Accuracy Management for Mixed-Mode Digital VLSI Simulation

Gary L. Dare
Motorola Labs *
Motorola, Inc.
Schaumburg, Illinois, USA 60196
Gary.Dare@motorola.com

Charles A. Zukowski
Department of Electrical Engineering †
Columbia University
New York, New York 10027-6699
caz@columbia.edu

ABSTRACT

Accuracy Management (AM) offers delay-oriented control of simulation accuracy in automated mixed-mode simulation of digital VLSI circuits. AM heuristics, with little additional computation, can achieve a required level of accuracy in output timing predictions while insuring computational efficiency by using simulations with appropriate levels of accuracy throughout a circuit. These results are demonstrated through both mathematical predictions on some benchmark circuit topologies, and actual mixed-mode timing simulations on large combinational circuit benchmarks. Simulation speedups of 5-10 over corresponding standard, waveform relaxation-based circuit simulations are demonstrated.

1. INTRODUCTION

Mixed-mode simulation [1] [2] [3] [4] is an attempt to combine simulators of different accuracy levels to obtain the verification ability of electrical circuit simulators [5] without incurring their extreme computational costs for VLSI circuits. Approximate methods can be used to analyze portions of a circuit containing non-critical digital logic much more quickly. The process of assigning accuracy levels to logic blocks in mixed-mode simulators, however, has not been well studied. Most programs simply make rough distinctions between critical and non-critical paths, and provide little guarantee of accuracy at circuit outputs. Our accuracy management (AM) approach uses delay-oriented heuristic algorithms in a mixed-mode framework to address these problems.

To be useful for large circuits, as well as accurate, a mixed-mode simulator must be able to automate its selection of models/algorithms across the circuit. If critical paths are searched for, either directly [6] or using sensitivity analysis [7], the amount of timing "slack" on the non-critical side paths is not accounted for. In a digital circuit, sensitivities fall quickly to zero on non-critical paths, so paths with different amounts of slack cannot be distinguished. As a result, the full efficiency that is possible with a mixed-mode approach cannot be achieved, and the accuracy level expected at the output cannot be predicted well.

The accuracy management (AM) approach, first proposed for bounding simulation over entire waveforms in [8], uses full information about timing slacks throughout the circuit to find efficient model/algorithm assignments, subject to output accuracy requirements. Assuming that the simulation is done within a waveform relaxation framework (for iterative refinement), which has been successfully applied to many types of digital circuits [9] [10] [4] [11], delay estimates from one iteration are available to guide assignments in later ones. In the past we have explored a number of AM heuristics for critical paths based on a rough model for the cost of computation as a function of block size and accuracy [12]. In this paper we describe a full set of linear-time AM heuristics for entire combinational logic circuits, and show through experimental results that they can significantly improve efficiency and confidence, without adding significant computation costs themselves.

2. BACKGROUND AND THEORY

The type of circuit considered in this study is purely combinational logic without feedback paths, such as in the simple example of Figure 1. The component blocks, each labelled with a number i, could implement simple logic or more complex functions. For the accuracy management algorithm, each block is characterized by its estimated delay, d_i, and a rough measure of its complexity, b_i. Digital 0/1 signal transition times and delays (triggering input to output) are defined in terms of $V_{DD}/2$ crossings. In this paper, the complexity b_i measures transistor count to reflect the relative computational effort required for timing simulation (CPU time, memory, storage, etc.) versus size (e.g., multiplexer versus inverter). It is assumed that each signal transition time and block delay has associated with it an estimated (or guaranteed, in the bounding simulation case) level of accuracy. The variable x_i represents the maximum allowable fractional error (or, relative error) assigned to block i by the AM algorithm, while e_i represents the absolute error, i.e. $e_i = d_i x_i$.

Assume that the delays listed inside each module in Fig-

*DigitalDNA Systems Architecture Research Laboratory.

†This work was supported by NSF Grant 88-11111 to the Center for Telecommunications Research at Columbia University, NSF PYI award MIPS-86-58112, and Engineering Foundation grant RI-A-86-10.

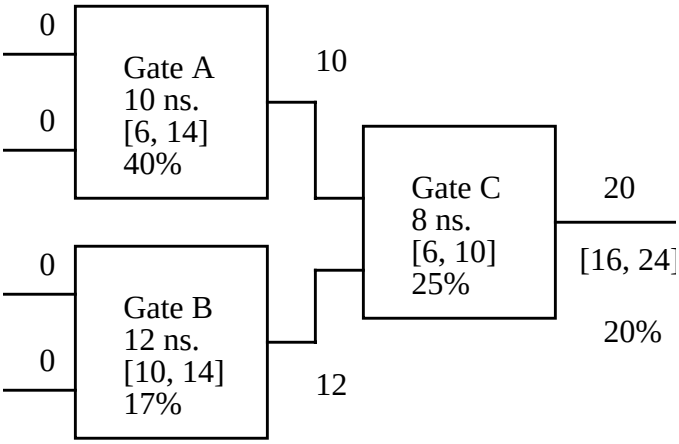

Figure 1: A simple combinational logic circuit.

ure 1 represent very rough estimates generated by an initial analysis using a simple timing simulation method. With a nominal propagation time of 20 ns, a circuit designer needing to verify that this circuit settles before t = 25 ns requires 20% accuracy at the output. One can achieve this by analyzing the entire circuit at 20% accuracy; however, there can be greater efficiency if we assume that accuracy for each block can be continuously traded for computation speed. Blocks B and C form a *critical path* that determines overall delay, so they must be assigned accuracy levels that result in 20% overall, but the optimal assignment may not be 20% for each. If C were assigned a 25% tolerance (yielding a delay interval of [6,10]) then B would have to be assigned a 17% tolerance ([10,14]) to achieve 20% at the output, and this might be more efficient depending on the analysis costs of each block. Such a choice would then constrain the tolerance of block A to be only 40% ([6,14]) due to its timing slack. The goal of our AM algorithm is to find the best (or close to best) assignments quickly.

The assignment heuristics used by the AM algorithms reported here are based on a simple model of simulation computation costs:

$$f(\vec{x}) = \alpha \sum_{i=1}^{N} \frac{b_i}{x_i} \tag{1}$$

for N total subcircuits. While there is no reason to believe that such linear scaling holds exactly, it can cover a wide range of models and simulators in a manner suitable for a rough heuristic. As shown in [12], this cost function leads to an optimal assignment for block i along a simple chain of

$$x_i = \frac{e\sqrt{\frac{b_i}{d_i}}}{\sum_{j=1}^{N}\sqrt{b_j d_j}} \tag{2}$$

to achieve a specified output delay tolerance e. Also derived in [12] are simplifications of (2) to obtain compact, single-calculation expressions. By modelling the circuit path as a set of uniform subcircuits of average complexity and delay, we get the **Uniform Dominant Path** model. Further, by considering the end block separately and applying the Uniform Dominant Path approximation to the preceding subpath, we get the **Partitioned Dominant Path** model. Expression (2) and its variations were all used in our simulation experiments, and since they all produced very similiar results only those for the basic model (equation (2)) will be shown in the following sections.

Our full AM algorithm starts with accuracy specifications at the outputs and works backwards recursively through the circuit, level by level, making accuracy assignments to all the blocks. Assignment is made using results calculated by equation (2) or its variants, applied for each block along the critical path that leads up to it. Focussing only on the critical path allows the use of those results for simple chains, keeping the algorithm execution linear time, and is reasonable because critical paths generally dominate the simulation effort. Fortunately our expressions require only aggregate characteristics of the critical path that can be trivially computed along with delays during the course of subcircuit simulations. Once the accuracy level is assigned to a block, a constraint is also placed on the required accuracy for its input signals, thus preparing the algorithm to analyze the next level in its recursion. Of course, the non-critical inputs to a block can be assigned lower accuracy at this step due to their extra timing slack. Total additional uncertainty on each non-critical path is limited to that slack, so that bounds on their signals will not impact the total accuracy established on the critical path. Finally, if inputs are assigned more than one level of accuracy due to multiple fanouts, then the tightest one is used to guarantee that the output accuracy constraints are met.

The use of a Waveform Relaxation-based framework for the simulation allows AM to be used in a natural way. During each iteration of simulation, statistics such as delay estimates and characterizations of critical paths can be collected with little overhead using block-oriented methods which propagate information only along the greatest delay paths [6]. These statistics can then be used by an invocation of the AM algorithm to refine the accuracy assignments for use in the next simulation iteration. In other words, AM and subcircuit simulations alternate, starting with a very rough (and thus low cost) simulation throughout the circuit. As long as the accuracy assignments are only allowed to get tighter over time, the addition of AM will not prevent the convergence of the simulation [13]. This monotonicity constraint removes the possibility of oscillation in assignments; while this potentially forces some blocks to slightly overshoot their optimal accuracies, it safeguards the goal of assuring the desired accuracy.

3. MODELLING OF ACCURACY MANAGEMENT

Our initial testbed demonstrated the Accuracy Management process at the theoretical level and relies on assigned delays for different gate structures to model the WR-based timing simulation process. It assumes that a corresponding virtual simulator is available at any required accuracy to "generate" the nominal circuit delay and appropriate tolerance. The

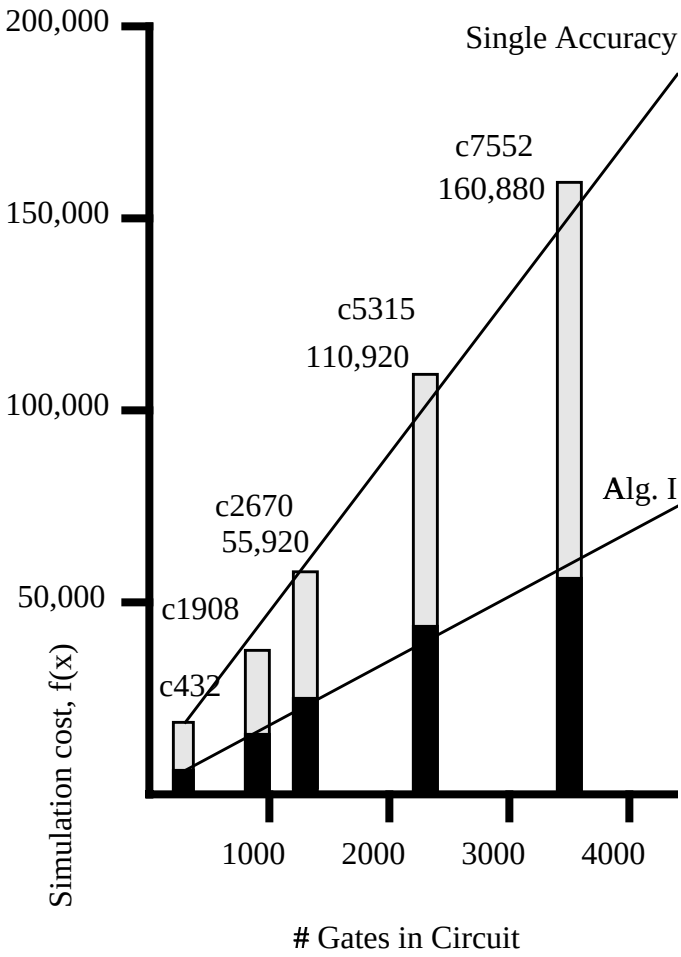

Figure 2: Predicted simulation costs for mixed-mode analysis w/AM, versus single highest accuracy.

overall algorithm uses event-driven selective trace scheduling (such as found in RSIM [16]) based on voltage waveform transition *events*.

The effect of Accuracy Management on simulation is reflected in evaluation of cost function $f(\vec{x})$ after the accuracy requirements of subcircuits are readjusted for the next relaxation using the latest data. This predicted computational cost is compared against simulation without Accuracy Management, at single accuracy, as a control.

Five benchmark circuits of varying sizes were selected from ISCAS '85 for our test [15]. An arbitrary value of 5% was chosen as the output specification while lowest available accuracy was set at 30%. Each set of accuracies for mixed-mode simulation of a circuit results from use of our basic AM heuristic. The computational costs calculated in Figure 2 show very pronounced savings when using Accuracy Management in comparison to simulation at a single, highest accuracy. These savings arise from greater uncertainty being allowed off of the critical paths, leading to gains in analysis efficiency from using less rigorous methods upon noncritical paths for consequent execution time improvements. This becomes more pronounced with increasing circuit size, indicating that the relative proportion of noncritical paths in these benchmarks also increases. We observed an asymptotic linear increase in computational cost versus the number of gates for both cases, which follows the linear increase in computation with number of subcircuits generally seen with waveform relaxation. However, note that the linear asymptote for mixed-mode with accuracy management grows at a much slower rate with circuit size.

The predicted costs for simulation from using any of the variations of our heuristic algorithm were virtually identical. This similarity in allocations arose partly from relative homogeneity of the "standard cell" approach that we used to generate circuits from benchmark logic, whereas more variety in subcircuits (like larger function blocks) could lead to greater variation. Otherwise, the relatively tight specification of 5% did not allow for great variation along the critical paths.

In a realistic mixed-mode simulation environment, we would use a limited number of methods. Accuracy requirements within a bounded range must be conservatively mapped to analysis at the next-highest accuracy to ensure that the specification is met. Subcircuits that would have been simulated at higher accuracy in one case might be analyzed at lower accuracy in another with finer resolution. We performed another set of trials like those in Fig. 2 but with discrete sets of available accuracies, which indicated that gains in computational savings arise at coarse resolution with small sets such as 3 accuracies; but, these savings increase at a diminishing rate as the resolution becomes much finer [13].

4. SIMULATION BENCHMARKS

In our second-generation testbed, actual timing simulation of subcircuits is carried out through a software interface to the appropriate circuit analysis methods: electrical (GSOLVER [11]), piece-wise approximate (SPECS [14]) and switch-level RC delay. Event-driven selective trace scheduling queues a subcircuit for simulation once all its input waveforms are available. These tests were conducted by remote execution of timing simulation jobs on an otherwise idle Sun workstation. The differences in computation can be seen directly from the execution times of mixed-mode simulations performed with accuracy management versus analyses at highest specified accuracy. However, a major problem encountered with the large benchmark circuits was exhaustion of local memory in our workstations using SPICE. Thus, we relied solely on waveform relaxation circuit simulation with GSOLVER as our baseline.

As was found with MOS VLSI combinational circuits without feedback loops simulated by Relax, our trials converged in less than 10 relaxations [9]. In our case, convergence of simulation accuracy from the AM process occurs prior to waveform convergence. CPU execution times for those tests are summarized in Figure 3, along with rough asymptotes for simulation performance under three test conditions (single accuracy WR, and mixed-mode WR using both single and multiple convergence criteria), confirming the modelled performance predictions of Figure 2.

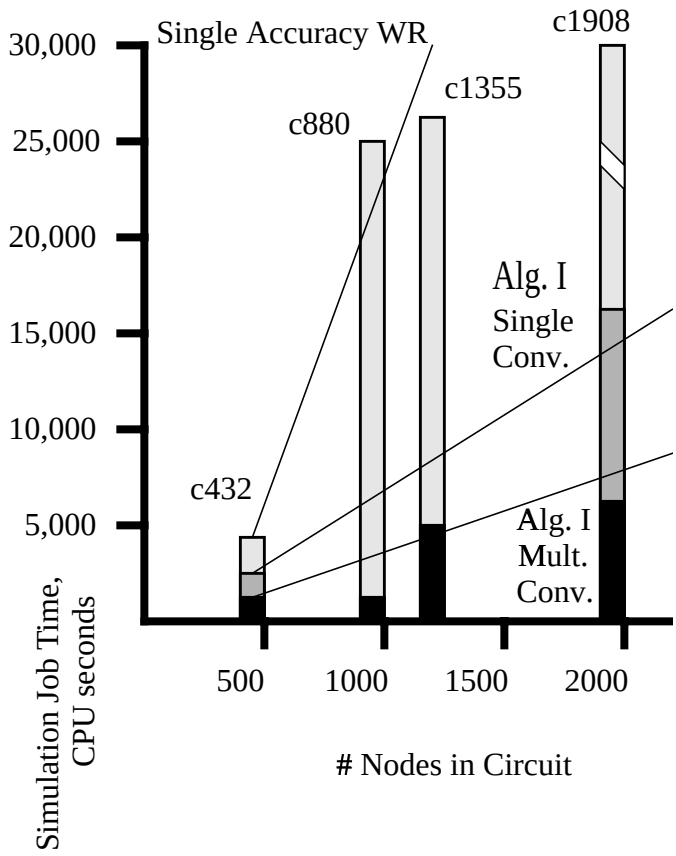

Figure 3: CPU job times for mixed-mode simulation w/AM, versus single highest accuracy.

5. CONCLUSIONS

We have been able to demonstrate that more thorough automation of management for mixed-mode timing simulation can be achieved. For large numbers of subcircuits, sufficient simulation accuracy can be achieved in the timing results while computational efficiencies arise from using less rigorous analysis on noncritical paths. We present a critical path-oriented technique which offers the increased appeal of simulating larger systems without isolating the critical paths for separate analysis. The ability to have all parts of the circuit present and account for their behaviour increases the overall quality of the simulation performed; this is desirable for digital systems and necessary for mixed-signal systems.

6. REFERENCES

[1] P. Reynaert, H. DeMan, G. Arnout and J. Cornelissen, "DIANA: A Mixed-Mode Simulator with a Hardware Description Language for Heirarchical Design of VLSI", *Proc. 1980 ICCC*, pp. 356-360.

[2] A. R. Newton, "Techniques for the Simulation of LSI Circuits", *IEEE Trans. Circuits and Systems*, vol. CAS-26, no. 9, September 1979, pp. 741-749.

[3] R. Chadha, C. Visweswariah and C. F. Chen, "M^3 — A Multilevel Mixed-Moded Mixed A/D Simulator", *IEEE Trans. CAD*, vol. 11, no. 5, May 1992, pp. 575-585.

[4] D. Dumlugöl, P. Odent, J. Cockx and H. DeMan, "The Segmented Waveform Relaxation method for mixed-mode switch electrical simulation of digital MOS VLSI circuits and its hardware acceleration on parallel computers", *Proc. 1986 ICCAD*, pp. 84-87.

[5] L. W. Nagel, "SPICE2: A Computer Program to Simulate Semiconductor Circuits", Technical Report ERL-M520, U. California, Berkeley, May 1975.

[6] R. B. Hitchcock, Sr., "Timing Verification and the Timing Analysis Program", *19th Design Automation Conf.*, 1982, pp. 594-604.

[7] P. Feldmann, T. V. Nguyen, S. W. Director and R. A. Rohrer, "Sensitivity Computation in Piecewise Approximate Circuit Simulation", *IEEE Trans. CAD*, vol. 6, no. 2, Feb. 1991, pp. 171-183.

[8] C. A. Zukowski, *The Bounding Approach to VLSI Circuit Simulation*, Kluwer, 1986.

[9] J. White and A. Sangiovanni-Vincentelli, "RELAX2: A Modified Waveform Relaxation Approach to the Simulation of MOS VLSI Circuits", *ISCAS 1983*, pp. 756-759.

[10] J. F. Beetem, P. Debefve, W. Donath, H. Y. Hsieh, F. Odeh, A. E. Ruehli, J. White and P. K. Wolff Sr., "A Large Scale MOSFET Circuit Analyzer based on Waveform Relaxation", *Proc. 1984 ICCD*, pp. 507-514.

[11] G. D. Gristede, *Methods for analyzing relaxation-based circuit simulation algorithms*, Ph.D. thesis, Dept. of Electrical Engineering, Columbia, 1992.

[12] C. A. Zukowski and G. L. Dare, "Accuracy Management for VLSI Timing Analysis", *ISCAS 1989*, pp. 2027-2031.

[13] G. L. Dare, "Accuracy Management for Delay-Oriented Control in Mixed-Mode Simulation of Digital VLSI Circuits", Ph.D. thesis, Dept. of Electrical Engineering, Columbia, 1993.

[14] C. Visweswariah and R. A. Rohrer, "Piecewise Approximate Circuit Simulation", *IEEE Trans. CAD*, vol. 10, no. 7, July 1991, pp. 861-870.

[15] F. Brglez P. Pownall and R. Hum, "Accelerated ATPG and Fault Grading via Testability Analysis", *Proc. 1985 ISCAS*, pp. 695-698.

[16] C. J. Terman, "RSIM - A Logic-Level Timing Simulator", *Proc. 1983 ICCD*, pp. 437-440.