Automated, Accurate Macromodelling of Digital Aggressors for Power/Ground/Substrate Noise Prediction

Zhe Wang
zhewang@ece.umn.edu
Department of ECE
University of Minnesota

Rajeev Murgai
murgai@fla.fujitsu.com
Fujitsu Laboratories of America

Jaijeet Roychowdhury
jr@ece.umn.edu
Department of ECE
University of Minnesota

Abstract

Noise analysis and power distribution network reliability assessment is extremely important in deep sub-micron digital and mixed-signal circuit design. Both relate closely to the nonlinear loading impact of digital circuits. Consequently, accurate estimation of the latter is critical. In this paper, we present extraction techniques that automatically generate a family of small, time-varying macromodels for digital cell libraries, at the time of their library characterization. Our approach is based on importing and adapting the Time-Varying Padé (TVP) method, for linear time-varying (LTV) model reduction, from the mixed-signal macromodelling domain. Our approach features naturally higher accuracy than previous ones, and in addition, offers the user a tradeoff between accuracy and macromodel complexity. A key attraction of our approach is that it can be merged into cell library extraction methodologies to produce accurate-by-construction noise models for digital blocks. Simulations and comparisons confirming the efficacy of our approach are provided.

I. Introduction

In digital and mixed-signal circuit design, noise analysis and power distribution network reliability assessment play increasingly important roles. In the future, two technology trends will make them even more important. First, there is an increasing trend for system-on-a-chip (SoC) designs for low system cost, reduced area and low power consumption. Such designs give rise to *signal integrity problems* between noisy digital circuits and sensitive analog sections. A major noise source is simultaneous switching noise (SSN) or delta-I noise, due to the inductance effects $L\ dI/dt$ [2]. With millions of transistors on a single chip, the SSN can easily reach hundreds of millivolts. Secondly, as digital technology advances, the average current entering the chip increases, and the supply voltage levels go down [1]. As a result, the inductance effects and the resistance of power supply network introduce potentially large voltage drop, which can have severe impact on the circuit performance [11], because the reduced supply voltage gives rise to limited noise margin.

To assess and avoid such problems, noise analysis and power distribution network reliability assessment need to be conducted. It is evident that both are closely related to IR drop and $L\ dI/dt$. Consequently, accurate estimation of current drawn by digital circuits consists of a critical component. As the power distribution relies on the currents drawn, the current drawn also relies on the power supply voltage. Therefore, accurate current estimation can only be achieved if the whole circuit is taken into account, as illustrated in Fig. 1.

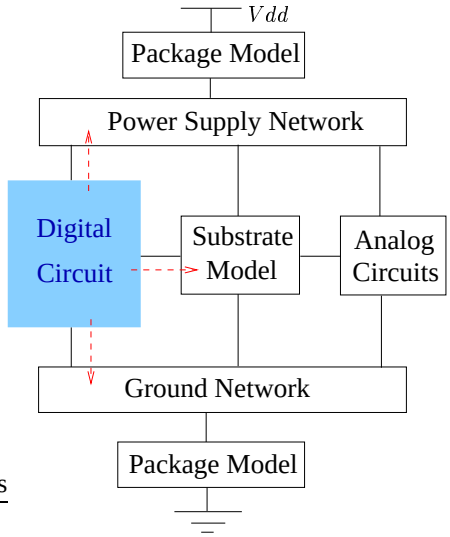

Fig. 1. The switch activities in digital circuits continuously introduce noise (shown as dashed line) to the power supply network, ground network and substrate. Through these connections, noise impacts the performance of other circuits on the same chip and even digital circuits themselves.

In principle, one could use the full SPICE model, but this is impractical because the number of transistors in modern digital circuits can easily reach tens or even hundreds of millions. Besides, the switching behavior of transistors is nonlinear in nature. But this way would be the most accurate to solve the problem. Therefore, people have tried to come up with much faster/smaller/simpler models which are still reasonably accurate, but are orders of magnitude faster. Here, we provide a brief review of the previous relevant approaches. In [12], Shepard and Kim presented a state diagram for estimating the body-potential of partially-depleted silicon-on-insulator (PD-SOI) FETs. To estimate the current injected to the substrate, Heijningen et al developed a macromodel that consists of a current source acting as a noise source, which is obtained by running the SPICE model for each gate [13]. Chen and Ling, in [4], created a model for each individual digital block. The model utilizes capacitors controlled by ideal switches to imitate the switching behavior of digital cells. The model developed by Dharchoudhury et al employs independent time varying current source that is stored in a cell library for each cell

1530-1591/04 $20.00 (c) 2004 IEEE

[5].

By replacing the SPICE model of digital blocks with their corresponding macromodels, all of them considerably speed up the computation. However, modeling MOSFETs as capacitors may result in loss of the device properties; and modeling nonlinear devices as independent time varying current sources without considering the interaction between the currents and power supply voltage introduces errors. It is worth mentioning that all the models are derived manually from SPICE model and their accuracy highly depends on the researchers' deep understanding of the physical nature of the digital circuits. To our best knowledge, to date, no-one appears to have looked into methods that automatically generate simpler models for substrate noise analysis and power supply network evaluation, starting from the SPICE-level circuit descriptions.

In this paper, we propose and demonstrate a fundamentally new approach towards creating good simple models for estimating *SSN and IR drop* in power supply network. The key idea is using algorithms to *automatically* generate small simplified models from the SPICE-level descriptions of digital cells. Using this method, each digital cell containing one or more nonlinear devices is converted into a LTV macromodel, which can be extracted off-line, stored in a cell library, and incorporated into a chip-level power distribution network representation. The advantages of such an approach include: i) automatic (as opposed to manual) generation. Cell designers have the SPICE circuit anyway, which is used for characterization. Simple models can be automatically generated as a part of the the characterization phase. ii) accuracy. Our macromodel extraction automatically takes into account, and is algorithmically based on, the important second-order effects, directly from the SPICE MOSFET model level (e.g., BSIM). Therefore, the accuracy does not depend on the knowledge/understanding of the cell designer, as opposed to existing manually generated macromodels. iii) opportunity to trade-off model complexity for accuracy. This can be achieved by adjusting the prescribed order of the macromodel and the time step of the transient analysis. and iv) drop-in replacement. Each digital cell can be represented by its corresponding macromodel generated off-line and stored in the cell library. However, the proposed macromodel is input pattern dependent, just the same as the model in [5]. For our noise analysis purpose, we prefer to store a macromodel with a pattern that can drive the cell to draw the largest current.

Capturing the switching nonlinearities by the time-varying Padé (TVP) method developed in [8], our macromodel also employs order reduction techniques. As a result, the chip-level power distribution network representation employing our macromodels is considerably less computation- and time-demanding, in comparison with that based on the SPICE-level models. Moreover, different from the existing macromodel approaches, our method allows for the integration of individual macromodels with imperfect power supply and/or ground bounce, and is thus able to capture the current change induced thereof. The simulation results confirm that: compared to SPICE model, our macromodel approach is up to *580* times as fast; compared [5], ours is up to *45* times as accurate. In a nutshell, our macromodel extraction requires small changes to cell characterization methodology, but enables generation of bottom-up accurate-by-construction models for power supply and substrate noise analysis.

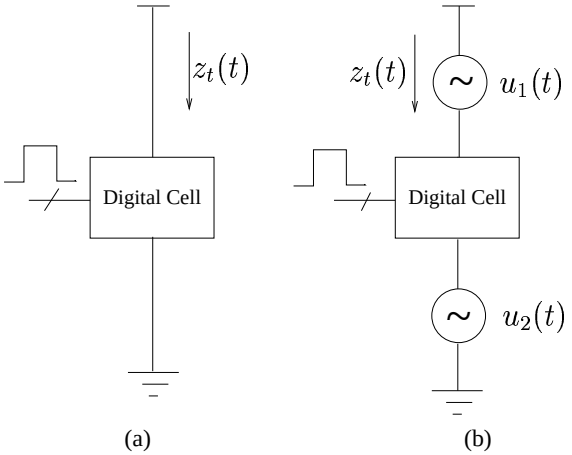

Fig. 2. (a) Simplified system-level circuit with perfect power supply and ground for macromodel extraction in [5], see also (1); (b) Realistic system-level circuit with noisy power supply and ground, see also (2).

The rest of the paper is organized as follows. In Section II, the macromodel will be developed. In this section, we will also give a brief review of TVP. Macromodel examples, together with simulations and comparisons, will be presented in Section III. Finally, concluding remarks will be given in Section IV.

II. Macromodel for Current Estimation

As mentioned before, two macromodels are prevalent in estimating currents drawn by the digital cells [4, 5]. To imitate the switching behavior of digital cells, the model established in [4] uses capacitors that are controlled by ideal switches; whereas the one in [5] uses time varying current sources. By replacing the digital cells with their corresponding macromodels, both [4] and [5] considerably speed up the computation. It is evident that the former ignores the switching behavior of digital circuits; and the latter is a solution of

$$\frac{d\boldsymbol{q}(\boldsymbol{y}(t))}{dt} + \boldsymbol{f}(\boldsymbol{y}(t)) = \boldsymbol{b}_l(t), \;\; z_t(t) = \boldsymbol{d}^T \boldsymbol{y}(t), \qquad (1)$$

over a time period (usually a clock cycle), for each digital cell connecting to the perfect ground and power supply, as shown in Fig. 2(a). In (1), $\boldsymbol{b}_l(t)$ is the logic inputs of the digital cell; $\boldsymbol{y}(t)$ is a $m \times 1$ vector containing a total of m unknown node voltages and branch currents; $\boldsymbol{q}(\cdot)$ and $\boldsymbol{f}(\cdot)$ are nonlinear functions describing the charge/flux and resistive terms in the cell, respectively; the system output $z_t(t)$ is the current drawn; and $\boldsymbol{d}$ is the vector that link the output to the rest of the system. Notice that power supply (and/or ground) voltage variation is not included in (1), which implies that no interaction between the device current and the supply voltage is captured. In the past, the error incurred by ignoring such interaction was thought small when voltage drop is no more than 10% of the ideal power supply voltage [5]. However, our experiment shows that when the power supply voltage drops 10% from (the ideal value) $2V$ in a 0.25μm technology, the peak current through an

inverter changes more than 30%. Unfortunately, such an error is inherited to any power noise analysis methods utilizing the model as in [5]. The demand for accurate current estimation calls for a macromodel with satisfactory accuracy and affordable complexity. To derive such a macromodel that captures also the nonlinear switching behavior, let us start with system equations that describe more accurately the digital cells.

As depicted in Fig. 2(b), power supply noise and ground bounce can be modeled as *small* system inputs, in addition to the *large* signal vector $\boldsymbol{b}_l(t)$. Gathering the small inputs into a vector $\boldsymbol{u}(t)$, a nonlinear system driven by both $\boldsymbol{b}_l(t)$ and $\boldsymbol{u}(t)$ can be obtained as follows:

$$\frac{d\boldsymbol{q}(\boldsymbol{y}(t))}{dt}+\boldsymbol{f}(\boldsymbol{y}(t))=\boldsymbol{b}_l(t)+\boldsymbol{B}\boldsymbol{u}(t),\ z_t(t)=\boldsymbol{d}^T\boldsymbol{y}(t), \quad (2)$$

where $\boldsymbol{B}$ is the matrix that links the small-signal inputs to the rest of the system. Notice that different from (1), now the system output $z_t(t)$ is the current drawn induced not only by the logical input of the digital cell $\boldsymbol{b}_l(t)$, but also by the power supply noise and ground bounce captured by $\boldsymbol{u}(t)$. For simplicity, we will take $\boldsymbol{u}(t)$ and $z_t(t)$ to be scalars and consider only the power supply noise hereafter. With $\boldsymbol{u}(t)=u(t)$ being a scalar, $\boldsymbol{B}$ becomes a vector $\boldsymbol{b}$.

Evidently, solving (2) directly can yield the accurate current $z_t(t)$, provided that $\boldsymbol{u}(t)$ is known. Notice that $\boldsymbol{u}(t)$ relies on all digital cells that are connected to the same power supply distribution network. As a result, $z_t(t)$ for each digital cell can only be obtained by solving (2) for all cells, and treating $\boldsymbol{u}(t)$ as nuisance unknown. In a realistic setup, however, this approach is not only time consuming, as will be shown by simulations in Section III, but also computational demanding. The latter renders direct solution of (2) not applicable to large chip scales, whereas it is well known that modern chip design can easily reach tens, or hundreds, of millions transistors.

To this end, we will develop a macromodel based on TVP method. As shown in Fig. 3, the resultant LTV macromodel consists of two major components: a current source $I_0(t)$, and a ODE system generating the current $\Delta I(t)$. The former ($I_0(t)$) is the current that the digital cell consistently draws, assuming perfect power supply and ground. It is clear that $I_0(t)$ is independent of the voltage variation(s), and can be obtained by solving (1), with $z_t(t)$ replaced by $I_0(t)$. The ODE system in our macromodel turns out to be LTV, which acts as a current source and the current $\Delta I(t)$ is determined by the voltages at node 2 and node 3, i.e., voltage variations at ground and power supply, respectively. To reach a model that is applicable to large chip scales, we will also apply model reduction techniques. As a result, the LTV ODE system contains only $1 \sim 10$ nodes, which corresponds to a marked reduction in comparison with hundreds of nodes in the original digital cell. Summing up $I_0(t)$ and $\Delta I(t)$ gives rise to the total current $I(t)=I_0(t)+\Delta I(t)$ drawn by the digital cell from the noisy power supply and grounds.

Starting from a brief review of TVP, we will next present in detail the establishment of the macromodel from (2).

A. TVP

In [8], a general method called TVP for deriving small macromodel from SPICE-level circuit descriptions was proposed. Main application of TVP was mixed-signal/RF/analog circuits and it was used there for mixers and switched-capacitor filters. However, if appropriately adapted, TVP is highly applicable to analysis and optimization of power supply network and substrate noise analysis, i.e., reduction of large digital logic blocks for *SSN and IR drop prediction purposes*, as we develop and demonstrate here. In this section, we provide a brief review of TVP.

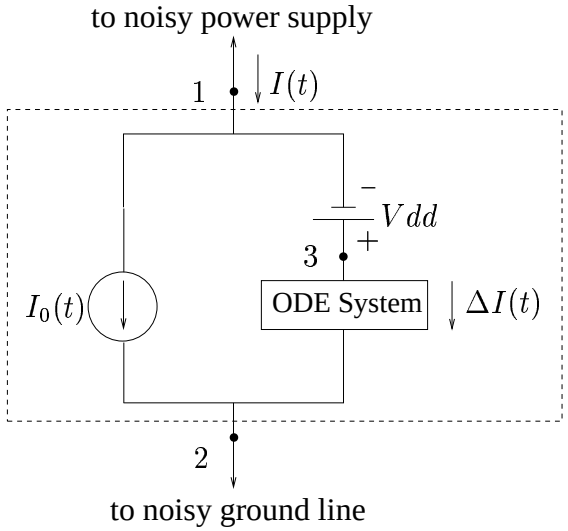

Fig. 3. Proposed macromodel of a digital cell.

Separating the time scales of the small input $u(t)$ and the logic input $\boldsymbol{b}_l(t)$, (2) can be re-expressed in MPDE form as:

$$\begin{aligned}\frac{\partial\boldsymbol{q}(\hat{\boldsymbol{y}})}{\partial t_1}+\frac{\partial\boldsymbol{q}(\hat{\boldsymbol{y}})}{\partial t_2}+\boldsymbol{f}(\hat{\boldsymbol{y}}(t_1,t_2)&=\boldsymbol{b}_l(t_1)+\boldsymbol{b}u(t_2)\\ \hat{z}_t(t_1,t_2)&=\boldsymbol{d}^T\hat{\boldsymbol{y}}(t_1,t_2)\\ z_t(t)&=\hat{z}_t(t,t),\end{aligned} \quad (3)$$

where the hatted variables are bivariate (i.e., two-time scales) forms of the corresponding variables in (2). In fact, it has been proved in, e.g., [9], that any solution of (2) generates a solution of (3).

In order to obtain the output linear in the input $u(t)$, we first solve (3) when $u(t_2)=0$. This step is equivalent to solving (1). With the solution denoted by $\hat{\boldsymbol{y}}^*(t_1)$, the output is given by $z_t(t)=\hat{z}_t(t,t)=\boldsymbol{d}^T\hat{\boldsymbol{y}}^*(t)$. Notice that $z_t(t)$ is nothing but $I_0(t)$ in Fig. 3. Linearizing (3) around $\hat{\boldsymbol{y}}^*(t_1)$, a linear MPDE form can be obtained by :

$$\begin{aligned}\frac{\partial(\boldsymbol{C}(t_1)\hat{\boldsymbol{x}})}{\partial t_1}+\frac{\partial(\boldsymbol{C}(t_1)\hat{\boldsymbol{x}})}{\partial t_2}+\boldsymbol{G}(t_1)\hat{\boldsymbol{x}}&=\boldsymbol{b}u(t_2)\\ \hat{z}(t_1,t_2)&=\boldsymbol{d}^T\hat{\boldsymbol{x}}(t_1,t_2)\\ z(t)&=\hat{z}(t,t),\end{aligned} \quad (4)$$

where vectors $\hat{\boldsymbol{x}}$, $\hat{z}$, and z are the small-signal versions of $\hat{\boldsymbol{y}}$, $\hat{z}_t$ and z_t, respectively; $\boldsymbol{C}(t_1)=(\partial\boldsymbol{q}(\hat{\boldsymbol{y}})/\partial\hat{\boldsymbol{y}})|_{\hat{\boldsymbol{y}}^*(t_1)}$ and $\boldsymbol{G}(t_1)=(\partial\boldsymbol{f}(\hat{\boldsymbol{y}})/\partial\hat{\boldsymbol{y}})|_{\hat{\boldsymbol{y}}^*(t_1)}$ are time-varying matrices. Eq. (4) implies that the bivariate output $\hat{z}(t_1,t_2)$ is linear in the small input signal $u(t_2)$, but the linear relationship is varying

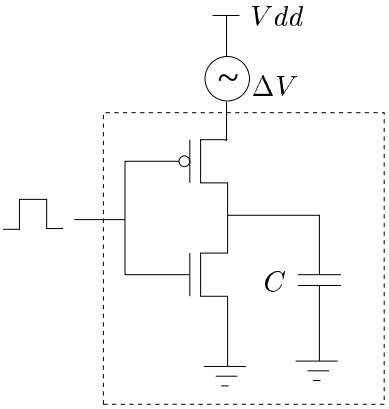

Fig. 4. A single inverter connected to a noisy power supply.

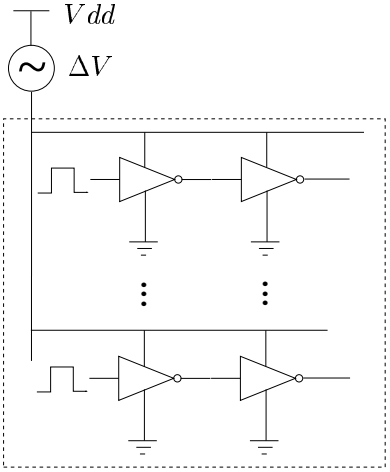

Fig. 5. A block consists of 80 inverters connected to a noisy power supply.

in the system time scale t_1. To obtain the time-varying transfer function from $u(t_2)$ to $\hat{z}(t_1, t_2)$, we start from the Laplace transform of (4) with respect to t_2. Since $\boldsymbol{C}(t_1)$ is independent of t_2, we have $\partial \boldsymbol{C}(t_1)/\partial t_2 = \boldsymbol{0}$. It then follows that

$$\begin{aligned} \frac{\partial(\boldsymbol{C}(t_1)\hat{\boldsymbol{X}}(t_1,s))}{\partial t_1} &+ s\boldsymbol{C}(t_1)\hat{\boldsymbol{X}}(t_1,s) \\ &+\boldsymbol{G}(t_1)\hat{\boldsymbol{X}}(t_1,s) = \boldsymbol{b}U(s) \\ &\hat{Z}(t_1,s) = \boldsymbol{d}^T\hat{\boldsymbol{X}}(t_1,s), \end{aligned} \tag{5}$$

where s denotes the Laplace variable along the t_2 time axis, and the capital symbols denote transformed variables.

Let us now collect samples of $\boldsymbol{C}(t_1)$, $\boldsymbol{G}(t_1)$, $\hat{\boldsymbol{X}}(t_1, s)$, and $\hat{Z}(t_1, s)$ over $t_1 \in [0, T_1]$, at a total of $N+1$ instances $\{t_{1,n}\}_{n=0}^{N}$ with $t_{1,0} = 0$, and $t_{1,N} = T_1$. In the following, we will consider the case where the system is periodic in t_1, and take T_1 to be one period of the system.[1] Eq. (5) can then be re-expressed in a differential form as:

$$\begin{aligned} &\frac{\boldsymbol{C}(t_{1,n})\hat{\boldsymbol{X}}(t_{1,n},s) - \boldsymbol{C}(t_{1,n-1})\hat{\boldsymbol{X}}(t_{1,n-1},s)}{\delta_n} \\ &+[s\boldsymbol{C}(t_{1,n}) + \boldsymbol{G}(t_{1,n})]\hat{\boldsymbol{X}}(t_{1,n},s) = \boldsymbol{b}U(s) \\ &\hat{Z}(t_{1,n},s) = \boldsymbol{d}^T\hat{\boldsymbol{X}}(t_{1,n},s), \;\; \forall n \in [1,N], \end{aligned} \tag{6}$$

where $\delta_n = t_{1,n} - t_{1,n-1}$. Notice that at each snapshot $t_{1,n}$, $\hat{\boldsymbol{X}}(t_{1,n}, s)$ still consists of m unknowns (i.e., the number of node voltages and branch currents of the system), while $\boldsymbol{C}(t_{1,n})$ and $\boldsymbol{G}(t_{1,n})$ are matrices of corresponding dimension. Stacking such vectors, we construct super vectors: $\bar{\boldsymbol{X}}(s) = [\hat{\boldsymbol{X}}^T(t_{1,1}, s), \ldots, \hat{\boldsymbol{X}}^T(t_{1,N}, s)]^T$, $\bar{\boldsymbol{B}} = \mathbf{1}_{N,1} \otimes \boldsymbol{b}$, where $\mathbf{1}_{\mathbf{N,1}}$ is a N by 1 all-one vector, and $\otimes$ denotes Kronecker product. Correspondingly, we also construct block matrices: $\boldsymbol{\mathcal{G}} = \text{diag}\{\boldsymbol{G}(t_{1,1}), \ldots, \boldsymbol{G}(t_{1,N})\}$, $\boldsymbol{\mathcal{C}} = \text{diag}\{\boldsymbol{C}(t_{1,1}), \ldots, \boldsymbol{C}(t_{1,N})\}$, $\boldsymbol{\mathcal{D}} = \boldsymbol{I}_N \otimes \boldsymbol{d}$, and $\boldsymbol{\Delta} = (\text{dial}\{1/\delta_1, \ldots, 1/\delta_N\}(\boldsymbol{I}_N - \boldsymbol{J}_N)) \otimes \boldsymbol{I}_m$, where $\boldsymbol{I}_N$ stands for a N by N identity matrix, and $\boldsymbol{J}_N$ a N by N circulant matrix with first column $[0, 1, 0, \ldots, 0]^T$, and first row $[0, \ldots, 0, 1]$. It can be readily verified that the time-varying transfer function is given by:

$$\begin{aligned} \boldsymbol{H}(s) &= [H(t_{1,1},s), \ldots, H(t_{1,N},s)]^T \\ &= \boldsymbol{\mathcal{D}}^T[s\boldsymbol{\mathcal{C}} + \boldsymbol{\mathcal{G}} + \boldsymbol{\Delta}\boldsymbol{\mathcal{C}}]^{-1}\bar{\boldsymbol{B}}, \end{aligned} \tag{7}$$

such that $\boldsymbol{H}(s)U(s) = \hat{\boldsymbol{Z}}(s)$ with definition $\hat{\boldsymbol{Z}}(s) = [\hat{Z}^T(t_{1,1}, s), \ldots, \hat{Z}^T(t_{1,N}, s)]^T$. Notice that the dimension of $\boldsymbol{\mathcal{C}}$ (and also $\boldsymbol{\mathcal{G}}$, $\boldsymbol{\Delta}$) is $mN \times mN$. Once Eq. (7) is obtained, model order reduction techniques can be applied directly. Along the lines of [8], a model of reduced order $q < mN$ can be obtained by casting (7) into the standard form $\boldsymbol{H}(s) = \boldsymbol{\mathcal{D}}^T[\boldsymbol{I}_{mN} - s\boldsymbol{\mathcal{A}}]^{-1}\boldsymbol{\mathcal{R}}$ with definitions $\boldsymbol{\mathcal{A}} = -[\boldsymbol{\mathcal{G}} + \boldsymbol{\Delta}\boldsymbol{\mathcal{C}}]^{-1}\boldsymbol{\mathcal{C}}$ and $\boldsymbol{\mathcal{R}} = [\boldsymbol{\mathcal{G}} + \boldsymbol{\Delta}\boldsymbol{\mathcal{C}}]^{-1}\bar{\boldsymbol{B}}$, and applying Krylov subspace methods [7, 10]. With block Arnoldi algorithm, the resultant qth order transfer function that approximates $\boldsymbol{H}(s)$ in (7) is given by [3]:

$$\boldsymbol{H}_q(s) = \boldsymbol{L}_q^T[\boldsymbol{I}_q - s\boldsymbol{T}_q]^{-1}\boldsymbol{R}_q, \tag{8}$$

where $\boldsymbol{L}_q = \boldsymbol{V}_q^T\boldsymbol{\mathcal{D}}$ is a $q \times N$ matrix, $\boldsymbol{T}_q$ is a $q \times q$ block-Hessenberg matrix, $\boldsymbol{R}_q = \boldsymbol{V}_q^T\boldsymbol{\mathcal{R}}$ is a $q \times 1$ vector, and $\boldsymbol{V}_q$ is the $mN \times q$ matrix consisting of the q orthogonal bases generated by applying block Arnoldi algorithm to $\boldsymbol{\mathcal{A}}$ and $\boldsymbol{\mathcal{R}}$.

Notice that the dimension of the matrix for model order reduction is $mN \times mN$. To make our methodology applicable to large cells (several hundreds of nodes) with several tenths of sampling time points, we choose Krylov-subspace techniques because of their $\boldsymbol{O}(N)$ complexity. Although SVD based techniques can yield optimal results, their $\boldsymbol{O}(N^3)$ complexity makes them less attractive in solving our problem.

B. Macromodel Representation

Developed on the basis of (4), the transfer function (8) of order q corresponds to the ODE system in Fig. 3, which translates the noise in power supply grids to its corresponding current change. In order to obtain this current, we first transform (8) into time domain as follows:

$$-\boldsymbol{T}_q\frac{d\boldsymbol{x}}{dt} + \boldsymbol{x} = \boldsymbol{R}_q u(t) \quad z(t) = \boldsymbol{l}_q(t)\boldsymbol{x}(t), \tag{9}$$

[1] For more general cases, and frequency domain treatments, the reader is referred to [8].

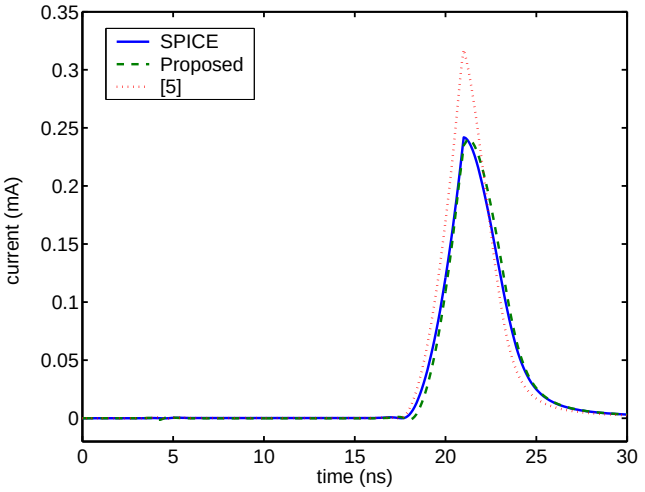

Fig. 6. The comparison of currents drawn by an inverter using SPICE MOSFET model, our proposed macromodel and the model in [5].

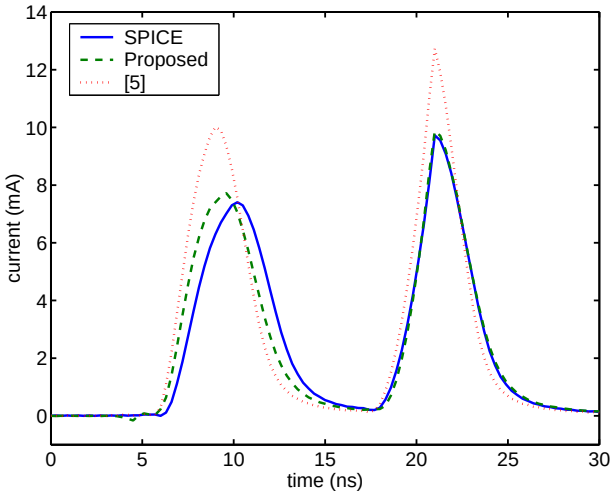

Fig. 7. The comparison of currents drawn by an 80-inverter-block using SPICE MOSFET model, our proposed macromodel and the model in [5].

where $\boldsymbol{x}$ is a vector of size q, $z(t)$ is the output (that is, $\Delta I(t)$ in our macromodel in Fig. 3), and $\boldsymbol{l}_q(t)$ is the $q \times 1$ time-varying vector that relates the system (states) to the output. To link $\boldsymbol{l}_q(t)$ with the $q \times N$ matrix $\boldsymbol{L}_q$ in (8), we notice that the nth column of $\boldsymbol{L}_q$ is nothing but $\boldsymbol{l}_q(t_{1,n})$, $\forall n \in [1, N]$.

Summarizing, the macromodel corresponding to any specific digital cell can be uniquely represented by $I_0(t)$ and the ODE system that generates $\Delta I(t)$ according to the power supply noise. Being independent of the power supply variation, $I_0(t)$ can be computed off-line. The other current $\Delta I(t)$, however, relies on the power supply noise, and has to be computed by taking the overall system as a whole. But notice that the ODE system parameters captured in $\boldsymbol{T}_q$, $\boldsymbol{R}_q$, and $\boldsymbol{L}_q$ do not depend on the voltage variation, and can thus be computed off-line, and stored together with $I_0(t)$ in a cell library. Moreover, different from the original system in (2), the LTV ODE system corresponding to each digital cell consists of a small number ($1 \sim 10$) of nodes. The latter enjoys high-accuracy and low-complexity, when included in a complete power supply distribution analysis circuit that contains the package model and on-chip power networks, and is thus readily applicable to large scale circuits.

III. Results

In this section, we first apply the macromodel extraction method detailed in the preceding section to a single inverter with imperfect power supply, in order to verify the correctness of our model. The single inverter with two transistors is depicted in Fig. 4. After that, we apply our method to a digital block constructed by stacking a total of 40 inverter chains, each containing 2 inverters, as shown in Fig. 5, and generate its corresponding small macromodel with all inputs from low to high.

In both examples, the MOSFET is simulated using Schichman-Hodges model with $\mu_n C_{ox} \frac{W}{L} = 1.6 \times 10^{-4}$A/V^2 for both NMOS and PMOS, load $C = 0.5$pF (see e.g., [6]). It is worth mentioning that the Schichman-Hodges model is not mandatory here, since our macromodel extraction is applicable to more comprehensive models, such as BSIM3. In our simulations, we adopt the Schichman-Hodges model for simplicity. Ideal power supply is 2V, and the voltage drop is always 10% of the ideal value, i.e., 0.2V. For both examples, period of logic input is 30ns.

Example 1: In the first example, we choose $t_{1,n} = 100n$ ps, $\forall n \in [0, 300]$, and the ODE system order $q = 1$, which infers that $\boldsymbol{T}_q$, $\boldsymbol{R}_q$, and $\boldsymbol{l}_q(t)$ are all scalers. The latter brings the computational complexity to the same level as the macromodel in [5]. The accuracy of the two, however, is markedly different, as shown in Fig. 6. When power supply voltage drops 10% from its ideal value, the peak current drops more than 30%. Notice that the current computed using our macromodel closely matches the true not only in peak value, but also in slope. As the former is important in the IR drops, the latter is critical for the $L\, dI/dt$ effects.

Example 2: In the example with 80 converters, we choose $t_{1,n} = 600n$ ps, $\forall n \in [0, 50]$, and the order of the ODE system in Fig. 3 to be $q = 2$. Despite the large block size, we observe the accuracy improvement with respect to [5] once again, as shown in Fig. 7.

TABLE I
Accuracy comparison

one inverter		80 inverters	
Proposed	[5]	Proposed	[5]
0.70%	31.22%	2.35%	35.18%

These examples show that the currents computed using our reduced-order macromodels are very close to that generated using the original MOSFET model (see also Table I). But as mentioned before, the computational complexity of the two are quite different. To illustrate the complexity reduction, we list the run-time of both models in the following table, for each

of the testing examples:

TABLE II
RUN-TIME COMPARISON

one inverter		80 inverters	
SPICE	Proposed	SPICE	Proposed
13.63s	5.42s	365.43s	0.63s

The seemingly inconsistency between 5.42s and 0.63s that corresponding to the one and 80 inverter cases, respectively, comes from their different step sizes in time revolution, as detailed in the example descriptions. Also notice that complexity reduction is achieved not only through the nonlinear-to-linear conversion, but also through order reduction. As a result, although the single-inverter already enjoys a 60% run-time reduction, the 80-inverter case exhibits more extensive reduction of 99.8%, since its relatively larger size leaves more space for order reduction.

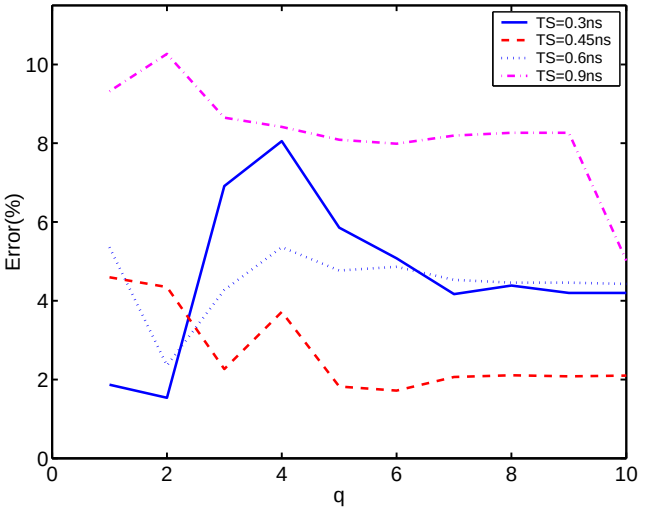

Fig. 8. The comparison of accuracy of proposed model for an 80-inverter-block using different qs and time steps (TS).

To explore the tradeoffs between the accuracy and the complexity, we present some results with different qs and time steps in Fig. 8. As shown in Fig. 8, choosing the right step size is critical to the accuracy. On the one hand, the time step should be fine enough to capture the switching properties of digital circuits. On the other hand, the number of time steps determines the size of the system matrix for Arnoldi to compress. Larger number of time steps imply higher reduction ratio in Arnoldi, thus the accuracy is less. Also, larger number of time steps increase the required memory for storing the model and slow down the simulation. We also notice that larger q tends to yield more accurate macromodel, but it also increases the complexity of the model.

IV. CONCLUSIONS

In this paper, we established a LTV macromodel of digital cells for current estimation. Relying on TVP and model reduction techniques, this macromodel is linear and has small size. As a result, the macromodel is readily applicable to perform system-level noise analysis and assessing the reliability of power distribution networks, even in large-scale circuits. Moreover, as a replacement of its corresponding original digital cell during system level analysis, our digital cell macromodel can interact with power supply voltage variations just as the original cell does, thus provides very reliable results. As we presented in section III, our proposed model significantly speeds up the computation while still offering excellent accuracy ($2 \sim 10\%$ peak current error).

In the future, we will also take into account timing information when building system-level model with our proposed macromodel. This will allow us to obtain even more accurate results. Since propagation and gate delay varies with the power supply voltage, we will develop methodologies that are able to simultaneously incorporate noise analysis and timing analysis. Furthermore, we will also apply hierarchical macromodel extraction to the subsystems at every level, and pursue the challenge of deriving a whole-chip macromodel in a bottom-up manner, in order to further speed up the computation, and to prepare for the billion-gate circuits of the future.

V. ACKNOWLEDGMENT

This work was supported by NSF under Grant No.s CCR-0204278 and CCR-0312079, by DARPA under Grant No. SA0302103, and by SRC CADT program.

REFERENCES

[1] Semiconductor Industry Association, international technology roadmap for semiconductors. 2001.
[2] X. Aragones, J. Gonzalez, and Rubio. *Analysis and Solutions for Switching Noise Coupling in Mixed-Signal ICs*. Kluwer Academic Publishers, Boston, 1 edition, 1999.
[3] D. L. Boley. Krylov space methods on state-space control models. *Circuits, Systems and Signal Processing*, 13:733–758, 1994.
[4] H. H. Chen and D. D. Ling. Power supply noise analysis methodology for deep-submicron VLSI design. In *Proc. of IEEE DAC*, pages 638–643, Anaheim, CA, June, 1997.
[5] A. Dharchowdhury, R. Panda, D. Blaauw, R. Vaidyanathan, B. Tutuianu, and D. Bearden. Design and analysis of power distribution networks in PowerPC^{TM} microprocessors. In *Proc. of IEEE DAC*, pages 738–743, Anaheim, CA, June 15-19, 1998.
[6] D. A. Johns and K. Martin. *Analog Integrated Circuit Design*. John Wiely & Sons Inc, 1996.
[7] A. Odabasioglu, M. Celik, and L.T. Pileggi. Prima: passive reduced-order interconnect macromodeling algorithm. In *IEEE/ACM International Conference on Computer-Aided Design*, pages 257–260, San Jose, CA, Nov 9-13, 1997.
[8] 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.
[9] J. Roychowdhury. Analysing circuits with widely separated time scales using numerical pde methods. *IEEE Trans. on Circuits and Systems I: Fundamental Theory and Applications*, 48(5):578–594, May 2001.
[10] Y. Saad. *Iterative Methods for Sparse Linear Systems*. PWS-Kent, Boston, 1996.
[11] S. S. Sapatnekar and H. Su. Analysis and optimization of power grids. *IEEE Design & Test of Computers*, 20(3):7–15, May 2003.
[12] K. L. Shepard and D. J. Kim. Static noise analysis for digital integrated circuits in partially-depleted silicon-on-insulator technology. In *Proc. of IEEE DAC*, pages 239–242, Los Angeles, CA, June 5-9, 2000.
[13] M. v. Heijningen, M. Badaroglu, S. Donnay, M. Engels, and I. Bolsens. High-level simulation of substrate noise generation including power supply noise coupling. In *Proc. of IEEE DAC*, pages 738–743, Los Angeles, CA, June 5-9, 2000.