DD_{ij} \approx 1$.
So, when the coefficient DD_{ij} is close to 1, it means that the probability to detect the fault is high $((DD_{ij} = 1) \Longrightarrow$ the fault is detected), and when DD_{ij} is close to 0 the probability to detect the fault is low $((DD_{ij} = 0) \Longrightarrow$ the fault is not detected).

At the end of all simulations, we obtain a matrix of detectability degrees DD_{ij}, in which rows $T_1 \cdots T_n$ of the matrix represent the n tests and the columns $F_1 \cdots F_m$ represent the m faults, see tables 2 and 3.

2.4 Test Optimization

The aim of test optimization is to drop those tests that are never failed and to order the set of the chosen tests. Because production testing is terminated as soon as a test is failed, the average production test time varies depending on the ordering of the tests. More specificaly, if tests for gross failures are performed first and a given circuit frequently fails such tests, then the average production testing time will be shorter compared to another circuit which mostly fails undergoing system tests which are performed last.
The optimization method uses the matrix of detectability degrees DD_{ij} given by the fault simulation module. To drop the test that are never failed and to order the set of tests, for each test T_i we define the fault coverage FC_i given by:

$$\boxed{FC_i = \frac{\sum_{j=1}^{m} DD_{ij}}{m}} \tag{7}$$

If the fault coverage FC_i is high, the test T_i detects more faults. Thus, we compute the fault coverage for each test, then we order the set of tests and drop the tests that are never failed.

If we suppose that the optimal test set is :

$$\mathbf{T_o} = \{T_{o1}, T_{o2}, ...T_{ol}\}$$

of l tests, then the global fault coverage FC is given by :

$$\boxed{FC = \frac{\sum_{j=1}^{m} (Max\{DD_{kj}/k \in [1..l]\})}{m}} \tag{8}$$

the term $Max\{DD_{kj}/k \in [1..l]$ computes for each fault F_j the best detectability degree that can be obtained from all tests in the optimal test set.

3. EXPERIMENTAL RESULTS

We have applied our approach to a 2-stage CMOS opamp shown in figure 4, designed for a 0.8μ process. All the design parameters (W and L of the transistors and the capacitance values) are assigned

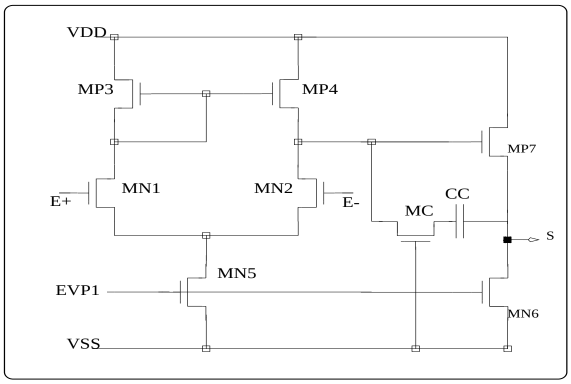

Figure 4: *2-stage CMOS opamp.*

fault no	description
1:MN2	parametric deviation, $W = W * 0.5$
2:MP3	parametric deviation, $L = L * 0.5$
3:MP4	parametric deviation, $W = W * 0.5$
4:MC	parametric deviation, $L = L * 0.5$
5:CC	parametric deviation, $CC = CC * 0.5$

Table 1: *Description of the faults used for the DD_{ij} matrix in tables 2 and 3*

5% of statistical variation. We considered 25 faults, 5 parametric faults, 10 open-circuit faults and 10 short-circuit faults. The AC analysis has been considered for frequencies between $0.5Hz$ and $5E6Hz$ with 8 frequency points, both magnitude and phase measurements are computed.
Because parametric faults are more difficult to detect than catastrophic faults, a sample of 5 different parametric faults described in table 1 has been used for illustration.

In tables 2 and 3, we have the computed matrix of the detectability degree considering both the magnitude and the phase of the gain at the output S for the 5 faulty circuits. To optimize the magnitude test, according to table 2 we can see that 4 of the 5 faults can be detected, that the highest probability to detect the fault number 4 is 0.45 and that the set of tests can be reduced to only the test number 4 (frequency $= 5E2Hz$) with global fault coverage $FC = FC_4 = 4.45/5 = 86\%$.

For the gain phase test, we can see in table 3 that we have a best fault coverage than for the magnitude test. All faults are detectable, and the optimum solution consists of test number 7 followed by test number 5 because $FC_7 > FC_5$. The global fault coverage is $FC = 100\%$.

For all 25 faults, the average of the speed-up with respect to Monte Carlo simulation with 30 samples is increased by a factor of around 5.6 (see table 4), without degrading the accuracy of the results. The speed-up of our technique depends on the faults used, for catastrophic faults the speed-up is more significant than for parametric faults, because of the large deviations on the outputs in case of the catstrophic

test no	freq (Hz)	fault no 1	2	3	4	5	FC %
1	0.5	1	1	1	.35	.35	74
2	5.0	1	1	1	.35	.35	74
3	5e1	1	1	1	.33	.51	76
4	5e2	1	1	1	.45	1	89
5	5e3	1	1	1	.43	1	88
6	5e4	1	1	1	.43	1	88
7	5e5	1	1	1	.43	.83	85
8	5e6	1	1	1	.42	.46	77

Table 2: *Matrix DD_{ij} measuring the Gain magnitude of the opamp*

test no	freq (Hz)	fault no 1	2	3	4	5	FC %
1	0.5	1	1	.33	.19	1	70
2	5.0	1	1	.13	.20	1	66
3	5e1	1	1	0	.17	1	63
4	5e2	1	1	0	.08	1	61
5	5e3	1	1	.43	.08	1	70
6	5e4	1	1	1	.81	.47	85
7	5e5	1	1	1	1	.50	90
8	5e6	1	.43	1	1	.60	80

Table 3: *Matrix DD_{ij} measuring the Gain phase of the opamp*

faults than that for parametric faults.

circuit	CPU time (s) Monte Carlo (30 samples)	Our approach	factor
fault-free	4.14	4.14	1
25 faulty	98.32	17.56	5.6

Table 4: *Comparative CPU time results.*

4. CONCLUSION

We have presented a new methodology for production test optimization for analog circuits under parameter variations. We have described our approach using the statistical theory, and demonstrated the new technique on a 2-stage CMOS opamp by optimizing the set of tests for the gain magnitude and phase test with fewer Monte Carlo simulations.
This method is currently studied to implement an automated tool for production test optimization and test diagnosis for analog circuits.

THE AUTHORS

Abdelhakim Khouas and Dr. Anne Derieux are with the ASIM Department of LIP6 Laboratory, Université Pierre et Marie Curie, Couloir 55-65, 2ème étage, 4 place Jussieu, 75252 Paris Cedex 05, France.
E-mail: Abdelhakim.Khouas@lip6.fr

References

[1] Linda Milor and A. L. Sangiovanni-Vincentelli. Minimizing Production Test Time to Detect Faults in Analog Circuits. *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, 13(6), 1994.

[2] W. Maly, A.J. Strojwas, and S.W. Director. VLSI Yield Prediction and Estimation: A unified Framework. *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, 5:114–130, 1986.

[3] D.M.H. Walker. *Yield Simulation for Iintegrated Circuits*. Kluwer Academic Publishers, 1987.

[4] C. Sebeke, J. P. Teixeira, and M.J Ohletz. Automatic Fault Extraction and Simulation of Layout Realistic Faults for Iintegrated Analogue Circuits. In *Proc. European Design and Test Conference*, pages 464–468, 1995.

[5] Manoj Sachdev and Bert Atzema. Industrial Relevance of Analog IFA : A Fact or Fiction. In *Proc. International Test Conference*, pages 61–70, 1995.

[6] C.-J. Richard Shi and Michael W. Tian. Automatic Test Generator of Linear Analog Circuits Under Parameter Variations. In *Proc. Asian and South Pacific Design Automation Conference*, 1998.

[7] Naveena Nagi, Abhijit Chatterjee, Ashok Balivada, and Jacob A. Abraham. Fault-Based Automatic Test Generator for Linear Analog Circuits. In *Proc. IEEE International Conference on Computer-Aided-Design*, pages 88–91, 1993.

[8] S. Mir, M. Lubaszewski, and B. Courtois. Fault-Based ATPG for Linear Analog Circuit with Minimal Size Multifrequency Test Sets. *Journal of Electronic Testing, Kluwer Academic Publishers*, 1996.

[9] H. Kerkhoff, R. Tangelder, H. Speek, and N. Engin. MISMATCH: A Basis for Semi-Automatic Functional Mixed-Signal Test-Pattern Generation. In *Proc. IEEE International Conference on Electronics, Circuits, and Systems*, pages 1072–1075, 1996.

[10] Bechir Ayari, Naim BenHamida, and Bozena Kaminska. Automatic test vector generation for mixed-signal circuits. In *Proc. European Design and Test Conference*, pages 458–463, March 1995.

[11] Giri Devarayanadurg and Mani Soma. Analytical Fault Modeling and Static Test Generation for Analog ICs. In *Proc. IEEE International Conference on Computer-Aided-Design*, pages 44–47, 1994.

[12] Wim Verhaegen, Geert Van der Plas, and Georges Gielen. Automated test pattern generation for analog integrated circuits. In *Proc. IEEE VLSI Test Symposium*, pages 296–301, 1997.

[13] Zwolinski M. Spinks S.J., Chalk C.D. and Bell I.M. Generation and Verification of Tests for Analogue Circuits Subject to Process Parameter Deviations. In *IEEE Int. Symp. Defect and Fault Tolerance in VLSI Systems, Paris, France*, October 1997.

[14] Michael W. Tian and C.-J. Richard Shi. Efficient DC Fault Simulation of Nonlinear Analog Circuits. In *Proc. Design And Test in Europe*, 1998.