A Methodology for Full-Chip Extraction of Interconnect Capacitance using Monte-Carlo-Based Field Solvers*

R.B. Iverson
Random Logic Corporation
2934 Beau Lane
Fairfax, VA 22031

Y.L. Le Coz
Center for Integrated Electronics
Rensselaer Polytechnic Institute
Troy, NY 12180

Abstract—We present a full-chip extraction methodology for evaluating self-capacitance of interconnects in complex digital ICs. We propose that a Monte-Carlo-based field solver be used to evaluate critical net capacitances and to accurately characterize a faster, less accurate empirical extractor. The fast extractor can then be used to find noncritical net capacitances. To facilitate *a priori* partitioning of nets into critical and noncritical categories, we have developed a procedure for estimating absolute computational error of any capacitance extractor. We also report that Monte Carlo extractors can efficiently evaluate coupling capacitance between IC nets. In this case, statistical error cancellation occurs during a subsequent circuit simulation.

I. INTRODUCTION

Modern digital ICs can contain millions of interconnects. To a large extent, parasitic capacitance associated with these electrical nets influences propagation delay and capacitive cross talk. Accurate, physically based 3D solvers are generally too slow for full-chip capacitance extraction; massively parallel implementations for such solvers may be necessary. On the other hand, fast extraction methodologies that employ empirical layout-parameter-fitting or library-lookup strategies, unfortunately, have limited accuracy, in part due to their small effective calculation window.

We propose here a methodology for full-chip capacitance extraction that uses an accurate, physically based Monte Carlo capacitance extractor in conjunction with a fast, lower-accuracy empirical extractor (layout parameter, library, or pattern matching). Monte Carlo extractors consume relatively small amounts of memory, handle complex geometries efficiently, allow huge calculation windows, report absolute errors in capacitance, and are computationally robust.[1–3] These properties make Monte Carlo extractors attractive for accurate capacitance extraction in a full-chip environment. We define our methodology as follows: (1) Use the accurate extractor to characterize the error of the fast extractor over actual nets in the full-chip environment. (2) Use the fast extractor for noncritical nets. (3) Use the accurate extractor for critical nets.

In the following sections we describe our methodology for full-chip extraction based on Monte Carlo solution. We also cover the basic theory underlying one Monte Carlo solver, *QuickCap*™[4], and give a procedure for estimating RMS (root-mean-square) error for any capacitance extractor. Lastly, we briefly discuss issues relating to coupling capacitance between specific nets.

II. METHODOLOGY FOR FULL-CHIP EXTRACTION

To extract interconnect capacitance in complex ICs, Monte Carlo field solvers can be significantly more efficient than other physically based solvers. Even so, without resorting to parallel processing, it is doubtful that a Monte Carlo solver could extract *all* net capacitances within a modern digital chip.

Our proposed methodology for this type of problem is to employ a faster, though less accurate, capacitance extractor for noncritical nets, typically most of the nets. For critical nets we employ an extractor based on Monte Carlo. Noncritical nets are those for which the accuracy of the fast extractor is, for practical purposes, sufficient. Naturally, the accuracy of the fast extractor must be quantified to allow *a priori* partitioning of nets into critical and noncritical categories. If this is not possible, the next best approach is to use the fast method to find capacitance on all nets and then use the accurate extractor to extract as many accurate capacitance values as time allows.

Our full-chip methodology assumes that computational error in extracted capacitance is known. Monte Carlo extractors can furnish with each capacitance value an absolute numerical error, in terms of a statistical standard deviation. However, we must generally estimate the absolute error of fast, lower-accuracy extractors.

We propose that fast-extractor error be characterized by direct comparison with Monte Carlo results over a reasonable sample population of the nets in the actual full-chip environment. The simplest characterization is an estimate of an RMS error over the sample of nets. RMS error is, in fact, an analog of Monte Carlo statistical error. Note, RMS error may not generally be sufficient to describe large-error probabilities, even if the error distribution appears to be gaussian.

* The work of the authors is supported in part by the SRC Center for Advanced Interconnect Science and Technology at Rensselaer, Contract IC-408.

0-7803-3775-1/97/$10.00 © 1997 IEEE.

III. PERFORMANCE OF MONTE CARLO EXTRACTORS

The computational performance of a Monte Carlo extractor can be markedly different from other types of physically based extractors. For example, *QuickCap*'s convergence time for evaluating self-capacitance of a net depends weakly both on the size of the net and on the number of surrounding nets.

In addition, memory requirements are extremely low (no numerical meshing) and computational error depends only on run time. Below, we give a brief theoretical overview of our example Monte Carlo extractor *QuickCap*. We also present some data relating net size and convergence time for this extractor.

A. Monte Carlo Integration and Random-Walks[1]

To find the self-capacitance C of a particular net, *QuickCap* performs a Monte Carlo integration to evaluate the nested integral below:

$$C = \oiint d^2 r_1 \, \varepsilon(r_1) E(r_1),$$

$$E(r_1) = \oiint d^2 r_2 \, f(r_2 - r_1) V(r_2),$$

$$V(r_k) = v(r_k) + \oiint d^2 r_{k+1} \, g(r_{k+1} - r_k) V(r_{k+1}), \; k=2, 3, \ldots; \quad (1)$$

where E is the component of electric field normal to the surface of integration; f and g are known functions, independent of geometry; and v is a known function, dependent on the net locations. The domain of the integral for C is a surface enclosing the net of interest. The domain of the integral for $E(r_1)$ is a surface enclosing r_1. The domain of the integral for $V(r_k)$ is the part of a surface enclosing r_k that not coincident with the surface of any net.

Because Eq. 1 contains no approximations, the principal source of computational error associated with Monte Carlo integration is statistical error.

The series of points associated with a single Monte Carlo estimate of the capacitance integral can be described as a floating random walk—floating because the size of a walk step is proportional to the distance to the nearest net.

B. The Effect of Net Size on Convergence Time

The number of Monte Carlo integration samples (or, equivalently, the number of random walks) required to find the total capacitance to within a given statistical error is a weak function of net size and density of surrounding nets. The number of samples depends, more so, on the environment around the net. The data in Table 1 illustrate the effect of net size on convergence time at 1% standard-deviation error. *QuickCap* was used here to extract self-capacitance values of sample nets in a 450-MB geometric IC data base. Capacitance and, presumably, net length and net complexity vary by a factor of 700; while, convergence time varies by a factor of 4.5 and the number of samples varies by a factor of 2.

TABLE 1
CONVERGENCE TIME *vs.* NET SIZE

Capacitance (±1%)	*Convergence Time*	*Number of Samples*
0.0119pF	70s	132,000
0.127pF	80s	127,000
0.906pF	40s	233,000
8.63pF	180s	252,000

IV. COMPUTATIONAL ACCURACY OF EXTRACTORS

We have suggested in our proposed extraction methodology that error of a fast noncritical-net extractor can be characterized by comparison with an accurate, but slower, Monte Carlo extractor. In this section, we first estimate capacitance errors associated with a variety of "generic" extractors. Then, we give results for an example fast layout-parameter extractor (LPE).

To estimate the error associated with various extractors, we represent each type of extractor as, essentially, ideal within an *effective* calculation window. Thus, error will depend on the environment outside the window. We use *QuickCap* to compute variations in net self-capacitance within the window as a function of nets outside. This approach furnishes worst-case error estimates. Longer nets will have a lower relative error because of cancellation effects resulting from geometric variations in the actual chip layout.

A. Accuracy of Generic Extractors

Fig. 1 shows a pair of 2D capacitance problems that are identical within a window, but have different environments outside. If an extractor does not include nets outside the window, it can not differentiate between the two problems. Fig. 1 illustrates the worst-case variation for a 5-level-metal process with 1μm layer thicknesses and a 1μm minimum line width: the worst case consists of one small net or wire in top-level metal with no other objects inside the window. (Note, much greater capacitance variation would be observed, actually, if nets just outside the window were connected to the net within.)

Fig. 1: 2D examples depicting the influence of the external window environment on critical net self-capacitance. Metal thickness and vertical spacing are 1μm. The critical net is 1μm by 1μm, centered 9.5μm above the ground plane. The window is 20μm wide. Nets outside the window, including ground, extend laterally to infinity.

Fig. 2 is a plot of the relative capacitance variation between the two cases of Fig. 1 as a function of window size. This can be used to estimate the error inherent in methods that can be characterized with a finite calculation window.

Table 2 lists estimates of the worst-case (rare) errors and RMS errors of various generic extractors due to finite window size. After estimating effective window sizes for each extractor, we use data from Fig. 2 to estimate the RMS errors. Here are some of our considerations in estimating window sizes for various extractors:

- *Simple LPE.* Extraction takes into account nets above and below the net of interest.
- *Complex LPE.* Extraction takes into account, as well, lateral nets within a few microns.

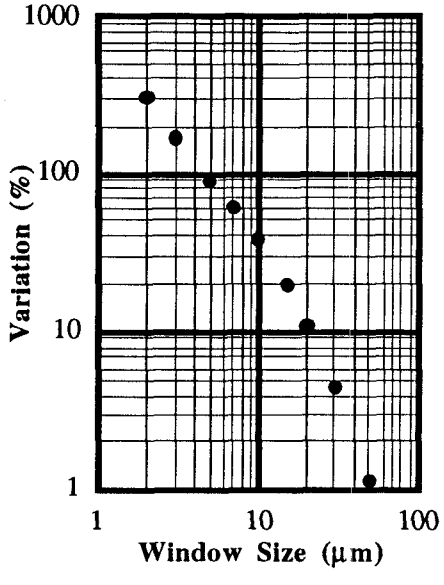

Fig. 2: Relative variation in capacitance between the two geometries depicted in Fig. 1 as a function of window size.

- *Library or Pattern Matching.* Extraction uses pre-calculated structures that match or nearly match the local geometry.
- *Mesh.* Extraction is by solution of the underlying Laplace equation. The numbers given apply to window-induced error, only. Discretization error is assumed to be negligible.

The window sizes we list for these methods are rough estimates. Rare-error values are taken directly from the Fig. 2. Because of statistical error cancellation in large nets, we are assuming that rare error represents 3–6 standard deviations. Methods that use smaller windows involve more statistical cancellation over the length of the net and thus will have less probability of a rare-error event. We believe that present mesh methods can not easily, if at all, perform high-accuracy extraction for large nets because of the large window sizes required.

TABLE 2
ESTIMATES OF CALCULATION ERRORS FOR SOME GENERIC EXTRACTORS

Method	*Window Size*	*Rare Error*	*RMS Error*
Simple LPE	3μm 5μm	200% 90%	15%?
Complex LPE	5μm 7μm	90% 60%	10%?
Library	5μm 15μm	90% 20%	7%?
Mesh	15μm 30μm	20% 4%	5%? 1%?

B. A Simple LPE Example

We now characterize the error of a simple LPE extractor. In this extractor, net self-capacitance is described by a weighted sum of the areas for each interconnect layer and via layer.

We would like to estimate the normalized RMS error σ_X of the LPE extractor, where the normalized RMS error of n extracted capacitance values is

$$\sigma_X = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(\frac{C_{X,i} - C_i}{C_i}\right)^2} \,. \tag{2}$$

$C_{X,i}$ is the extracted capacitance value of the ith net and C_i is its exact capacitance.

Our methodology uses Monte Carlo results, rather than exact capacitance values, to find an *uncorrected* normalized RMS error σ_{XM}. This error includes intrinsic Monte Carlo statistical error.

$$\sigma_{XM} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(\frac{C_{X,i} - C_{M,i}}{C_{M,i}}\right)^2} \,, \tag{3}$$

where $C_{M,i}$ is the ith capacitance value found by the Monte Carlo method. The normalized RMS error may be approximately found from the Monte Carlo statistical error σ_M. For reasonably accurate Monte Carlo data:

$$\sigma_X \approx \sqrt{\sigma_{XM}^2 - \sigma_M^2} \,. \tag{4}$$

We have used *QuickCap* with this approach to evaluate the RMS error of a simple LPE extractor on a small layout. The LPE capacitance is a weighted sum of the areas of 7 interconnect and via layers. We first determined the best fit of the 7 coefficients using *QuickCap* results. Over 153 nets this fit gave an RMS error of 18%. Out of these 153 nets, the worst error was 54%. These values are consistent with the simple-LPE error values listed in Table 2.

V. COUPLING CAPACITANCE

While the major thrust of our presentation concerns self-capacitance extraction, the issue of coupling capacitance between nets deserves some discussion. We previously noted that self-capacitance convergence time for Monte Carlo extractors is weakly dependent on net size. This does not apply to the convergence time for the coupling capacitance between two specific nets. On a large net with, say, thousands of neighboring nets, the individual capacitance values to all of the nets can *not* be found to high accuracy in reasonable time.

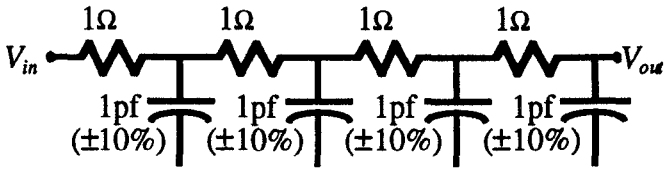

Fig. 3: Statistical cancellation during circuit analysis. The Elmore delay time for this circuit is 10ps (±5.5%).

Often in large digital IC designs, no particular *small* coupling capacitance value significantly affects the analysis of a net. ("Small," here, means small relative to the self-capacitance of the net in question.) Those coupling-capacitance values that *are* of importance individually are also generally large enough that the Monte Carlo convergence time required for reasonable statistical accuracy is acceptable—that is, on the order of the time required to find the total net capacitance.

Nonetheless, small capacitances associated with a net can be collectively important. Monte Carlo extractors demonstrate a great efficiency in this situation: the weighted sum of many small capacitances will have reduced error due to statistical cancellation. Consequently, relatively large errors in coupling capacitances are reduced in subsequent circuit analysis. By way of example, Fig. 3 shows a simple RC circuit where each capacitance element has a statistical uncertainty of ±10%. The Elmore delay time for this circuit has a statistical error of ±5.5%, lower by almost a factor of 2 due to statistical error cancellation.

VI. SUMMARY

We have proposed a methodology for full-chip capacitance extraction. It relies on a physically based Monte Carlo extractor for high-accuracy extraction of critical net capacitance in conjunction with a faster, but less accurate, empirical extractor for noncritical nets. Our methodology requires, as well, that the Monte Carlo extractor be used to quantify absolute capacitance errors in the fast-extractor.

REFERENCES

[1] Y.L. Le Coz and R.B. Iverson, "A Stochastic Algorithm for High-Speed Capacitance Extraction in Integrated Circuits", *Solid-St. Electron.* 35, pp. 1005–12, 1991.

[2] Y.L. Le Coz, H.J. Greub, A. Garg, J.F. McDonald, and R.B. Iverson, "A Floating Random-Walk Algorithm for Extracting IC-Interconnect Capacitances: Simulation and Experimental Verification of Critical-Path Delay within a Fast-RISC ALU", *Proc. 13th Int'l VLSI Multilevel Interconnection Conf.*, Santa Clara, CA, pp. 230–2, June 1996.

[3] Y.L. Le Coz, H.J. Greub, A. Garg, P.M. Campbell, S.A. Steidl, J.F. McDonald, and R.B. Iverson, "Interconnect-Limited Oscillations within a Fast-RISC ALU Implemented with a 50-Ghz HBT Process using Anisotropic Polyimide ILD: Simulation and Measurement", *Proc. 14th Int'l. VLSI Multilevel Interconnection Conf.*, Santa Clara, CA, June 1997, in press.

[4] *QuickCap* is a trademark of Random Logic Corporation, Fairfax, VA.