

Automated Extraction of Accurate Delay/Timing Macromodels of Digital Gates and Latches using Trajectory Piecewise Methods

Sandeep Dabas

Department of ECE
University of Minnesota, MN-55455
Tel: 612-626-7203
Fax: 612-625-4583
email: dabas001@umn.edu

Ning Dong

Texas Instruments Inc.
Dallas, TX-75231
Tel: 214-480-1396
Fax: 214-480-7287
email: ningd@ti.com

Jaijeet Roychowdhury

Department of ECE
University of Minnesota, MN-55455
Tel: 612-626-7203
Fax: 612-625-4583
email: jr@umn.edu

Abstract—We present a fundamentally new approach, ADME, for extracting highly accurate delay models of a wide variety of digital gates. The technique is based on trajectory-piecewise automated nonlinear macromodelling methods adapted from the mixed-signal/RF domain. Advantages over prior current-source models include rapid automated extraction from SPICE-level netlists, transparent retargetability to different design styles and technologies, and the ability to correctly and holistically account for complex input waveform shapes, nonlinear and linear loading, multiple input switching, effects of internal state, multiple I/Os, supply droop and substrate interference. We validate ADME on a variety of digital gates, including multi-input NAND, NOR, XOR gates, a full adder, a multilevel cascade of gates and a sequential latch. Our results confirm excellent model accuracy at the detailed waveform level and testify to the promise of ADME for sustainable gate delay modelling at nanoscale technologies.

I. INTRODUCTION

Advanced CAD techniques for analog, mixed-signal and RF circuits have gained considerably in importance over the last decade, driven partly by the explosion in portable communications and the presence of analog/RF/MS circuitry in virtually every IC, SoC or SiP. In particular, model order reduction (MOR) techniques [1]–[6] enjoyed considerable development and adoption, originally in the context of interconnect reduction (*e.g.*, [7]–[9]) and more recently, in the more challenging domain of nonlinear circuits and systems (*e.g.*, [5], [6], [10]–[13]). MOR refers to the task of applying *algorithms* to reduce a large circuit or system to a much smaller one that behaves similarly from an input/output perspective. The small macromodels are used to replace the originals to achieve speedups in large system simulations, as well as for other advantages [14]. Recent techniques like TPWL [10], [11] and PWP [12], [13] constitute significant milestones in the quest for push-button macromodel generation techniques. These methods can extract general-purpose macromodels of broad classes of circuits with nonlinear and dynamic behavior (including op-amps, I/O buffers and the like) from their SPICE-level descriptions [13] – the extracted macromodels are capable of reproducing the gamut of continuous-time, nonlinear behaviors important in mixed-signal design. The power of techniques such as TPWL and PWP stems from the fact that they maintain accuracy at the lowest levels of the circuit – *i.e.*, at the level of individual node voltages and currents. As an illustration, Fig. 1 depicts clipping and slew-rate limiting correctly captured in a PWP-generated macromodel of a current-mirror op-amp [13].

At the same time, over the last decade, shrinking device dimensions and ever higher switching speeds have blurred the once-distinct boundary between purely digital circuits and mixed-signal/RF circuits. Analog and high-frequency effects now critical in digital gates include nonlinear resistive/capacitive loading, various interconnect (capacitive, inductive and transmission line) and crosstalk effects, imperfect intrinsic transistor performance (*e.g.*, lowered g_m and r_{ds} , Miller effects), dynamic IR drops, driver weakening, *etc.* As a result, traditional high-speed digital design methodologies – in particular, *gate delay modelling techniques for timing analysis and closure* – are under considerable stress today.

The problem of accurate delay modelling of gates has long been central to digital design. At low switching speeds, a simple but often adequate gate delay model is a constant “dead” delay, typically chosen to correspond to the 50% point of the waveform at the gate’s output. With increasing switching speeds, this model rapidly

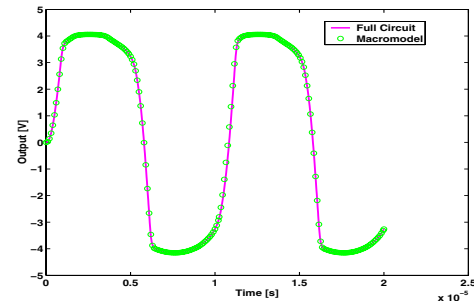


Fig. 1. Transient analysis of current-mirror op-amp with large sinusoidal input.

becomes inadequate for predicting the timing behavior of large collections of interconnected gates. Much attention has therefore been paid to the problem of devising better delay models – broadly termed *current-source models* [15] to reflect their typical structure (controlled current sources and other nonlinear resistive and capacitive elements). Single-pole, dominant delay models [16] were devised to take into account the increasing importance of interconnect delay. To better account for the strongly nonlinear switching inherent in digital gates, approaches such as [17], [18] parameterize delay using data from several points in the output waveform, the input slope and linear capacitive load. [19] incorporates nonlinear capacitors into gate delay models in order to capture important nonlinear dynamical effects. It has recently been recognized that multiple input switching (MIS) has important influences on delay [20], [21]. Other factors crucially affecting delay in modern high-speed digital gates include the effects of complex input waveform shapes, power/ground supply droop and nonlinear dynamical loading.

In spite of considerable R&D activity in gate delay modelling in recent years, current approaches continue to suffer from a variety of limitations. Virtually all nonlinear gate-delay models are based on relatively simple, manually-decided topological templates, with parameters extracted from I/O measurements and transient simulations. Often, the models are tailored to specific gate design styles and even to specific process technologies. Delay models are typically “extracted” manually, a slow and tedious process that is often ad-hoc and prone to error. “Second-order” device effects, which in today’s technologies are often critical, are difficult to abstract reliably in manual macromodelling methodologies. Indeed, the gate modelling approaches used today capture, at best, only aspects of the nonlinear DC I/O transfer function, together with (largely linear) capacitive/inductive effects at the gate periphery (*i.e.*, at the inputs and outputs). Such approaches cannot easily extend to incorporate effects of internal state and dynamics, hence are unable to adequately model, *e.g.*, delay due to significant internal capacitances, or phenomena such as persistent internal memory (as in latches and registers).

In this paper, we present a fundamentally new approach towards gate delay modelling: the use of automated nonlinear macromodelling methods, adapted from the MS/RF domain, to generate nonlinear gate delay macromodels suitable for timing analysis of digital

systems. We demonstrate that nonlinear MOR techniques are able to extract excellent timing macromodels of digital gates from their SPICE level descriptions at the push of a button. Our approach, termed ADME (Automated Delay Model Extraction) subsumes prior gate delay models as special cases and confers a variety of key advantages that circumvent or alleviate the problems mentioned above with current gate delay modelling techniques. Since ADME is based on algorithmic operations on underlying nonlinear differential-algebraic equation formulations, it is completely general in its applicability to any kind of gate or digital circuit, irrespective of design style, complexity, number of internal nodes, topology, process technologies used, *etc.*. The PWP-based algorithms underlying ADME fully incorporate effects of internal state and multiple inputs/outputs. As a result, MIS effects, delay effects of internal capacitors and nodes, and internal memory storage in latches are all well captured, as we show later in this paper. It is also possible to include power/ground and substrate nodes as additional “inputs” in order to capture the effects of power/ground droop (including dynamic droop) and substrate interference on delay. Since ADME starts by reading in a detailed SPICE-level circuit description, it fully incorporates second- and third-order effects modelled in advanced semiconductor device models such as BSIM and PSP. ADME also provides for a smooth tradeoff between macromodel accuracy and size/computation.

Simply the fact that ADME-based delay model generation is automated confers important advantages. By largely eliminating the need for manual intervention, ADME can reduce gate model development times from months to minutes or hours. This has significant implications for gate libraries being retargetted to new technologies or design rules, in terms of time to market and timeliness of chip tapeout. By eliminating the need for the gate modeller to have any specialized skills or knowledge regarding the internal operation or structure of the gate being modelled, ADME can alleviate manpower/staff bottlenecks in high-performance digital design. Mistakes and errors, inevitable during the complex process of manual gate modelling, can also be largely eliminated via a reliable automated process.

We validate and demonstrate ADME on a range of digital circuits and provide accuracy and performance comparisons of ADME-extracted gate models against full SPICE-level circuits. Starting with universal gates (NAND and NOR), we show how ADME correctly captures timing and MIS effects under a variety of input waveforms. We demonstrate ADME also on larger gates and blocks of gates (3-input XOR, a full adder and a multi-level cascade of gates) to exercise its ability to capture internal state and parasitic effects. Finally, we use ADME to extract a timing macromodel of a NAND/NOR based Reset-Set latch, verifying that it is capable of capturing the timing behavior of sequential circuits.

In our current implementation, ADME decreases circuit size by 4~6 \times and speeds up simulation by factors of 3~8 \times , depending on the type of circuit. Although we consider such gains to be relatively modest at this initial stage, due in part to the newness of the technique and a MATLAB-based implementation, we expect that further development and careful implementation can yield substantial gains. Even at this stage of ADME’s development, the excellent fidelity of the macromodel and the broad applicability, automated nature and short generation time of the extraction technique make it of compelling interest for many applications.

The remainder of the paper is organized as follows. In Section II, we provide a brief overview of relevant previous work in automated nonlinear macromodelling, including TPWL and PWP. In Section III, we present important implementation and usage details of ADME and illustrate its application using a 2-input XOR gate as an example. In Section IV, we apply ADME to a variety of combinatorial and sequential digital gates and provide accuracy/speedup comparisons.

II. OVERVIEW OF PWP AND ADME

In this section, we provide a brief overview of the PWP method, on which ADME is based, for obtaining general purpose macromodels for a nonlinear system, details may be found in [12], [13].

Piece-wise polynomial (PWP) based macromodelling combines the piece-wise idea of TPWL [10], with polynomial representations. Thus, it approximates each piece-wise region with higher order polynomials instead of purely linear models. Each polynomial model is based around one *special* state-space point of the system’s response/trajectory. These points, referred as expansion points can be selected either from full simulation or from DC sweeps corresponding to some *training input*. More information on training input and expansion points is provided in Section III.

A nonlinear system can be represented by the following vector differential algebraic equation (DAEs [1])

$$E\dot{x} = f(x) + Bu(t), \quad y(t) = Cx(t) \quad (1)$$

where, for an order n system, $x(t) \in \mathbb{R}^n$ is the state vector of internal node voltages and branch currents, $u(t) \in \mathbb{R}^m$ and $y(t) \in \mathbb{R}^p$ are m -inputs and p -outputs waveform to the circuit. $E \in \mathbb{R}^{n \times n}$ and $f(\cdot)$ represents the linear and nonlinear charge/flux and current terms respectively. The remaining matrices are $B \in \mathbb{R}^{n \times m}$ and $C \in \mathbb{R}^{p \times n}$.

Nonlinear macromodelling refers to reducing the order of this system to a much smaller size $q \ll n$ via a Krylov-subspace based projection basis $V \in \mathbb{R}^{n \times q}$ through the operation $x = Vz$, $z \in \mathbb{R}^q$ (e.g. [22], [5], [23]), such that

$$\hat{E} = V^T E V, \quad \hat{A} = V^T A V, \quad \hat{B} = V^T B, \quad \hat{C} = C V.$$

This leads to the following reduced model, while still providing good fidelity by matching the first q moments of the transfer functions of original and reduced system.

$$\hat{E}\dot{z} = \hat{A}z(t) + \hat{B}u(t), \quad y = \hat{C}z(t). \quad (2)$$

As mentioned earlier, the nonlinear system function $f(\cdot)$ is modelled using polynomial representations, around each s expansion points, $\{x_1, x_2, \dots, x_s\}$ as shown

$$E\dot{x}_i = f(x_i) + A_i^{(1)}x^{(1)} + A_i^{(2)}x^{(2)} + Bu(t), \quad y = Cx. \quad (3)$$

where $x^{(1)} = x - x_i$, $x^{(2)} = (x - x_i) \otimes (x - x_i)$, $A_i^{(1)}$ and $A_i^{(2)}$ are first and second order derivatives of $f(x_i)$. These separate piecewise regions are finally combined together:

$$\begin{aligned} \hat{E}\dot{z} &= \sum_{i=1}^m w_i(z)(\hat{f}(x_i) + \hat{A}_i^{(1)}z^{(1)} + \hat{A}_i^{(2)}z^{(2)} + \hat{B}_i u(t)), \\ y &= C[\sum_{i=1}^m w_i(z)(x_i + V(z - z_i))], \end{aligned} \quad (4)$$

to get macromodelled system response y in (4). Here V is obtained by Singular Value Decomposition (SVD) on individual V_i and $w_i(z)$ is weight function to smooth out transitions while merging different regions.

III. DELAY MODELS VIA ADME: AN ILLUSTRATED EXAMPLE WITH IMPLEMENTATION AND USAGE DETAILS

In this section, we illuminate some of the features as well as controlling parameters of ADME, by taking the example of a simple two-input exclusive-OR (XOR) gate as shown in Fig. 2

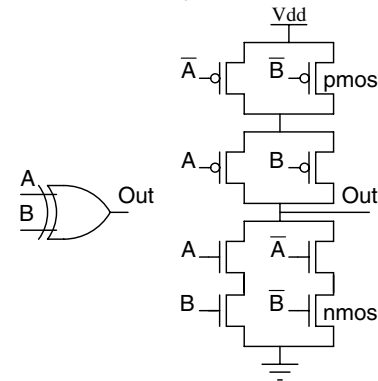


Fig. 2. 2-input XOR: Logic symbol and transistor level block diagram.

For all the examples in this paper, the gates/circuits are designed for 0.18micron static CMOS technology. All the MOS devices have been modelled using BSIM3 model. It should be noted that macromodels generated using ADME-based approach automatically abstract relevant features of all underlying device models in the circuit description, irrespective of its complexity. All circuit simulations and verifications represent apples-to-apples comparisons in MATLAB prototyping environment on a 2.4GHz, 256MB RAM Pentium-4 Linux (kernel x86-2.6.12) machine.

A. Training input and expansion points

To create a useful macromodel that can cater to a wide range of input variations, the system's frequently visited state-space should be covered by a *good* choice of training input. A good training input should vary fast enough to drive the system to upper bounds of state-space, besides capturing dynamic nonlinearities. For such training input, expansion points can then be selected along the trajectory. If one state x_i ensures that the relative error $err = \frac{|f(x_i) - f_{linear}(x_i)|}{|f_{linear}(x_i)|} > \alpha$, the relative error tolerance, then it can be added to the expansion points list. Here $f_{linear}(x)$ refers to the linearized model of $f(x)$. As obvious, α , a user controlled parameter, affects the number of expansion points generated for one particular trajectory and thus governs the tradeoffs between accuracy and speedup of macromodel generated.

For the XOR gate example, we have varied α in the range of $5E-3 \sim 5E-2$. The original circuit order, $N = 36$ for XOR gate is reduced to $q = 10$ using ADME-based MOR. To provide a decent training input, we vary both inputs to XOR gate. The input-output characteristics for one set of training input is as shown in Fig. 3

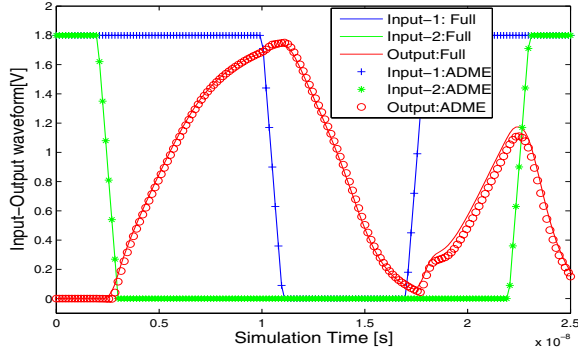


Fig. 3. System response of training input for 2-input XOR.

The simulation time for full simulation was recorded as 81.4s, while for macromodelled XOR gate it was 42.1s, about $2\times$ speedup in simulation time. Next in Fig. 4, we show the plot for one different set of inputs to the circuit, but the macromodel is not re-generated and thus it uses the *same* previous training input based macromodel. It can be seen that good accuracy, compared to full simulation, is achieved, besides saving macromodel regeneration time.

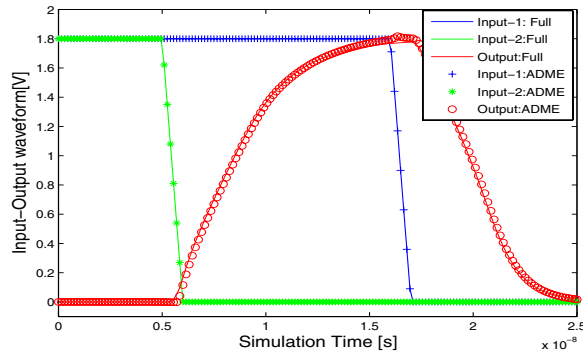
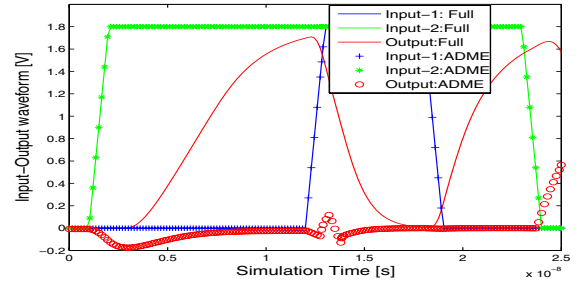


Fig. 4. System response using previous macromodel.

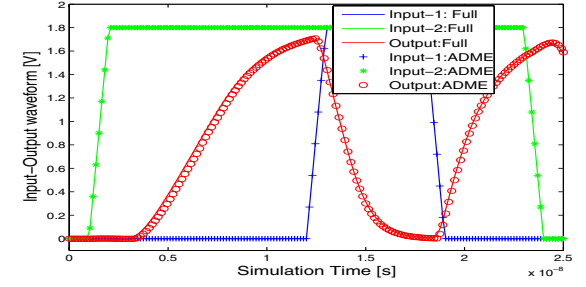
B. Merging of trajectory

Yet, for another set of inputs, the trajectory covered may be quite different from the case considered above, though they may have similar looking waveforms at select output nodes. Fig. 5-(a) shows the effect of such scenario. Thus to improve the state-space coverage, and thus get broadly applicable macromodel, the trajectories for different sets of training input can be merged. Fig. 5-(b) shows the response after merging.

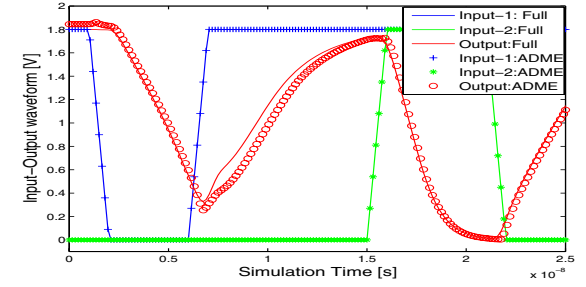
Merging may lead to large number of regions. This redundancy is minimized in ADME-based macromodelling by examining the similarities (using norm distance among expansion points) among regions. For this example, the total expansion points, and hence the



(a) Trajectory mismatch



(b) System response with merged trajectory



(c) Merged trajectory: different input variation

Fig. 5. Merging multiple trajectories: better state-space coverage.

number of regions grew to 45 from initial individual values of 36 and 26, still getting a speedup of $1.5\times$. The usefulness of merged macromodel is further illustrated in Fig. 5-(c) for another set of inputs.

C. Optimal order size

The optimal reduced order of the system can be predicted by using SVD of projection bases of individual piece-wise regions. For optimal model size this value shows a sudden drop, corresponding to a common minimum subspace, indicating minimum redundancy. From Fig. 6-(a), it is clear that the minimum order size of XOR gate for this set of input combinations is $q = 10$, and forcing an order less than that, say 8, can lead to mismatch of generated macromodel, as shown in Fig. 6-(b). The model does not converge for values of q less than 8.

IV. APPLICATION AND VALIDATION OF ADME

In this section, we perform in-depth evaluations of ADME-based macromodels for digital circuits. It involves, for each circuit, generation of macromodels, transient based simulations and finally comparison with full simulation for validation. Also discussed are some important timing metrics for digital gates. Model generation and speedup numbers are also provided. Examples consist of variety of multi-input as well as cascaded combinatorial and sequential digital circuits, so as to demonstrate the successful capture of internal nonlinearities and loading effects by ADME-based macromodels.

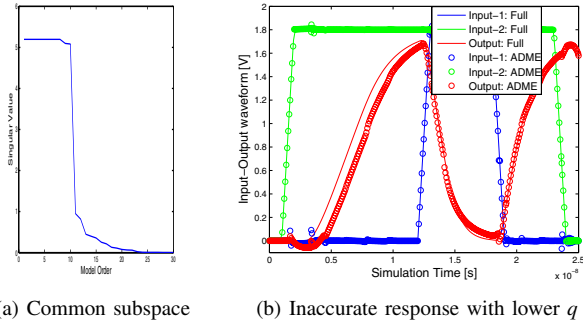


Fig. 6. Detecting optimal model order.

A. Multi-input combinatorial gates

We consider 2-input NAND gate, 2-input NOR gate, 3-input XOR gate and 1-bit full adder as examples of basic digital gates/circuits. It should be noted that as the number of inputs to the gate increases, so is the probability of having many input combinations of being randomly high or low. Besides, more internal node capacitances come into picture, making the problem of finding a good gate level model more challenging, even by using recent techniques like current-source modelling.

1) 2-Input NAND: We consider the example of 2-input NAND gate. It has been sized with aspect ratio (W/L) of 3 and 6 for nmos and pmos respectively, for 0.18micron CMOS technology.

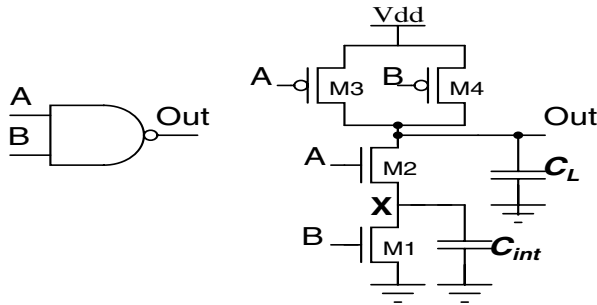


Fig. 7. Transistor level circuit diagram of 2-input NAND gate.

To consider the multi-input effects as well as the effect of internal node (node X in Fig. 7) capacitance on the intrinsic propagation delay of NAND gate, we simulate the low-to-high delay for different input patterns. Here, internal node capacitance consists of junction capacitance of transistors M1 and M2, as well as gate-source and gate-drain capacitances. The worst case delay happens when the internal node point X is initially charged up to $V_{DD} - V_{Tn}$, which can be ensured by making input A transition from $1 \rightarrow 0 \rightarrow 1$, and input B a $0 \rightarrow 1$ transition, as shown by waveform case(b) in Fig. 8

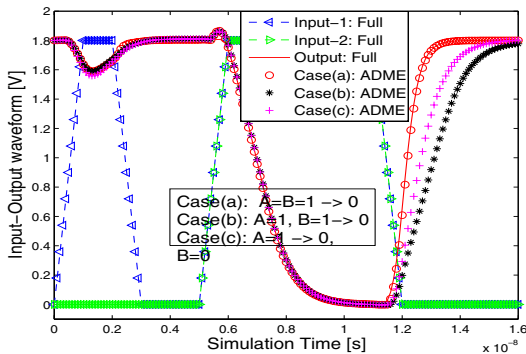


Fig. 8. Transient system response highlighting the worst case delay for 2-input NAND gate.

For multi-input digital gates, important performance matrices like delay calculations as well as noise margins depend to a large

extent upon the data input patterns because of presence of internal linear/nonlinear node capacitance and body-bias effect. Thus, though simultaneous transition of inputs may have higher noise margin, but it may not represent the worst case delay as evident from Fig. 8 where it is clearly visible that worst case delay occurs when input A is high and B makes a high to low transition. The low-to-high delays for case(a), case(b) and case(c) are $0.59ns$, $1.5ns$ and $1.1ns$ respectively. Thus these data-dependencies should be carefully modelled in most of the gate models. ADME-based macromodels do not have to worry about such input patterns dependent restrictions.

The simulation time for full simulation was recorded as $28.7s$, while for MOR based NAND gate it was $16.6s$, which is roughly a $1.7\times$ speedup in simulation time. We are unable to present similar data for 2-input NOR gate because of space limitations.

2) 3-Input XOR Gate: To further illustrate that ADME-based macromodels are useful for multi-input gates, we apply it to a 3-input XOR gate. The large number of possible combinations of various input transitions and waveforms for three inputs is especially laborious to deal with in manual methodologies for timing model generation. Although the combinatorial explosion of input waveforms is also an issue for ADME-based macromodels, the automated nature of the generation process makes it a far more tractable, sustainable and efficient alternative.

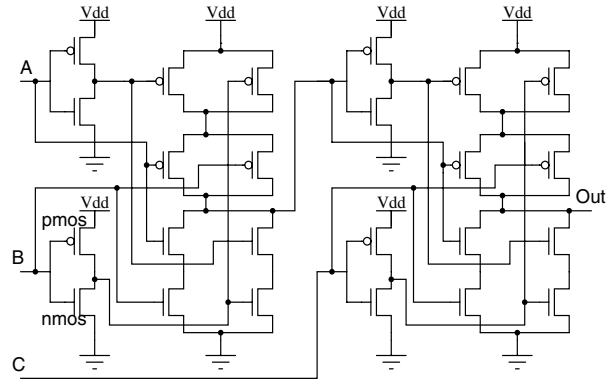


Fig. 9. Transistor level circuit diagram of 3 input XOR gate (24 MOSFETS)

To consider the multi-input effect as well as the effect of internal and load capacitance, we simulate the 3-input XOR gate, consisting of 24 MOSFETS. To obtain an effective macromodel as well as to consider the optimum delay of the circuit, we provide the input waveform as shown in Fig. 10. Two separate cases are considered, one in which we consider intrinsic propagation delay, with no load capacitance, and the other case with $1pF$ of load capacitance.

The output waveforms for ADME-based macromodel exactly match the full simulation values. Further, we also plot the waveform at one internal node in this 3-input XOR gate, as shown by star format waveform in Fig. 10. It corresponds to the node corresponding to 2-input XOR output for inputs A and B. As expected, the propagation delay of the 3-input XOR gate is more than that for 2-input NAND gate considered earlier. This is further confirmed here by the simulation results that the propagation delay of CMOS gate deteriorates rapidly as a function of the fan-in of the gate.

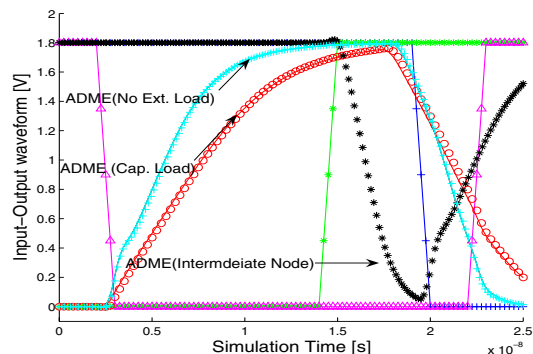


Fig. 10. Transient system response of 3-input XOR gate.

As mentioned above and as expected in this case of 3-input XOR

gate, a better speedup of $4.2\times$ is achieved, as the full simulation time is $168.7s$ while the reduced order macromodel simulation gets completed in just $39.5s$ for a reduced order of $q = 24$. Note that as the order of the system is increased, so is the likelihood of greater redundancy in the state space, leading to the possibility of better optimized timing models generated via ADME-based macromodels. The speedup figure for the case of capacitive load of $1pF$ is a slightly higher value of $5.5\times$, which can be attributed to the generation of lesser number of expansion points due to more redundancy in circuit. Nevertheless it proves that ADME-based macromodels capture output load, as well as internal node capacitance behaviors effectively.

3) *1-bit full adder*: Full adders are an integral component of microprocessors, and of arithmetic logic units (ALUs) in particular. The block diagram of a 1-bit full adder is shown in Fig. 11, designed using static CMOS based NAND and XOR gates, involving a total of 42 MOS transistors.

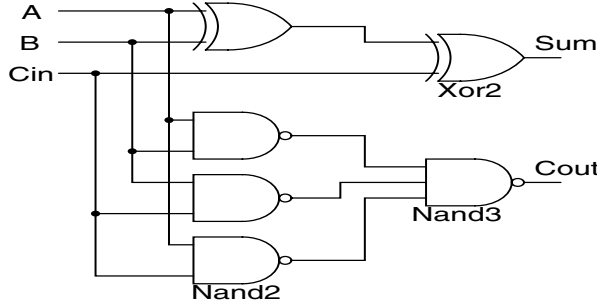


Fig. 11. Gate level circuit diagram of 1-bit full adder circuit.

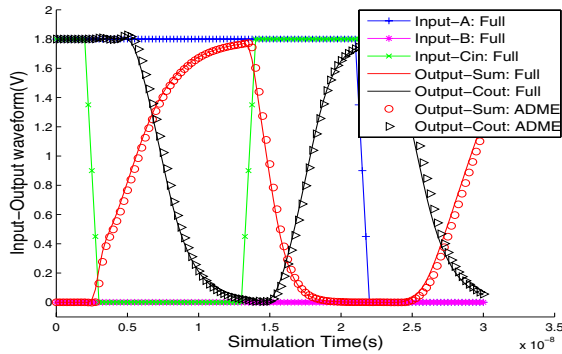


Fig. 12. System response of 1-bit full adder circuit: ($N=121$, $q=28$), speedup= $6.7\times$

Since the full-adder presented here consists of NAND and XOR gates, which we have already analyzed, we choose this example to show how effectively ADME-based macromodels behave when the individual components are brought together to form a system. We have not used these individual macromodels as drop-in replacements for this example, and we are going to incorporate it in our future work. In Fig. 12, the *circle* points plot represents the ADME-based macromodelled output for *Sum* bit, and the *triangle* waveform shows the corresponding values of *Cout*. It is apparent from the figure that the macromodelled *Sum* and *Cout* outputs are following the full simulation waveforms very accurately.

Using the ADME-based macromodel resulted in a speedup of about $6.7\times$, as the reduced order transient simulations took just $32.8s$ compared to about $219.2s$ for full transient simulations. The size of original system for 1-bit full adder, without considering any external load, was reduced from 113 to 28 for the macromodel. It took just $15 \sim 20s$ to generate the macromodel for 1-bit full adder, with the number of expansion points generated to be 21.

B. Multi-level cascade of gates (many internal nodes/capacitors)

As the number of inputs to the circuits increases, the internal nonlinearities as well as the inputs combinations and transitions occurring anytime, increases manifold. Also the nonlinear effects of one stage may get passed on, with increasing undesirable effects, to stages towards the load. To consider all this effects, we illustrate and

apply ADME-based macromodel generation on the circuit shown in Fig. 13, which consists of 2-input NAND, NOR and XOR gates in a multilevel cascade. The last two stages of cascade consists of two inverters driving a capacitive load of $C_L = 5pF$. Since the chain of gates has been sized for minimum propagation delay using logical effort [24] approach, no external capacitance has been applied at the last two intermediate nodes.

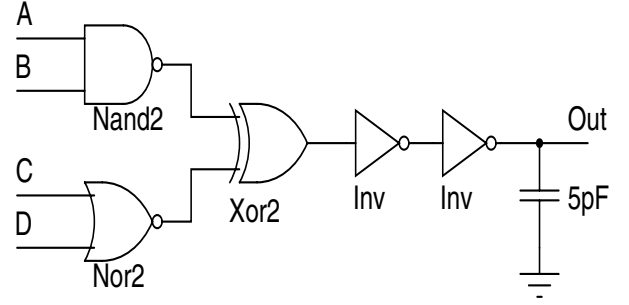


Fig. 13. Gate level block diagram of multilevel gate chain.

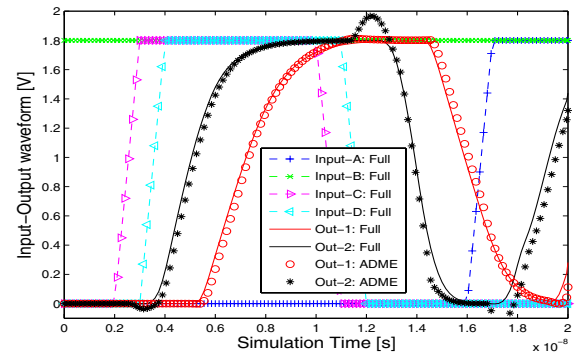


Fig. 14. System transient response of cascaded ckt: ($N=70$, $q=22$), speedup= $5\times$

For this particular input, full transient simulation takes $143.8s$, while the ADME-based macromodel takes $28.2s$, a speedup of approximately $5\times$. The XOR gate output waveform, which is an internal node, has also been captured and plotted in Fig. 14. It is represented in *star* format waveform and it verifies that the output of 2-input XOR gate is high when one of the inputs is high and other low. The results displayed in Fig. 14 verifies the accuracy obtained by using ADME-based macromodels.

C. Sequential circuits

As a final validation, we consider NAND and NOR based set-reset (SR) latch, which are considered as basic building blocks for sequential digital circuits. They also have been designed using static CMOS technology, for $0.18\mu m$ process technology.

Coming up with a gate model that can preserve all the internal *memory* state of a sequential circuit is a challenging task. Issues such as crosstalks, clock skew may get propagated along the digital system. Further, input patterns may force the system to a state where the output can not be determined analytically, as the case when both inputs to the SR latch gets high level.

Observing the output waveforms in Fig. 15-(a) and Fig. 15-(b), we can verify that the ADME-based macromodels accurately capture the internal states of the sequential system, thus showing same output as full simulations output. For the NAND based latch, the *Set* input signal, shown as *plus* waveform is governing the next state of output Q , shown in *circle* waveform, while the state of output Q' shown in *triangle* format plot is being dictated by the *Reset* input signal in *star* format waveform. Also observe that the macromodelled output is matching the full simulation output even when the system is in a unknown state, when both *Set* and *Reset* are high and both Q and Q' are high. Similarly for NOR based latch, when both inputs are high, the state of the system is unknown, visible by same values for Q and Q' .

For the NOR based SR latch, the full transient simulation took about $53.6s$, while the ADME-based macromodel consumed about

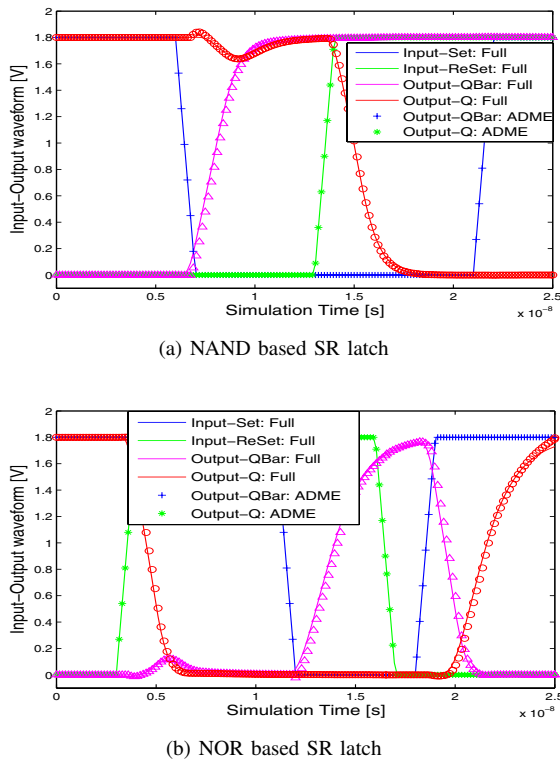


Fig. 15. System transient response of NOR and NAND based SR latch: (N=26, q=8), speedup=3x for NOR based latch, speedup=2x for NAND based latch

18.3s, getting a $3\times$ speedup. For the NAND based SR latch ADME-based macromodel simulations took 20.9s against 46s for full simulation, making a speedup of $2.2\times$. The macromodel generation time was about 10s.

V. CONCLUSIONS

We have presented ADME, a novel automated approach towards generating gate delay models, and demonstrated its many advantages over the prior state of the art in gate delay modelling. We emphasize that this paper represents only a first exploration of the many dimensions and possibilities of this approach. Even so, the results presented, and the facts that ADME is automated, broadly applicable and based on solid algorithmic underpinnings, testify to its promise as a sustainable technique for cell/gate delay and timing model generation for current and future high-performance nanoscale designs.

REFERENCES

- [1] J. Roychowdhury. Reduced-order modelling of time-varying systems. *IEEE Trans. Ckts. Syst. - II: Sig. Proc.*, 46(10), November 1999.
- [2] A. Odabasioglu, M. Celik, and L.T. Pileggi. PRIMA: passive reduced-order interconnect macromodelling algorithm. In *Proc. ICCAD*, pages 58–65, November 1997.
- [3] L.T. Pillege and R.A. Rohrer. Asymptotic waveform evaluation for timing analysis. *IEEE Trans. CAD*, 9:352–366, April 1990.
- [4] J. Phillips. Model Reduction of Time-Varying Linear Systems Using Approximate Multipoint Krylov-Subspace Projectors. In *Proc. ICCAD*, November 1998.
- [5] J. Phillips. Projection frameworks for model reduction of weakly nonlinear systems. In *Proc. IEEE DAC*, June 2000.
- [6] J. Roychowdhury. Reduced-order modelling of time-varying systems. *IEEE Trans. Ckts. Syst. - II: Sig. Proc.*, 46(10), November 1999.
- [7] Tak K. Tang and Michel S. Nakhla. Analysis of High-Speed VLSI Interconnects Using the Asymptotic Waveform Evaluation Technique. In *Proc. ICCAD*, pages 542–545, 1990.
- [8] N. Gopal, D.P. Neikirk, and L.T. Pillege. Evaluating RC-Interconnect Using Moment-Matching Approximations. In *Proc. ICCAD*, pages 74–77, November 1991.
- [9] A. Odabasioglu, M. Celik, and L.T. Pileggi. PRIMA: passive reduced-order interconnect macromodelling algorithm. *IEEE Trans. CAD*, pages 645–654, August 1998.
- [10] M. Rewinski and J. White. A Trajectory Piecewise-Linear Approach to Model Order Reduction and Fast Simulation of Nonlinear Circuits and Micromachined Devices. In *Proc. ICCAD*, November 2001.
- [11] M. Rewinski and J. White. A trajectory piecewise-linear approach to model order reduction and fast simulation of nonlinear circuits and micromachined devices. *IEEE Trans. CAD*, pages 155–170, February 2003.
- [12] N. Dong and J. Roychowdhury. Piecewise Polynomial Model Order Reduction. In *Proc. IEEE DAC*, pages 484–489, June 2003.
- [13] Ning Dong and Jaijeet Roychowdhury. Automated Extraction of Broadly Applicable Nonlinear Analog Macromodels from SPICE-level Descriptions. *Proc. IEEE CICC*, 2004.
- [14] J. Roychowdhury. Reduced-order modeling of time-varying systems. *IEEE Trans. on Circuits and Systems II: Analog and Digital Signal Processing*, 46(10):1273–1288, October 1999.
- [15] Inc. Synopsys. Composite Current Source Timing White Paper, 2005.
- [16] F. Dartu, N. Menezes and L.T. Pileggi. Performance computation for precharacterized CMOS gates with RC loads. *IEEE Trans. CAD*, 15(5):544–553, May 1996.
- [17] J. Croix and D. Wong. Blade and Razor: Cell and interconnect delay analysis using current-based models. In *Proc. of IEEE DAC*, pages 386–389, June 2003.
- [18] B. Tutuianu. Nonlinear driver models for timing and noise analysis. *IEEE Trans. CAD*, 23(11):1510–1521, November 2004.
- [19] P. Li and E. Acar. Waveform-independent gate models for accurate timing analysis. In *Proc. ICCD*, pages 363–365, October 2005.
- [20] V. Chandramouli and K. Sakallah. Modeling the effects of temporal proximity of input transitions on gate propagation delay and transition time. In *Proc. of IEEE DAC*, pages 617–622, June 1996.
- [21] C. Kashyap. Personal Communications, 2006.
- [22] P. Li and L. Pileggi. NORM: Compact Model Order Reduction of Weakly Nonlinear Systems. In *Proc. IEEE DAC*, pages 472–477, June 2003.
- [23] J. Roychowdhury. Reduced-order modelling of linear time-varying systems. In *Proc. ICCAD*, November 1998.
- [24] Ivan Sutherland, Robert F. Sproull, and David Harris. *Logical Effort-Designing Fast CMOS Circuits*. Morgan Kaufmann, 1999.