Robust Optimization Based Backtrace Method for Analog Circuits *

Alfred V. Gomes and Abhijit Chatterjee
School of Electrical and Computer Engineering
Georgia Institute of Technology
Atlanta, GA 30332

Abstract

In this paper, we propose a new robust approach to signal backtrace for efficiently testing embedded analog modules in a large system. The proposed signal backtrace method is formulated as a solution to a multi-point boundary value problem(BVP), with constraints on the output state and the input. This error constraint minimizes large spurious deviations in the input signal and the convergence problems that arise if multiple solutions exist or if the desired signal does not exist in the feasible signal space. As an additional attractive advantage, this formulation preserves the core iteration structure of a SPICE-like simulator without modifications, greatly easing implementation.

1 Introduction

With the rapid evolution of mixed-signal IC technology to support a number of applications in telecommunications, consumer electronics and mobile computing, efficient techniques for testing these circuits has assumed a lot of importance. Although many AC, DC and transient test generation techniques[1, 2, 3, 4, 5] have been proposed, most of these are restricted to small circuits mainly due to high simulation complexity and lack of realistic fault models which can handle large number of possible faults. Since complex analog circuits are typically built using well understood analog macro cells like opamps, comparators, current mirrors, multipliers, references, data converters, etc., the problem of testing large complex circuits can be reduced to hierarchical testing of these macro cells, as test generation techniques for testing these cells is available. A major difficulty in applying these tests is the limited controllability and observability of the input and output nodes of the embedded cell. A brief motivation for test generation is presented in section 2. *An important problem we address in this paper is determining a stimulus on the primary input which will produce the desired signal at an internal node, after propagating through intermediate modules which can potentially transform the input signal*.

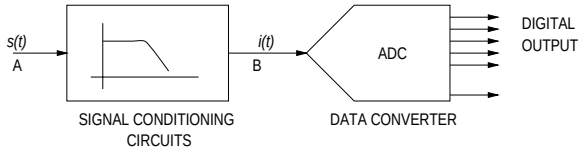

Figure 1: Test circuit to illustrate backtrace

For illustration purposes, consider the circuit shown in Figure 1, which shows a low pass filter in cascade with an A-to-D Converter(ADC). This configuration is typically found in many microcontroller and DSP ICs. The specifications for the combined block is a combination of specifications of the filter and the ADC. To test the specifications of the ADC (like INL, DNL, monotonicity, etc.) we have to apply a specified signal *i(t)* to the input of the ADC, at node *B*. As the signal at the node *B* is dependent on the signal at the primary input *A* and all circuit blocks which affect the functional mapping between *A* and *B*, we have to determine an input signal *s(t)* at *A*, which will produce the desired waveform *i(t)* at *B*.

Initial work in signal backtrace in the context of test generation has been presented by Ramadoss and Bushnell in [6] and [7]. In [6], the authors use signal flow graphs to model a network, and inverted signal flow graphs are used to backtrace signals. This method is mainly restricted to linear circuits. In [7], the authors use the MNA formulation and swap the variables of solution to determine an input signal which will produce the desired signal at the output. Both methods need the desired waveform to be in feasible signal space so that a solution exists. This paper is organised as follows: in the next section, a brief overview of analog test generation is provided. In section 3, we describe the proposed backtrace method. Section 4 presents the experimental results and section 5 concludes the paper.

2 Overview of analog test generation

Explicit verification of all specifications of a complex mixed signal device is expensive and time consuming. Research in developing alternate tests has resulted in development of many DC, AC and transient test methods. The DC tests[3, 4] are simple develop and apply, and are mainly used as screening tests to reject obviously defective circuits. AC tests[8, 9] are based on multi-tone periodic stimuli, and are typically used to test a class of circuits whose specifications are strongly related to frequency domain behavior of the circuit. Transient tests[10, 5, 11, 12] are superset of AC and DC tests, hence offering higher *fault and yield coverage*(see [13] for defns) and greater flexibility in applying complex test sequences. Comprehensive fault models are a key factor, which control the complexity of test generation and the resulting fault coverage. For analog circuits, the use of simple catastrophic fault models preclude their use to detect parametric faults, as the fault models is not linked to performance specifications. On the other hand, test generators using statistical fault models[13] can detect parametric faults, but require large number of simulations. Although alternate tests generation techniques are limited to small circuits due to computational complexity, reseachers are developing hierarchical frameworks to apply these tests to large modular circuits. This paper develops a method to backtrace sub-circuit tests to the primary inputs of the circuit.

3 Proposed backtrace method

Major problems involved with signal backtrace can be illustrated by analyzing a linear system. Consider a circuit whose transfer function is given by $H(\omega)$ in the frequency domain. If we desire a output signal *y(t)*, an input signal *x(t)* can be determined by deconvolving[14] the desired response with the system transfer function, given as $x(t) = IFT\{Y(\omega)/H(\omega)\}$, which essentially requires a frequency domain division and a inverse Fourier transform to obtain *x(t)*. In general, the poles and zeros of $H(\omega)$ are also poles

*This work has been supported by DARPA under the contract F33615-95-2-5562.

and zeros of $Y(\omega)$. At these poles and zeros, a straight forward division leads to indetermination, leading to noise like errors in the division operation. The main implications are, any output signal lying outside the realizable region will fail to converge, and any error in specifying the signal in the regions where input to output sensitivity is very low, will lead to large error in the input signal, which dominates the entire waveform. Hence, a direct solution requires a very carefully specified desired signal, which in most practical cases is not possible. For general non-linear circuit, a frequency domain deconvolution is not defined, and hence has to be solved in the time domain. Additionally, nonlinearities like saturation, deadbands and delays complicate straight forward implementation of backtrace.

3.1 Problem formulation

To illustrate the proposed approach, consider a general circuit, which is represented by:

$$\dot{\overline{x}}(t) = f(\overline{x}(t), u(t)),\ \overline{x}(0) = x_0 \tag{1}$$

where $\overline{x}(t)$ is the state vector and $u(t)$ is the input signal. In the signal backtrace problem, the trajectory of one of the components of the state vector $x_k(t)$ is specified as the desired waveform. If an input $u(t)$ is given, equation (1) can be solved numerically as an initial value problem(IVP). We use the trapezoidal method, a well accepted single-step integration formula to descretize and solve equation (1). Indexing the time points by $t = n$ and $t + \Delta t = n + 1$ we obtain:

$$\overline{x}_{n+1} = \overline{x}_n + (\Delta t/2)(f(\overline{x}_n, u_n) + f(\overline{x}_{n+1}, u_{n+1})) \tag{2}$$

Equation (2) is a non-linear equation with $\overline{x}_{n+1}$ as unknown. This formulation is identical to the one in analog circuit simulators, where equation (2) is efficiently solved by Newton-Raphson method due to the availability of analytical partial derivatives.

For the backtrace problem, since $u(t)$ is unknown, we can reformulate equation (1) as a multi-point boundary value problem(BVP), where the boundary conditions specified by the trajectory of $x_k(t)$ is satisfied by choosing a suitable $u(t)$. A common technique to solve this type of BVP is by *shooting method*[15]. This method uses a directed search for $u(t)$, whereby for each candidate input signal, equation (2) is solved iteratively, till boundary conditions are satisfied.

To illustrate this process, assume that we are given the desired signal $x_k(t)$, and we have determined an input $u(t)$, for the interval $0 \le t \le t_1$ such that equation (2) produces the desired output. To obtain a solution at the next time increment, satisfying the boundary condition:

$$x_k(t + \Delta t) = \beta \tag{3}$$

we can employ the shooting method over a single time step. In our case since at each time step, a constraint on the state is known, equation (3) can be incorporated into equation (2) and we can directly solve for $u(t + \Delta t)$, incurring no more additional cost than the usual forward simulation. Since equation (2) is a recurrence relation, the BVP can be incrementally solved, starting from a consistent initial state. Although, this solution strategy is very attractive from efficiency point of view, this method will succeed only if a solution to equations (2)and (3) exists at *every time point*. To illustrate this shortcoming, consider the simple circuit shown in figure 2. For a sinusoidal input, figure 3, the output waveform is shown(solid thin line). Given this output waveform as the desired waveform, the backtrace method described above will return, the sine wave as the backtraced output. If the previous output scaled up, it can be immediately seen that the desired signal lies outside the feasible signal space and the backtrace method as described will fail.

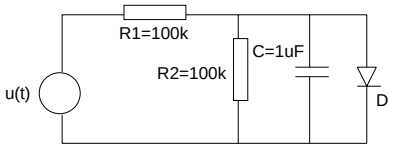

Figure 2: Simple non-linear circuit

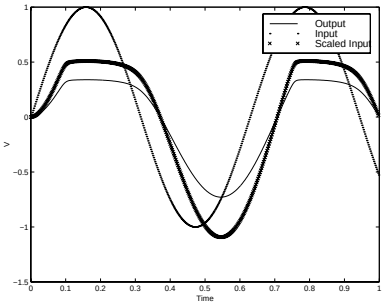

Figure 3: Response of the non-linear circuit

To increase the applicability of the backtrace method to signals which are crudely specified and may not completely feasible to generate, we reformulate the problem as a least squares approximation to the desired signal, by minimizing:

$$\min_{u(t+\Delta t)\in\Re} (x_k(t + \Delta t) - \beta)^2 \tag{4}$$

at each time point. Although this formulation is more robust, due to non-uniqueness of the solution $u(t)$ and the presence of input signals components which have no effect on the output, a large noise-like changes in the input can occur.

To control the *noise* in the input we use a modified error criterion given by:

$$E_0(u(t + \Delta t)) = (x_k(t + \Delta t) - \beta)^2 + \lambda(u(t + \Delta t) - u(t))^2 \tag{5}$$

$$\min_{\alpha\in\Re} E_0(\alpha) \tag{6}$$

so that change in the input is accounted during minimization of eqn (6). The extra term in equation (5) represents weighted (by λ)*input effort* and constrains the shooting method to choose the smallest deviation which will minimize output error as a function of $u(t + \Delta t)$. This formulation reduces to minimization of a nonlinear equations (2) and (5) as a function of $u(t + \Delta t)$.

3.2 Optimization method

As seen in the previous section, this formulation of signal backtrace reduces to minimization of a univariate function, $E_0(\alpha)$. We note that at the minima/maxima α^*, $\partial E_0/\partial\alpha = g(\alpha^*) = 0$. If the partial derivative of $g(\alpha)$ w.r.t α exist, an efficient method like Newton-Raphson can be used to determine the minima. In a general circuit simulation environment, these partial derivatives will not be available and have to be approximated by some numerical difference scheme. For reasons of efficiency we prefer forward differences:

$$d = \frac{\partial E_o}{\partial u} = \frac{E_o(u^i(t + \Delta t)) - E_o(u^{i-1}(t + \Delta t))}{u^i(t + \Delta t) - u^{i-1}(t + \Delta t)} \tag{7}$$

where $u^i(t + \Delta t), i = 2, 3, \ldots$ are solution iterates and $u^0(t + \Delta t)$ and $u^1(t + \Delta t)$ are initial guesses for start up. The next iterate, $u^{i+1}(t +$

$\Delta t)$ are chosen based on the line search formula:

$$u^{i+1}(t+\Delta t) = u^i(t+\Delta t) + \gamma \cdot d \tag{8}$$

where γ is the step length in the search direction d. To determine the step length γ, we use quadratic interpolation, where a function of the form

$$m(\gamma) = a\gamma^2 + b\gamma + c \tag{9}$$

is fitted with data from last 3 function evaluations, to determine the constants a,b and c. The extremum of this function occurs at $\gamma^* = (-b/2a)$ and is used in equation (8) to determine next iterate. Hence minimization proceeds as minimization of a sequence of quadratic subproblems.

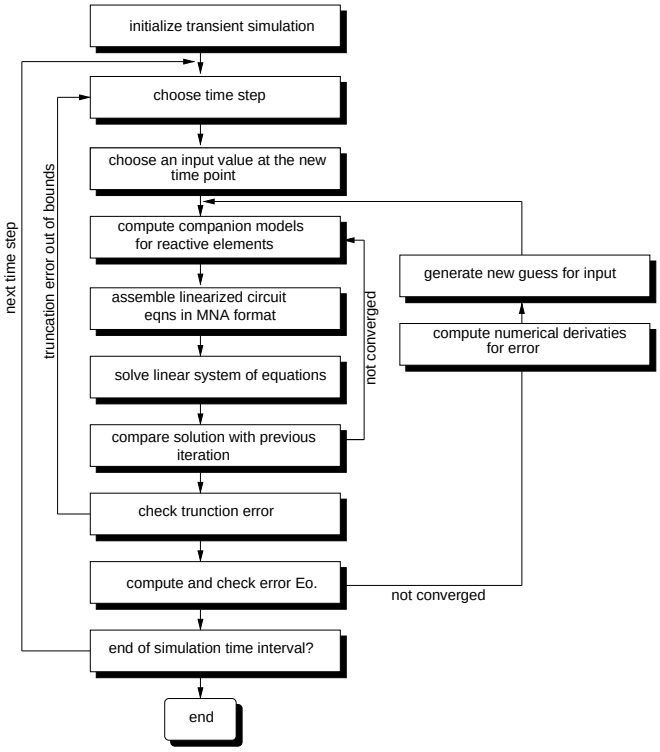

Figure 4: Flow diagram for SPICE transient simulation incorporating backtrace

3.3 Implementation details

The formulation as described in the previous section solves the backtrace problem as a solution to a multi-point BVP. The solution of a BVP using shooting methods in turn, uses the solution of the IVP. The innermost iteration loop, *i.e* the solution of the IVP, is identical to the one used in SPICE-like analog simulators. This has a very significant advantage as the backtrace method can use the SPICE core to solve the IVP, thereby precluding the need to develop special device models or new methods for solving a modified set of non-linear equations as needed in [7]. Figure 4 shows the flow diagram of the backtrace algorithm and has a few additional blocks added to the typical SPICE transient iteration structure.

4 Experimental results

To demonstrate the backtrace technique on a realistic circuit, we use a 5th order Chebyshev low pass filter, whose frequency response

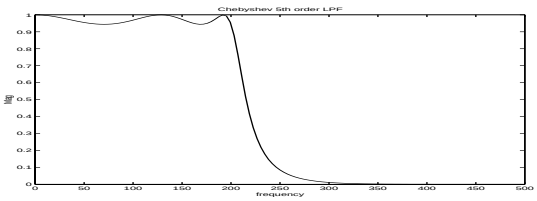

Figure 5: Frequency response of a 5th order LPF.

is shown in Figure 5. This circuit is difficult to backtrace, as it has significant lag, which for non-smooth output signals can easily give rise to large spurious deviations in the input. Additionally, a *smoothing* filter for a forward signal, will act like an high order differentiator when backtracing an output signal to the input. Figure 6 shows the result of an experiment where a step input is applied, and the resulting output(bottom part-solid line) is used a input for the back-trace program. The output of the backtrace program is shown in the top-part of figure 6. The forward simulation of the backtraced signal(bottom part-line marked by x) shows that it matches the desired response. The variable *k* in the figure represents λ. It is noted that in this case, the desired signal is clearly in the feasible signal space of the output.

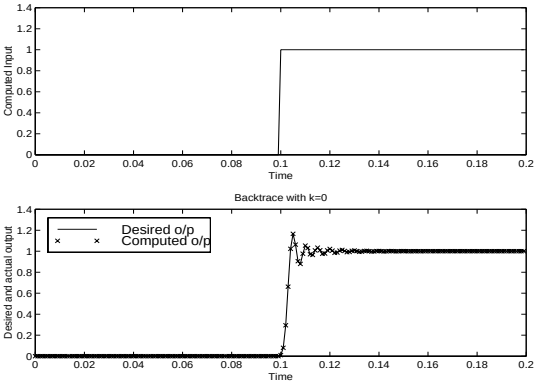

Figure 6: Backtrace of a feasible signal through the LPF.

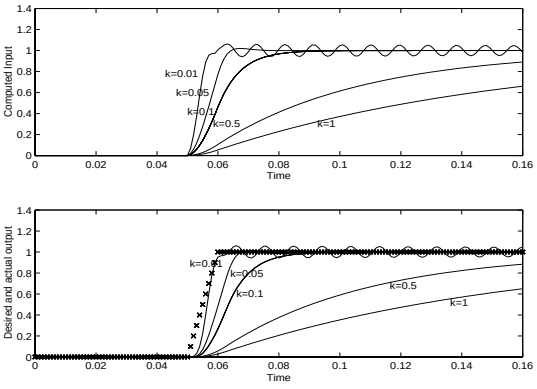

Figure 7: Backtrace of a ramp signal through the LPF for diffrent λ.

In figure 7, a slew rate limited signal(lower figure, marked by x), with sharp corners is specified as the desired signal. Since this signal not differentiable over the entire time interval, clearly it cannot be a solution to equation (1), hence it does not exist in the feasible

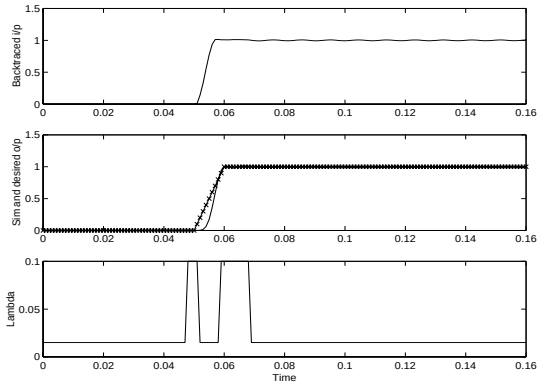

Figure 8: Backtrace of a ramp signal with weighted λ to control convergence and accuracy.

signal space. This figure shows the variation of the backtraced signal for different weighing factors k(or λ) illustrating the trade-off between accuracy of the approximation to the output signal and stability of the backtrace procedure. As $k(\lambda)$ decreases, the output of the backtraced signal closely matches the desired signal, but in the limit may become unstable. Figure 8, demonstrates a better approximation, by using variable λ so that larger value of λ can be used at time intervals which can cause convergence problems and reducing the value of $k(\lambda)$ over other parts of the signal.

5 Conclusions

Test generation for large analog circuits need novel methods to handle simulation and modeling complexity. We present a robust backtrace technique which can be used to backtrace sub-module tests to system level primary inputs, even in the presence of global feedback loops. This technique uses a novel formulation, which permits specifying output signals, which need not lie in the feasible signal space. Additionally, to reduce the problem of spurious noise in the input signal, a penalty factor, dependent on the change in input is incorporated in the cost function.

References

[1] L. Milor and A. S. Vincentelli, "Optimal test set design for analog circuits," *Int'l Conf. on Computer Aided Design*, pp. 294–297, 1991.

[2] J. B. Brockman and S. W. Director, "Predictive subset testing: optimizing ic parametric performance for quality, cost and yield," *IEEE Transactions on Semiconductor Manufacturing*, pp. 104–113, 1989.

[3] M. J. Marlett and J. A. Abraham, "DC-IATP: An iterative analog circuit test generation program for generating dc single pattern tests," *International Test Conference*, pp. 839–845, 1988.

[4] L. Milor and V. Viswanathan, "Detection of catastrophic faults in analog integrated circuits," *IEEE Transactions on Computer Aided Design*, vol. 8, pp. 114–130, 1989.

[5] G. Devarayanadurg and M. Soma, "Dynamic test signal design for analog ics," *International Conference on Computer Aided Design*, pp. 627–630, 1995.

[6] R. Ramadoss and M. Bushnell, "Test generation for mixed-signal devices using signal flow graphs," *Proc. of the IEEE 9th International Conference on VLSI Design*, pp. 242–248, 1996.

[7] R. Ramadoss and M. Bushnell, "Test generation for analog circuits using partitioning and inverted system simulation," *4th IEEE International Mixed-Signal Testing Workshop*, pp. 68–73, 1998.

[8] M. Salamani and B. Kaminska, "Multifrequency analysis of faults in analog circuits," *IEEE Design and Test of Computers*, pp. 70–80, 1995.

[9] N. B. Hamida, K. Saab, D. Marche, B. Kaminska, and G. Qusnel, "Limsoft: automated tool for design and test integration of analog circuits," *International Test Conference*, pp. 571–580,, 1996.

[10] S. J. Tsai, "Test vector generation for linear analog devices," *International Test Conference*, pp. 592–597, 1991.

[11] A. Balivada, J. Chen, and J. A. Abraham, "Analog testing with time response parameters," *IEEE Design and Test of Computers*, vol. 13, pp. 18–25, 1996.

[12] A. V. Gomes and A. Chatterjee, "Minimal length diagnostic tests for analog circuits using test history," *Design, Automation and Test in Europe conference*, pp. 189–194, 1999.

[13] W. M. Lindermeir, H. E. Graeb, and K. J. Antreich, "Design based analog testing by characteristic observation inference," *International Conference on Computer Aided Design*, pp. 620–626, 1995.

[14] A. Bennia and S. M. Riad, "An optimization technique for iterative frequency domain deconvolution," *IEEE transactions on instrumentation and measurement*, vol. 39, pp. 358–362, 1990.

[15] G. Golub and J. Ortega, *Scientific computing and diffrential equations*. Academic Press Inc, 1992.