Fast and Accurate Waveform Analysis with Current Source Models

Vineeth Veetil, Dennis Sylvester, David Blaauw

Dept of EECS, University of Michigan, Ann Arbor – 48109

tvvin,dennis,blaauw@eecs.umich.edu

ABSTRACT

Recently current source models (CSMs) have become popular for use in standard cell characterization and static timing analysis. However, there has not been any detailed study of what aspects of the gate parasitics and DC current source behavior should be modeled for sufficient accuracy, and there have been no results reported incorporating a CSM with the above complexity into a timing analysis flow with reasonable runtime. This paper addresses these two limitations by investigating complexity/accuracy tradeoffs in CSMs. We then present a novel technique to perform fast, accurate waveform analysis using current source models. Timing analysis results on benchmark circuits show significantly reduced errors (and error spreads) compared to a traditional Thevenin-based flow. In terms of $\mu+\sigma$ percentile, we gain by 20-150% in slew through this approach.

I. Introduction

Traditional standard cell libraries have modeled logic gates as voltage sources with precharacterized slopes and 50% delay points. There are a number of problems inherent in this simple approach that have become paramount in modern nanometer scale CMOS. Signal integrity issues are very dependent on signal waveform shapes. This is not captured under this model. Also, the mapping of complex loads to a single Ceff during the process is often criticized for its inability to capture the complexity of the loads.

There have been several approaches towards tackling this issue. Some of them attempt to keep the model independent of the load. The authors of [1] propose an RC interconnect insensitive linear time varying model. In [4], the authors propose linear and non-linear driving models. The work emphasizes the need of proper basis functions and sets up a broad framework for various driver model possibilities.

Other approaches to keeping the model independent of the load characterize the DC behavior of the gate completely, and derive transient behavior with the help of models for the parasitics in the gate. A model called 'Blade' was proposed by Croix and Wong [2]. Here, the basic idea is to first model the DC current characteristics of the gate seen as a two-port device, the ports being the active input and the output. The parasitic behavior is then modeled with a single calibrating capacitance from the output to the ground. The issue with such a single capacitance model is that it does not adequately capture non-linearity in the capacitance. Further, the approach in runtime is to perform numerical integration at the gate output waveform, which is very expensive.

In [6], the authors map the time shift parameter proposed in [2] to an RC ladder, and try to model non linearity in capacitance. This approach is followed up by [5] who proposed a multi-port current source model to deal with multiple switching effects. This approach, while accurate, adds significant complexity to the model and it is unclear how much the computational efficiency will be in the runtime engine for timing analysis. An interesting approach towards statistical analysis using current source models considering process variations has been proposed by [8]. These approaches are interesting, yet they fall short of giving us an efficient runtime engine. In particular, there is no work showing that the parasitic model and the DC current source model can be incorporated in an efficient runtime engine.

We attempt to address the specific issues to be dealt with in implementing a practical timing analysis approach with an efficient runtime engine, based on current source models.

Our contributions are twofold. First, we model the *DC current behavior* and the *transient behavior* for optimality in the accuracy vs runtime tradeoff. We make observations regarding efficient ways to capture the *DC current source model*. We propose a *Bicubic Spline based DC Current Source Model*, and show that this is highly accurate, as well as amenable to fast runtime analysis. For modeling the *transient*, we show that a *two-piece output capacitance* and a *time shift* parameters make an accurate model. The model has different parameters for high to low and low to high transitions. The *time shift* parameter in this work is a function of output voltage for a given *sequential cell* and a constant for a given *combinational cell*.

Second, we propose a solution for fast and accurate run time waveform analysis utilizing the current source model. Our work is the missing link between the fact that Weibull functions represent waveforms effectively [3], and that CSMs have the potential to be the future in timing analysis. Specifically, we propagate voltage waveforms as *Weibull functions* and exploit the properties of our current source model to efficiently solve for Weibull parameters at every gate. Additionally, the method can be easily extended to more elaborate load models for the future, and for the case of noise analysis, as well as to easily model process variations. Timing analysis results on benchmark circuits show significantly reduced errors (and error spreads) compared to a traditional Thevenin-based flow. In terms of $\mu+\sigma$ percentile, we gain by 20-150% in slew and up to 220% in delay through this approach.

We present our work in the following sections. Section II deals with our approach towards precharacterization of gates in the context of current source models. In Section III, we describe a technique for fast and accurate waveform analysis during runtime. Section IV shows results on benchmark circuits while Section V concludes the paper.

II. Precharacterization

The precharacterization step in Current Source Models, for a given process/voltage/temperature (PVT) corner, involves two stages, as mentioned before. The gate is seen as a two port device – the active input port and the output port. First, the DC behavior is modeled. For this, the DC current sourced by the two port device is fitted as a function of the port voltages. The parasitic behavior is then modeled with a capacitance-based or charge-based model.

A. Bicubic Spline based DC Current Source Model

For a given process/voltage/temperature (PVT) corner, DC supplies are attached to input and output pins. These are swept from 0-ΔV to 1.2+ΔV and a 2-D table of output current versus input and output voltages is obtained [2]. Unlike in [2], we are sweeping beyond rails. The next step is to extract an I(Vin,Vout) model from this data. We propose a *Bicubic spline model*. We compare a fourth order polynomial fit in two variables to the *bicubic spline* fit [10] for the data, and find that the bicubic spline fit is stable with higher accuracy. Figs. 1(a) & (b) show typical error plots for the two models. A fourth-order polynomial fit is unstable at the steady-state points (0, 1.2) and (1.2, 0) because of sharp trends in the region. Also, the peak error magnitude is an order lower for bicubic spline. Table 1 shows data for the stagewise timing analysis performance of standard cells in an industrial 90nm library for the two approaches, where it is clear that the proposed approach has much higher accuracy. From our experiments, we also observe that a bicubic spline with 3 by 3 internal knots is comparable in accuracy to 4 by 4 internal knots for this purpose.

B. Modeling Transient

We use a two-piece intrinsic capacitance model as a function of the output gate voltage, and a constant time shift for combinational library cells. For sequential library cells, i.e. latches, we use output voltage dependent time shift.

We find it worth mentioning that the *transient* model in *current source models* is a set of calibrating parameters, and does not correspond to the actual parasitic capacitance values. Hence, the actual parasitic model may be complex, but the output voltage curve is observed to respond smoothly in spite of this. This is the basis of our choice of a simple *transient model*.

### C. Model for sequential library cells

See Fig 3 below for the sequential cell considered here. Consider the case of enable inputs 'gi' and 'gni' firing. It is assumed that one of the enable inputs is at the final steady state throughout the transition, with the intuition that it does not play a major role in the transition. For example, with D input initially at 0, when 'gni' and 'gi' fire, it is assumed that 'gi' is fixed at 1.2V. Also, for the same transition, it is assumed that the input pin, originally from Q', to the feedback gate is fixed at the initial value of 1.2V. The justification is that 'gni' has a faster falling transition than Q', and thus has a larger effect on disabling the pull down network. Also, the pull up network is disabled by the time Q' is low enough to enable it.

Table 1. Comparison of stagewise timing analysis for a bicubic spline fit (spline) vs. a fourth order polynomial fit (Poly) for standard cells in an industrial 90nm library.

	10-30% (% error)		50% (% error)		20-80% (%error)	
	Spline	Poly	Spline	Poly	Spline	Poly
Avg	-0.5	2.8	-0.8	-1.5	0.6	2.6
Stdev	0.9	2.7	0.3	1.0	0.4	1.7
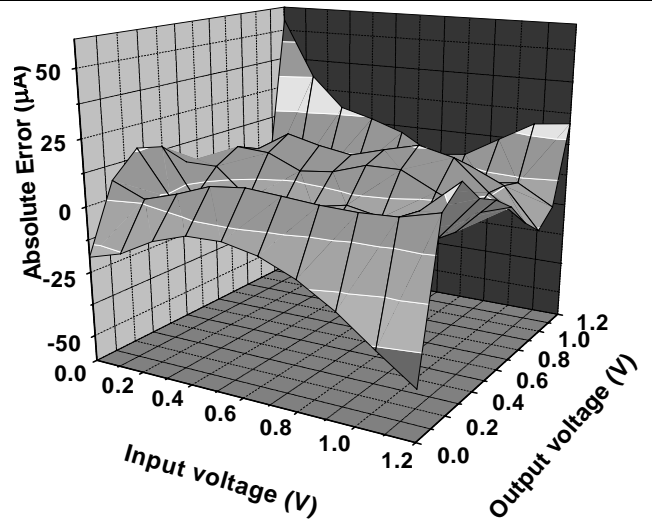
(a)
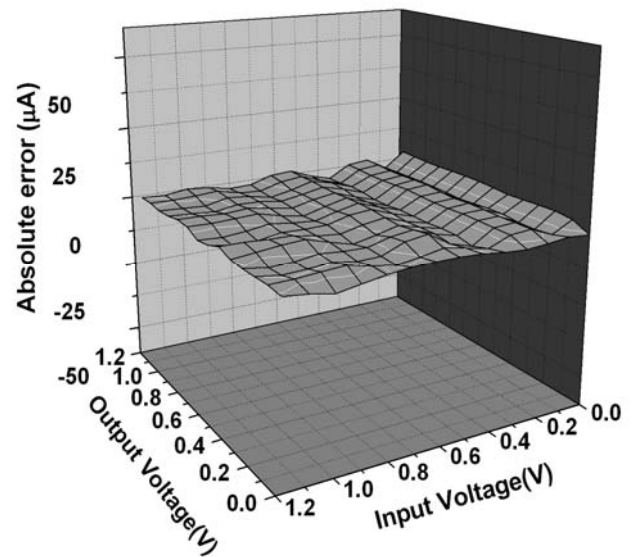
(b)

Fig 1: (a) Polynomial-based fitting model error in current relative to SPICE as a function of input and output voltages. (b) Spline-based fitting model error in current relative to SPICE.

With these assumptions, we retain single input switching models for individual gates. Thus, we can use the *DC current source* model and *transient model* for each gate involved in the transition for this situation. Fig 4 below shows a comparison of waveforms with one or more of the assumptions stated above with the correct waveform.

Now, we observe that the time difference in the waveform modeled with assumptions and the correct waveform, for different loads at QN, is a function of the voltage at Q'. This

function is independent of the load at QN. Thus, we add a voltage dependent time shift correction term at Q', a function of V(Q'), to complete the model.
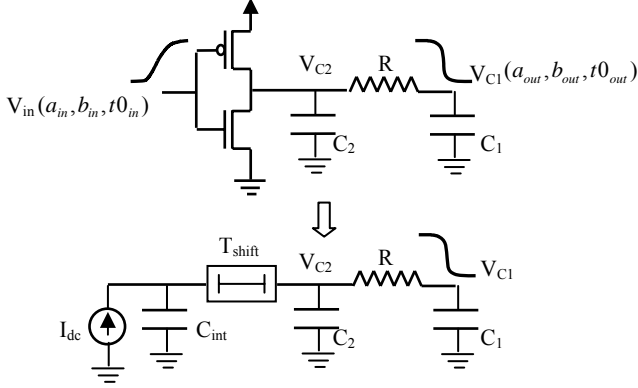

Fig 2. Schematic of the proposed modified Blade-based model.

III. Weibull-based Runtime Engine

This section presents a novel method to perform timing analysis for a circuit. Our method exploits the fact that the *bicubic spline based DC current source model* obeys smoothness properties, and therefore lends itself to various simple mathematical analyses.
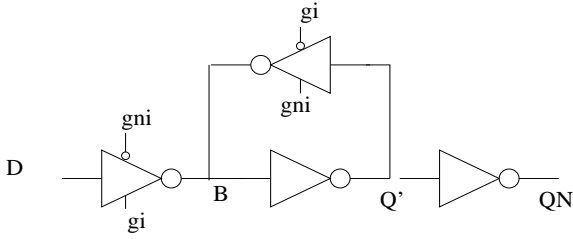

Fig 3. Schematic of the latch used for analysis.

It has been noted in [3] that the cumulative distribution function of a Weibull function is very efficient in capturing waveform shape. This, coupled with the *Bicubic Spline based DC current source model*, enables a simple and fast yet accurate method to propagate waveforms as *Weibull-based functions*.

A. Basic Concept and Flow

For simplicity, we consider here the simplest three-parameter Weibull function. CDF of Weibull functions can be written as follows:

$$W(a,b,t0) = 1 - \exp(-((t-t0)/b)^a) \qquad (1)$$

Refer to Fig 2. Let the rising input waveform to a gate be represented by $V_{dd} \times W_{in}(a_{in}, b_{in}, t0_{in})$. Let output waveform V_{c1} be of the form $V_{dd}(1 - W_{c1}(a_{out}, b_{out}, t0_{out}))$. Note that this is for an output falling transition of a gate ; for the output rising case, the forms are interchanged. Also, let the bicubic spline model of I_{dc} be as follows (refer Fig 2):
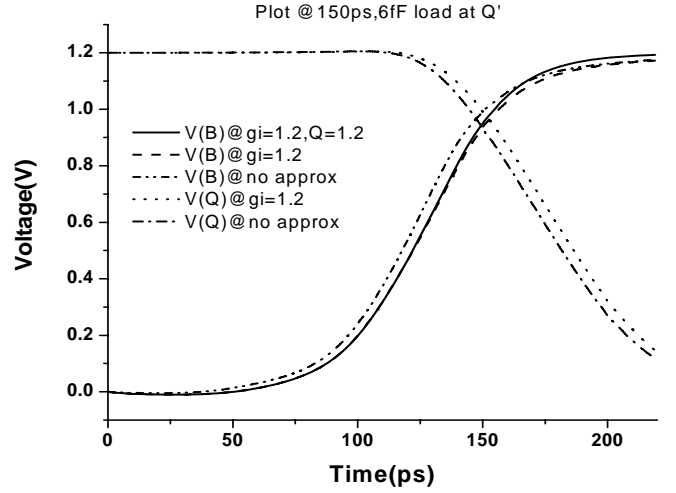

Fig 4. SPICE simulated waveforms based on various assumptions regarding the switching behavior in the latch of Fig. 3.

$$I_{dc} = \sum_{i=0}^{3} \sum_{j=0}^{3} \alpha_{ij} V_{in}^{i} V_{c2}^{j} \qquad (2)$$

where α_{ij} are coefficients of a piecewise bicubic polynomial.

Now, consider our model of a library cell loaded with a π load, as in the schematic in Fig 2. The KCL equation for current in this situation can be written as

$$I_{dc} + I_{load} = 0$$

Our aim is to come up with parameters $(a_{out}, b_{out}, t0_{out})$ to minimize the error function given by

$$f(t) = I_{dc} + I_{load}$$
$$= (\sum_{i=0}^{3} \sum_{j=0}^{3} \alpha_{ij} V_{in}^{i} V_{c2}^{j}) + (\partial(p_1 W_{c1})/\partial t + \partial^2(p_2 W_{c1})/\partial t^2) \qquad (3a,b)$$

where

$$V_{c2} = V_{dd}(1 - W_{c1} - RC_1 . \partial W_{c1}/\partial t)$$
$$V_{c1} = V_{dd} . (1 - W_{c1}(a_{out}, b_{out}, t0_{out}))$$
$$p_1 = V_{dd} . (C_1 + C_2')$$
$$p_2 = V_{dd} . R . C_1 . C_2'$$
$$C_2' = C_2 + C_{int}$$

The first term in eqn 3(b) refers to the current sourced by the DC current model (refer to as source current), and the sum of second and third is the current flowing into modeled and real loads (refer to as load current) (Fig 6). Therefore, the problem can be formulated as solving for parameters $(a_{out}, b_{out}, t0_{out})$ such that the error in f(t) is minimized for all t. A least square approximation by integrating *f(t)* [2] for all t may be tried. However, the function is not explicitly integrable. Also, the large number of parameters makes a look-up table form for the integration infeasible – it will include parameters a_{in}, b_{in}, a_{out}, b_{out} and $t0_{out}$, along with the time limits of the integration, say t_1 and t_2 (since α_{ij} are coefficients of a piecewise bicubic, this involves piecewise integration).

However, the function is continuous and differentiable with respect to all three parameters; therefore an iterative method

may be adopted to solve a system of non-linear equations. We considered several methods including Newton Raphson and Conjugate Gradient based steepest descent method. We intend to solve the following system of non-linear equations by Newton-Raphson iterations (where $f(t)$ is the function as above).

$$f(t_1, a_{out}, b_{out}, t0_{out}) = 0$$
$$f(t_2, a_{out}, b_{out}, t0_{out}) = 0 \quad (4)$$
$$f(t_3, a_{out}, b_{out}, t0_{out}) = 0$$

The time points (t_1, t_2, t_3) are chosen to be the 20, 50 and 80% transition points of V_{C1}. Fig 5(b) illustrates this. Note that this figure considers 10 and 15% points; they are for accuracy enhancement to be described shortly.

In our case, Newton Raphson is observed to converge faster, typically in 3-4 iterations. Steepest descent methods have the inherent disadvantage because the parameters $(a_{out}, b_{out}, t0_{out})$ we search for are not homogenous quantities.

For starting values of a_{out}, b_{out} and $t0_{out}$ – we need the following fitting coefficients per gate:

$$t_{d\_out} = a_1 + a_2 * tr\_in + a_3 * cap$$
$$t_{tr\_out} = a_1 + a_2 * tr\_in + a_3 * cap \quad (5)$$

where t_{d_out}, t_{tr_out} are output delay and slew respectively, cap is the gate load cap, tr_in is the input slew. This can either be taken from the vendor device datasheet or extracted during device characterization. The full flow is summarized in figure 5(a).

B. Enhancing Accuracy

It is possible to improve the accuracy by using basic understanding of the current flow in a gate. It is observed that though the error function (when seen as a function of t) at time points of 50% and 80% transition points in output voltage (corresponding to equations 4(ii) and 4(iii)), are smooth in the neighbourhood of t, the error function near the 20% transition point (equation 4(i)) can have local fluctuations. For an improved solution, therefore, it is desirable to fit the early part of the transition (corresponding to the 20% point) with more points. Note that the above procedure in section III A basically seeks to obtain a charge flow waveform by matching its derivative, i.e., current at t_1, t_2 and t_3. Near the 20% point t_1, it helps to obtain an approximation for the average current flow in the neighborhood of t_1 in t, and use this quantity directly as the error to be minimized, instead of current at just t_1. For this, we derive an approximation for the total charge flow between two time points $t_{1,0}$ and $t_{1,2}$ in the neighborhood. Equation (6) below computes this error charge err_c($t_{1,0}$,$t_{1,2}$). We then divide it by the time interval. 6(b) means that we resort to a simple quadratic interpolation for I_{dc} using calculated values at time points $t_{1,0}$, $t_{1,1}$ and $t_{1,2}$ near the 20% transition region. We have chosen (10%, 15%, 20%) of output voltage transition for this purpose (refer Fig 5(b)). Now, since comparing one charge quantity and two current quantities (at 50% and 80% points of the transition) in a system of equations creates difficulties in convergence, we normalize this charge term with the time interval over which the approximation is considered. Thus, effectively the first quantity becomes an average current as in eqn 6(d). This is used in place of f(t_1) in eqn (4).

$$err\_c(t_{1,0}, t_{1,2}) = \int_{t_{1,0}}^{t_{1,2}} f(t)dt = \int_{t_{1,0}}^{t_{1,2}} (I_{dc} + I_{load})dt$$

$$I_{dc} = \sum_{i=0}^{2} I_{dc}(t_{1,i}) * \frac{(t - t_{1,(i+1)\bmod 3}) * (t - t_{1,(i+2)\bmod 3})}{(t_{1,i} - t_{1,(i+1)\bmod 3}) * (t_{1,i} - t_{1,(i+2)\bmod 3})}$$

$$\int_{t_{1,0}}^{t_{1,2}} (I_{load})dt = (p_1(t_{1,2})W_{c1}(t_{1,2}) - p_1(t_{1,0})W_{c1}(t_{1,0}))$$

$$+ \left[p_2(t)\partial W_{c1}/\partial t \right]_{t_{1,0}}^{t_{1,2}}$$

$$f(t_{1,0}, t_{1,2})_{eff} = err\_c(t_{1,0}, t_{1,2}) \bigg/ (t_{1,2} - t_{1,0})$$

$$(6-a, \ b, \ c, \ d)$$

C. Mapping a non-linear parasitic model

As mentioned before, we used a 2-piece capacitance to model gate output parasitic (different for rise and fall) – one for 10-50% transition region, and the other for 50-90% transition region. In the iterations during runtime, this model is mapped to a single linear function of voltage V_{C1} with the constraint that average load currents in 10-50% and 50-90% regions remain unchanged. Note that the model is updated every iteration to satisfy the above constraint. This illustrates the flexibility of the proposed runtime model in terms of ability to handle parasitic models.

D. Time shift for error compensation

The current flow before 5% output transition is not accurately represented in the Weibull-based formulation. This is related to the nature of the derivative of weibull cdf near the zero intersection. This relatively small charge is easily mapped to a time shift for the entire waveform.

Another *time shift compensation* is as follows. Refer to Fig 6. The finite area in the error function plotted can be compensated for in a way similar to that described above. We divide the transition time into 3 regions, i.e., 0%-10%, 10%-50%, and 50%-80% of the output transition, and roughly calculate the error charge in each region. In this case the compensation at a point (e.g., 50%) only applies for error charge preceding that point.

After the above corrections are made, Weibull parameters are generated again, comprising inputs to the next gate. It may be noted that this error correction is optional and is aimed at improving accuracy.

IV. Results

We performed simulations on benchmark circuits synthesized in an industrial 90nm technology. The results of the Weibull-based analysis were compared with numerical integration results based on the current source model. For comparison of accuracy with a model comparable in time efficiency, we use a Thevenin model. This converges in less than 4 iterations for most cases [7]. This model is at the heart of most of the timing analysis tools.

Note that our experiment sought to compare performance of the two methods in moderate to high resistive shielding conditions, since these represent the most difficult cases traditionally. Hence, the benchmark circuits were synthesized targeting such load conditions so that the various approaches can be evaluated in a stringent environment.

Table 2 shows a comparison of the two methods for several ISCAS85 benchmark circuits [9]. As a result of the improvements shown, the error at the μ+σ percentile (68th percentile for normally distributed errors) is reduced by 20-150% in slew. For computing 50% delay the new approach provides up to 220% smaller error at the μ+σ percentile. Fig 7 and 8 show data for large ISCAS85 benchmark circuits. Fig 8 visually depicts how errors in slew rate estimation are reduced with this approach compared to a Thevenin-based flow. Fig. 7(a) shows slew rate error of our approach. Figs 7(b), (c) show the delay performance for our approach.

Thevenin-based models are criticized for being unphysical in mapping any complex load to a single Ceff. This is precisely the factor that leads to larger errors in slew rate for the Thevenin case here. We have observed that errors in 10-30% and 70-90% transition time improve substantially because of the underlying physical approach of current source models. This coupled with comparable efficiency is the advantage of the proposed approach. As noted before, convergence of the Newton Raphson system occurs in 3-4 iterations, which is similar to the Thevenin approach.
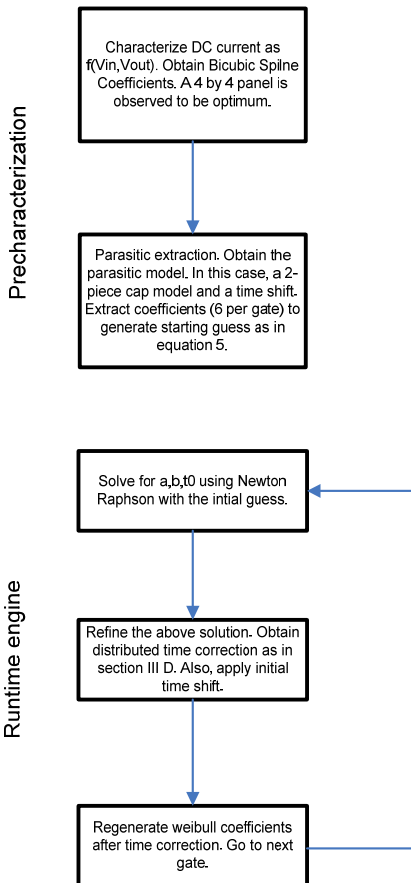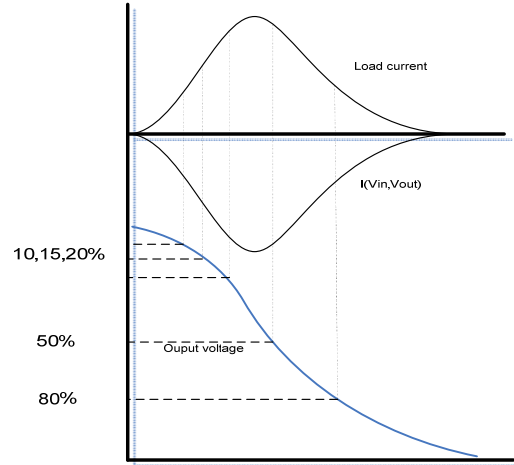

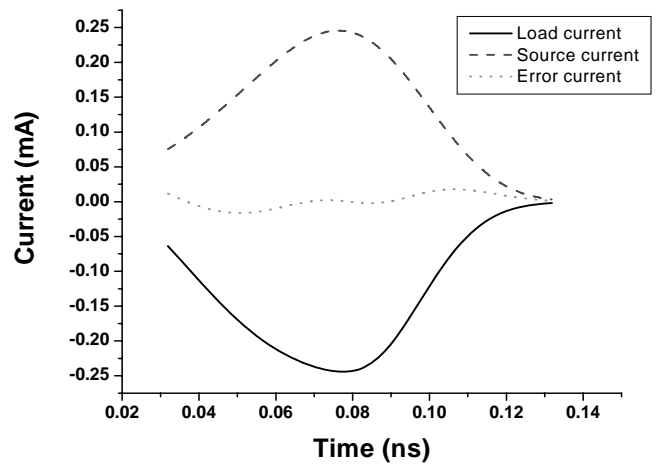Fig 5(b) Points used in determining current error function.

Fig 6. Typical waveforms for source and load currents as in the proposed model for an output falling case. The difference is the error function (referred to as f(t)) and is compensated for using 'Time Shift' techniques.

Fig 5. (a) Proposed flow chart

Table 2. Error statistics compared to SPICE of delay and slew for proposed and traditional techniques for various benchmark circuits.

Ckt	Weibull slew error μ+σ (%)	Thevenin slew error μ+σ (%)
C3540	4.6	7.7
C499	3.2	7.6
C2670	3.2	7.5
C1908	5.1	6.1
C880	2.3	5.7

Ckt	Weibull, delay error μ+σ (%)	Thevenin, delay error μ+σ (%)
C3540	3.2	7.2
C499	3.9	3.6
C2670	5	7
C1908	2.3	7.5
C880	3.5	7.4

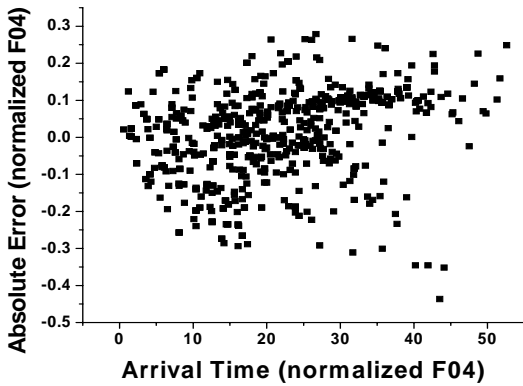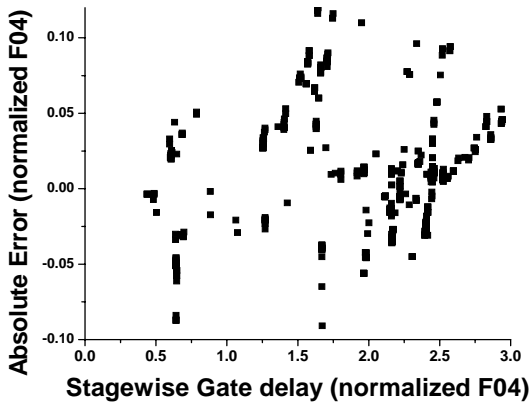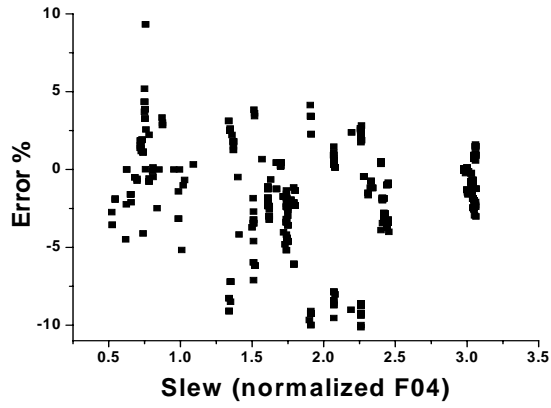Fig 8. Error histograms for slew estimations in two large ISCAS85 circuits given a primary input excitation.

Fig 7(a) % error in slew (normalized F04) (b) Absolute error (normalized F04) in gate delay (normalized F04) (c) Absolute error (normalized F04) in arrival time (normalized F04).

V. Conclusions

We have investigated the importance of various modeling decisions on the accuracy and complexity of CSMs. In particular we find that a bicubic spline approach to fitting DC current source as a function of input and output voltages is accurate and lends itself to efficient manipulation in timing analysis. Furthermore, we show that the use of a 2-piece internal capacitance model provides good accuracy, while remaining tractable. We then propose a Weibull-based method to perform waveform analysis using the suggested CSM. This technique allows the higher accuracy capabilities of current
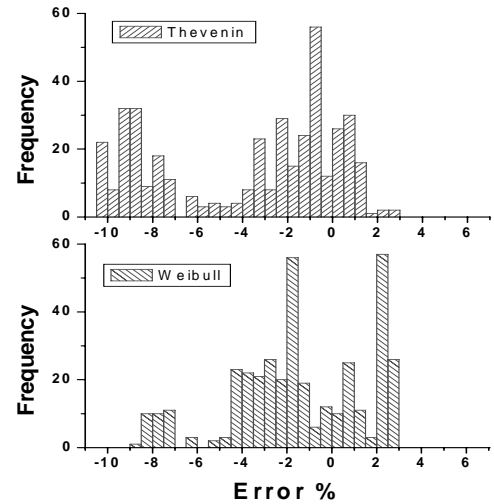
source models to be leveraged in efficient static timing analysis tools. We show that errors in delay and slew across gates in various benchmark circuits are reduced substantially (by $\mu + \sigma$ error quantile) compared to traditional Thevenin-based approaches. In addition, the approach retains computational efficiency as the Newton-Raphson approach converges in 3-4 iterations, as is the case in Thevenin-based timing flows. Also, very importantly, the approach can be scaled to other parasitic models that have been proposed with a reasonable complexity.

References

[1] C.K. Tsai, M. Marek-Sadowska: "An Interconnect Insensitive Linear Time-Varying Driver Model for Static Timing Analysis," *International Symposium on Quality Electronics Design*, pp. 654-661, 2005

[2] J.F. Croix and D.F. Wong, "Blade and Razor: Cell and Interconnect Delay Analysis Using Current-Based Models", *Design Automation Conference*, pp. 386-389, 2003.

[3] C.S. Amin, F. Dartu, and Y.I. Ismail, "Weibull-based analytical waveform model," *International Conference on Computer-Aided Design*, pp. 161-168, 2003.

[4] B. Tutuianu , R. Baldick, and M.S. Johnstone, "Nonlinear Driver Models for Timing and Noise Analysis", *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, pp. 1510-1521, Nov. 2004.

[5] C.S. Amin, C. Kashyap, N. Menezes, K. Killpack, and E. Chiprout, "A multi-port current source model for multiple-input switching effects in CMOS library cells", *Design Automation Conference*, pp. 247-252, 2006.

[6] P.Li, E.Acar,"A Waveform Independent Gate Model for Accurate Timing Analysis", pp 363-365, ICCD 2005.

[7] F.Dartu, N.Menezes, L.T.Pileggi, "Performance Computation for Precharacterized CMOS Gates with RC Loads", *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, pp. 544-553, May 1996.

[8] H.Fatemi,S.Nazarian,M.Pedram, "Statistical logic cell delay

analysis using a current-based model", *Design Automation Conference*, 2006.

[9]Brglez, F. and Fujiwara, H. (1985). *Neutral netlist of ten combinational benchmark circuits and a target translator in FORTRAN*, Special session on ATPG and fault simulation, Proc. IEEE Int. Symp. Circuits and Systems, 663—698

[10] C.D.Boor, *A Practical Guide to Splines,* Springer, 2001.