Reduced-Order Modeling of Time-Varying Systems

Jaijeet Roychowdhury

***Abstract*— We present algorithms for reducing large circuits, described at SPICE-level detail, to much smaller ones with similar input–output behavior. A key feature of our method, called time-varying Padé (TVP), is that it is capable of reducing *time-varying* linear systems. This enables it to capture frequency-translation and sampling behavior, important in communication subsystems such as mixers and switched-capacitor filters. Krylov-subspace methods are employed in the model reduction process. The macromodels can be generated in SPICE-like or AHDL format, and can be used in both time- and frequency-domain verification tools. We present applications to wireless subsystems, obtaining size reductions and evaluation speedups of orders of magnitude with insignificant loss of accuracy. Extensions of TVP to nonlinear terms and cyclostationary noise are also outlined.**

***Index Terms*— AHDL, Arnoldi, Krylov, Lanzos, macromodelling, nonlinear systems, reduced-order modelling, time-varying systems.**

I. Introduction

VERIFYING systems hierarchically at different levels of abstraction is an important task in communications design. For this task, small macromodels need to be generated that abstract, to a given accuracy, the behavior of much bigger subsystems. For systems with time varying and nonlinear blocks, macromodels are typically constructed by manually abstracting circuit operation into simpler forms, often aided by extensive nonlinear simulations. This process has disadvantages. Simulation does not provide parameters of interest (such as poles and zeros) directly; obtaining them by inspection from frequency responses can be computationally expensive. Manual abstraction can miss nonidealities or interactions that the designer is unaware of. Generally speaking, manual macromodeling is heuristic, time consuming, and highly reliant on detailed internal knowledge of the system under consideration.

In this paper, we present an algorithmic technique for abstracting small macromodels from SPICE-type descriptions of many kinds of subsystems encountered in communication systems. Named time-varying Padé (TVP), the method reduces a large linear time-varying (LTV) system to a small one. The LTV model is adequate for many apparently nonlinear systems, like mixers and switched-capacitor filters, where the signal path is designed to be linear, even though other inputs (e.g., local oscillators, clocks) may cause "nonlinear" parametric changes to the system. For capturing distortion and intermodulation effects, we outline extensions for capturing low-order nonlinear terms in the input–output transfer function. We also sketch how TVP can be used to produce cyclostationary noise macromodels of time-varying systems.

Reduced-order modeling is well established for circuit applications (e.g., AWE [6], [21], [28], PVL [11]–[13], PRIMA [26]), but to the best of our knowledge, existing methods are applicable only to linear time-invariant (LTI) systems. Hence, they are inadequate for communication blocks with properties like frequency translation, which cannot be represented by LTI models. LTV descriptions of a system, on the other hand, can capture frequency translation and mixing/switching behavior. LTV transfer functions are often computed in the context of radio frequency (RF) simulation (e.g., plotting frequency-responses or calculating cyclostationary noise [23], [35], [39]), but a formulation suitable for model reduction has not been available. The basic difficulty in generalizing LTI model-reduction techniques to the LTV case has been the interference of system time variations with input time variations. A key step in this work is to separate the two time-scales, using recent concepts of multiple time variables and the multirate partial differential equation (MPDE) [3], [31], [34], resulting in forms for the LTV transfer function that are suitable for model reduction.[1] Padé approximation of this transfer function results in a smaller system, any desired number of moments of which match those of the original system.

TVP has several useful features. The computation/memory requirements of the method scale almost linearly with circuit size, thanks to the use of factored-matrix computations and iterative linear algebra [15], [24], [29], [35]. TVP provides the reduced model as a LTI system followed by a memoryless mixing operation; this makes it easy to incorporate the macromodel in existing circuit simulators, as well as in system-level simulators supporting any analog high-level description language (AHDL) with linear elements and ideal multipliers. TVP itself can be implemented easily in existing simulation tools, including nonlinear time-domain simulators like SPICE, nonlinear frequency-domain simulators using harmonic balance, as well as in LTV simulators like SWITCAP and SIMPLIS. Existing LTI model-reduction codes can be used as black boxes in TVPs implementation. Like its LTI counterparts, TVP based on Krylov methods (Section III-B) is numerically well conditioned and can directly produce dominant poles and residues. By providing an algorithmic means of generating reduced-order models, TVP enables macromodels of communication subsystems to be coupled to detailed realizations much more tightly and quickly than previously possible. This can significantly reduce the number of iterations it takes to settle on a final design. Furthermore, since there is no relation between the topology or components of the original circuit and

Manuscript received February 1, 1999; revised July 29, 1999.
The author is with Bell Laboratories, Murray Hill, NJ 07974 USA.
Publisher Item Identifier S 1057-7130(99)08719-4.

[1] An alternative formulation of this transfer function was also announced [27] shortly after the present technique first appeared [30], [32], [33].

the reduced one, macromodels generated by TVP can be used to protect intellectual property without sacrificing accuracy.

The remainder of the paper is organized as follows. In Section II, the MPDE is used to obtain the LTV transfer function in forms useful for model reduction. In Section III, Padé approximation and reduced-order modeling of the LTV transfer function is presented. Extensions to nonlinear terms are described in Section IV. Cyclostationary noise macromodeling with TVP is described in Section V. Finally, four examples of the application of TVP are presented in Section VI.

II. LTV Transfer Function

We consider a nonlinear system driven by a large signal $b_l(t)$ and a small input signal $u(t)$ to produce an output $z_t(t)$ (for simplicity, we take both $u(t)$ and $z(t)$ to be scalars; the generalization to the vector case is straightforward). The nonlinear system is modeled using vector differential-algebraic equations (DAEs), a description adequate for circuits [7] and many other applications

$$\frac{\partial q(y(t))}{\partial t} + f(y(t)) = b_l(t) + bu(t) \quad z_t(t) = d^T y(t). \quad (1)$$

In the circuit context, $y(t)$ is a vector of node voltages and branch currents; $q()$ and $f()$ are nonlinear functions describing the charge/flux and resistive terms, respectively, in the circuit. b and d are vectors that link the input and output to the rest of the system.

We now move to the MPDE [3], [31], [34] form of (1). Doing so enables the input and system time scales to be separated and, as will become apparent, leads to a form of the LTV transfer function useful for reduced-order modeling. The move to the MPDE (2), below, is justified by the fact (proved in, e.g., [31], [34]) that any solution of (2) generates a solution of (1)

$$\begin{aligned}\frac{\partial q(\hat{y})}{\partial t_1} + \frac{\partial q(\hat{y})}{\partial t_2} + f(\hat{y}(t_1,\ t_2)) &= b_l(t_1) + bu(t_2)\\ \hat{z}_t(t_1, t_2) &= d^T \hat{y}(t_1, t_2)\\ z_t(t) &= \hat{z}_t(t, t). \quad (2)\end{aligned}$$

The hatted variables in (2) are bivariate (i.e., two-time) forms of the corresponding variables in (1).

To obtain the output component linear in u, we perform a linearization around the solution of (2) when $u(t_2) \equiv 0$. Let this solution be $\hat{y}^*(t_1)$ (note that we can always select $\hat{y}^*$ to be independent of t_2). Linearization about $\hat{y}^*$ yields the linear MPDE

$$\begin{aligned}&\frac{\partial (C(t_1)\hat{x}(t_1, t_2))}{\partial t_1} + \frac{\partial (C(t_1)\hat{x}(t_1, t_2)))}{\partial t_2} + G(t_1)\hat{x}(t_1, t_2)\\ &\quad = bu(t_2)\\ &\hat{z}(t_1, t_2) = d^T\hat{x}(t_1, t_2); \quad z(t) = \hat{z}(t, t). \quad (3)\end{aligned}$$

In (3), the quantities $\hat{x}$, $\hat{z}$, and z are the small-signal versions of $\hat{y}$, $\hat{z}_t$, and z_t, respectively; $C(t_1) = (\partial q(\hat{y})/\hat{y})|_{\hat{y}^*(t_1)}$ and $G(t_1) = (\partial f(\hat{y})/\hat{y})|_{\hat{y}^*(t_1)}$ are time-varying matrices.

Note that the bi-variate output $\hat{z}(t_1, t_2)$ is linear in the input $u(t_2)$, but that the relationship is time-varying because of the presence of t_1. To obtain the time-varying transfer function from u to $\hat{z}$, we Laplace transform (3) with respect to t_2

$$\begin{aligned}&\frac{\partial (C(t_1)\hat{X}(t_1, s))}{\partial t_1} + sC(t_1)\hat{X}(t_1, s) + G(t_1)\hat{X}(t_1, s) = bU(s)\\ &\hat{Z}(t_1, s) = d^T\hat{X}(t_1, s). \quad (4)\end{aligned}$$

In (4), s denotes the Laplace variable along the t_2 time axis; the capital symbols denote transformed variables.

By defining the operator

$$\frac{D}{dt_1}[v] = \frac{\partial (C(t_1)v)}{\partial t_1} \quad (5)$$

we can rewrite (4) as

$$\begin{aligned}\left(\frac{D}{dt_1}[] + sC(t_1) + G(t_1)\right)\hat{X}(t_1, s) &= bU(s)\\ \hat{Z}(t_1, s) &= d^T\hat{X}(t_1, s) \quad (6)\end{aligned}$$

and obtain an *operator form* of the time-varying transfer function $H(t_1, s)$

$$\begin{aligned}H(t_1, s) &= d^T\left(\frac{D}{dt_1}[] + sC(t_1) + G(t_1)\right)^{-1}[b]\\ \hat{Z}(t_1, s) &= H(t_1, s)U(s). \quad (7)\end{aligned}$$

Finally, the frequency-domain relation between the output $z(t)$ and its bi-variate form $\hat{z}$ is

$$Z(s) = \int_{-\infty}^{\infty} \hat{Z}_s(s - s_2, s_2)\, ds_2 \quad (8)$$

where $Z(s)$ is the Laplace transform of $z(t)$ and $\hat{Z}_s(s_1, s_2)$ the two-dimensional Laplace transform of $\hat{z}(t_1, t_2)$, or equivalently, the Laplace transform of $\hat{Z}(t_1, s_2)$ with respect to t_1.

The operator form (7) is already useful for reduced-order modeling. We can proceed further, however, by expanding the t_1 dependence in a basis. This leads to matrix forms of the transfer function, to which existing model reduction codes can be applied—a very desirable feature for implementation. Frequency-domain basis functions, considered in Section II-A, are natural for applications with relatively sinusoidal variations, while time-domain ones (Section II-B) are better suited to systems with switching behavior and those that are not periodic.

A. Frequency-Domain Matrix Form

Assume $C(t_1)$ and $G(t_1)$ to be periodic with angular frequency ω_0. Define $W(t_1, s)$ to be the operator-inverse in (7)

$$\begin{aligned}W(t_1, s) &= \left(\frac{D}{dt_1}[] + sC(t_1) + G(t_1)\right)^{-1}[b]\\ &\Rightarrow \frac{D}{dt_1}[W(t_1, s)] + [sC(t_1) + G(t_1)]W(t_1, s)\\ &= b. \quad (9)\end{aligned}$$

Assume $W(t_1, s)$ also to be in periodic steady-state in t_1, and expand $C(t_1)$, $G(t_1)$, and $W(t_1, s)$ in Fourier series with coefficients C_i, G_i, and $W_i(s)$, respectively

$$\begin{aligned} C(t_1) &= \sum_{i=-\infty}^{\infty} C_i e^{\jmath i\omega_0 t_1} \\ G(t_1) &= \sum_{i=-\infty}^{\infty} G_i e^{\jmath i\omega_0 t_1} \\ W(t_1, s) &= \sum_{i=-\infty}^{\infty} W_i(s) e^{\jmath i\omega_0 t_1}. \end{aligned} \tag{10}$$

Now define the following long vectors of Fourier coefficients to be

$$\begin{aligned} \vec{W}_{FD}(s) &= [\cdots, W_{-2}^T(s), W_{-1}^T(s), W_0^T(s), W_1^T(s), W_2^T(s), \cdots]^T \\ \vec{B}_{FD} &= [\cdots, 0, 0, b^T, 0, 0, \cdots]^T. \end{aligned} \tag{11}$$

By putting (10) into (9) and equating coefficients of $e^{\jmath\omega_0 t_1}$, it can be verified that the following block-matrix equation holds:

$$[s\mathcal{C}_{FD} + \underbrace{(\mathcal{G}_{FD} + \Omega\mathcal{C}_{FD})}_{\text{denote this by } \mathcal{J}_{FD}}]\vec{W}_{FD}(s) = \vec{B}_{FD} \tag{12}$$

where

$$\begin{aligned} \mathcal{C}_{FD} &= \begin{pmatrix} \vdots & \vdots & \vdots \\ \cdots C_0 & C_{-1} & C_{-2}\cdots \\ \cdots C_1 & C_0 & C_{-1}\cdots \\ \cdots C_2 & C_1 & C_0 \ \cdots \\ \vdots & \vdots & \vdots \end{pmatrix} \\ \mathcal{G}_{FD} &= \begin{pmatrix} \vdots & \vdots & \vdots \\ \cdots G_0 & G_{-1} & G_{-2}\cdots \\ \cdots G_1 & G_0 & G_{-1}\cdots \\ \cdots G_2 & G_1 & G_0 \ \cdots \\ \vdots & \vdots & \vdots \end{pmatrix} \\ \mathcal{J}_{FD} &= \mathcal{G}_{FD} + \Omega\mathcal{C}_{FD}, \\ \Omega &= \jmath\omega_0 \begin{pmatrix} \ddots &&&&&& \\ & -2I &&&&& \\ && -I &&&& \\ &&& 0I &&& \\ &&&& I && \\ &&&&& 2I & \\ &&&&&& \ddots \end{pmatrix}. \end{aligned} \tag{13}$$

Now denote

$$E(t_1) = [\cdots, Ie^{-\jmath 2\omega_0 t_1}, Ie^{-\jmath\omega_0 t_1}, I, Ie^{\jmath\omega_0 t_1}, Ie^{\jmath 2\omega_0 t_1}, \cdots]^T. \tag{14}$$

From (12), (9), and (7), we obtain the following matrix expression for $H(t_1, s)$:

$$H(t_1, s) = d^T E^T(t_1)[s\mathcal{C}_{FD} + \mathcal{J}_{FD}]^{-1}\vec{B}_{FD}. \tag{15}$$

From (14), note that $E(t_1)$ can be written in the Fourier expansion

$$\begin{aligned} E(t_1) &= \sum_{i=-\infty}^{\infty} E_i e^{\jmath i\omega_0 t_1} \\ E_i &= [\cdots, 0, 0, \underbrace{I}_{i\text{th position}}, 0, \cdots, 0, \cdots]^T. \end{aligned} \tag{16}$$

Hence, we can rewrite (15) in a Fourier series

$$\begin{aligned} H(t_1, s) &= \sum_{i=-\infty}^{\infty} H_i(s) e^{\jmath i\omega_0 t_1} \\ H_i(s) &= d^T E_i^T [s\mathcal{C}_{FD} + \mathcal{J}_{FD}]^{-1}\vec{B}_{FD}. \end{aligned} \tag{17}$$

Equation (17) implies that any linear periodic time-varying system can be decomposed into LTI systems followed by memoryless multiplications with $e^{\jmath i\omega_0 t}$. The quantities $H_i(s)$ will be called *baseband-referred transfer functions*.

We proceed to re-write (17) for all values of i as a single block-matrix equation. Define

$$\vec{H}_{FD}(s) = [\cdots, H_{-2}(s), H_{-1}(s), H_0(s), H_1(s), H_2(s), \cdots]^T. \tag{18}$$

Then

$$\begin{aligned} \vec{H}_{FD}(s) &= \mathcal{D}^T[s\mathcal{C}_{FD} + \mathcal{J}_{FD}]^{-1}\vec{B}_{FD} \\ \mathcal{D} &= \begin{pmatrix} \ddots &&&& \\ & d &&& \\ && d && \\ &&& d & \\ &&&& \ddots \end{pmatrix}. \end{aligned} \tag{19}$$

Equation (19) is a block matrix equation for a single-input multioutput transfer function. If the size of the LTV system (3) is n, and N harmonics of the LTV system are considered in practice, then $\vec{B}_{FD}$ is a vector of size $Nn \times 1$, $\mathcal{C}_{FD}$, and $\mathcal{J}_{FD}$ are square matrices of size $Nn \times Nn$, $\mathcal{D}$ is a rectangular matrix of size $Nn \times N$, and $\vec{H}_{FD}(s)$ is a vector of size N.

B. Time-Domain Matrix Form

Consider (9) again

$$\begin{aligned} \frac{D}{dt_1}[W(t_1, s)] + [sC(t_1) + G(t_1)]W(t_1, s) &= b \\ H(t_1, s) &= d^T W(t_1, s) \end{aligned} \tag{20}$$

We collocate (20) over $t_1 \in [0, T_1]$ at $N+1$ samples $t_{1,0} = 0, t_{1,1}, \cdots, t_{1,N}$, using a linear multistep formula (say Backward Euler) to express the differential in terms of samples. Denote by the long vectors $\vec{W}_{FD}$ and $\vec{B}_{FD}$ the samples of $W(t_1, s)$ and b

$$\begin{aligned} \vec{W}_{TD}(s) &= [W^T(t_{1,0}, s), W^T(t_{1,1}, s), \cdots, W^T(t_{1,N}, s)]^T \\ \vec{B}_{TD} &= [b^T, b^T, b^T, \cdots, b^T]^T. \end{aligned} \tag{21}$$

We then obtain the following matrix form for the collocated equations:

$$[s\mathcal{C}_{TD} + \mathcal{J}_{TD}]\vec{W}_{TD}(s) = \vec{B}_{TD} \tag{22}$$

where

$$\begin{aligned}
\mathcal{J}_{TD} &= \mathcal{G}_{TD} + \Delta \mathcal{C}_{TD}, \\
\mathcal{G}_{TD} &= \begin{bmatrix} G(0) & & & \\ & G(t_{1,1}) & & \\ & & \ddots & \\ & & & G(t_{1,N}) \end{bmatrix} \\
\Delta &= \begin{pmatrix} \frac{1}{\delta_1} I & & & \\ -\frac{1}{\delta_2} I & \frac{1}{\delta_2} I & & \\ & \ddots & \ddots & \\ & & -\frac{1}{\delta_N} I & \frac{1}{\delta_N} I \end{pmatrix} \\
\delta_i &= t_{1,i} - t_{1,i-1} \\
\mathcal{C}_{TD} &= \begin{bmatrix} C(0) & & & \\ & C(t_{1,1}) & & \\ & & \ddots & \\ & & & C(t_{1,N}) \end{bmatrix}
\end{aligned} \tag{23}$$

and we have assumed zero initial conditions $W(0, s) \equiv 0$. If the system is periodic, then periodic boundary conditions can be applied; the only change in (22) and (23) is to Δ, the differentiation matrix, which becomes

$$\Delta = \begin{pmatrix} \frac{1}{\delta_1} I & & & -\frac{1}{\delta_1} I \\ -\frac{1}{\delta_2} I & \frac{1}{\delta_2} I & & \\ & \ddots & \ddots & \\ & & -\frac{1}{\delta_N} I & \frac{1}{\delta_N} I \end{pmatrix}. \tag{24}$$

Define

$$\vec{H}_{TD}(s) = [H(t_{1,0}, s), H(t_{1,1}, s), \cdots, H(t_{1,N}, s)]^T. \tag{25}$$

Then

$$\vec{h}_{TD}(s) = \mathcal{D}^T [s\mathcal{C}_{TD} + \mathcal{J}_{TD}]^{-1} \vec{B}_{TD} \tag{26}$$

with $\mathcal{D}$ as in (19). Equation (26) is in the same form as (19); both can be used directly for reduced-order modeling, as discussed in the next section.

III. Padé Approximation of the LTV Transfer Function

The LTV transfer function (7), (19), and (26) can be expensive to evaluate, since the dimension of the full system can be large. In this section, methods are presented for approximating $H(t_1, s)$ using quantities of much smaller dimension.

The underlying principle is that of Padé approximation, i.e., for any of the forms of the LTV transfer function, to obtain a smaller form of size q whose first several moments match those of the original large system. This can be achieved in two broad ways, with correspondences in existing LTI model-reduction methods. TVP-explicit (TVP-E), roughly analogous to AWE [6], [28] for LTI systems, involves calculating moments of the large system explicitly and building the reduced order model from these moments. The method is outlined in Section III-A. In Section III-B, we present another procedure called TVP-Krylov (TVP-K), which uses Krylov-subspace methods to replace the large system directly with a smaller one, while achieving moment-matching implicitly. TVP-K is analogous to LTI model-reduction techniques which use the Lanczos and Arnoldi processes (e.g., PVL and MPVL [11], [12], operator-Lanczos methods [4], [5], PRIMA [26], and other Krylov-subspace-based techniques [9], [22]). As in the LTI methods, TVP based on Krylov subspaces has significant accuracy advantages over explicit moment matching. Operator- or matrix-based techniques can be applied to both explicit and Krylov-based TVP; Section III-A describes an operator-based procedure and Section III-B a matrix-based one.

A. TVP-E: TVP Using Explicit Moment Matching

Any of the forms (7), (19), and (26) can be used for explicit moment matching. Here, we illustrate an operator procedure using (7). Rewrite $H(t_1, s)$ from (7) as

$$H(t_1, s) = d^T (I[] + sL[])^{-1} [r(t_1)]$$

where $I[]$ denotes the identity operator

$$\begin{aligned}
L[v] &= \left(\frac{D}{dt_1}[] + G(t_1)\right)^{-1} [C(t_1)v] \\
r(t_1) &= \left(\frac{D}{dt_1}[] + G(t_1)\right)^{-1} [b].
\end{aligned} \tag{27}$$

$H(t_1, s)$ in (27) can be expanded as

$$\begin{aligned}
H(t_1, s) &= d^T (r(t_1) - sL[r(t_1)] + s^2 L[L[r(t_1)]] + \cdots) \\
&= \sum_{i=0}^{\infty} m_i(t_1) s^i
\end{aligned}$$

where

$$m_i(t_1) = (-1)^i \, d^T \underbrace{L[L[\cdots L[r(t_1)] \cdots]]}_{i \text{ applications of } L[]}. \tag{28}$$

$m_i(t_1)$ in (28) are the *time-varying moments* of $H(t_1, s)$. Note that these moments can be calculated explicitly from their definition in (28), by repeated applications of $L[]$. From its definition in (27), applying $L[]$ corresponds to solving a LTV differential equation. If the time-varying system is in periodic steady state, as is often the case in applications, $L[]$ can be applied numerically by solving the equations that arise in the inner loop of harmonic balance or shooting methods. Iterative methods (e.g., [15], [24], [29], [38]) enable large systems of these equations to be solved in linear time, hence the time-varying moments can be calculated easily.

Once the moments $m_i(t_1)$ have been computed, t_1 can be fixed at a given value, and any existing LTI model reduction technique using explicit moments (e.g., AWE) can be run q steps to produce a qth-order reduced model. This step can be repeated for all t_1 values of interest, to produce an overall reduced-order model for $H(t_1, s)$ in the form

$$\tilde{H}^q(t_1, s) = \frac{\sum_{i=0}^{q-1} a_i(t_1)s^i}{\sum_{j=0}^{q} b_j(t_1)s^j} = \sum_{i=0}^{q} \frac{c_i(t_1)}{s+p_i(t_1)}. \tag{29}$$

The simple procedure outlined above has two disadvantages. The first is that model reduction methods using explicit moment matching suffer from numerical ill-conditioning, making them of limited value for q more than ten or so [11]. The second is that the form (29) has time-varying poles. It can be shown (see the Appendix) using Floquet theory that the transfer function $H(t_1, s)$ has a potentially infinite number of poles *that are independent of* t_1 (these poles are simply the Floquet eigenvalues shifted by multiples of the system frequency), together with residues that do, in fact, vary with t_1. It is desirable to obtain a reduced-order model with similar properties. In fact, this requirement can be met by obtaining a reduced system in the time-domain form of (3), which is very desirable for system-level modeling applications. The Krylov-subspace procedures for TVP in Section III-B eliminate both problems.

B. TVP-K: TVP Using Krylov Subspace Methods

In this section, we describe the application of block-Krylov methods [1], [12], [16], [17], [26], [37] to any multi-output matrix form of the LTV transfer function. Krylov-subspace methods provide a numerically stable means of obtaining a reduced-order model; in addition, the reduced transfer functions are in the same form as $H(t_1, s)$ in (7), with similar properties like a possibly infinite number of t_1-invariant poles.

Both (19) and (26) are in the form

$$\vec{H}(s) = \mathcal{D}^T[s\mathcal{C} + \mathcal{J}]^{-1}\vec{B} = \mathcal{L}^T[I - s\mathcal{A}]^{-1}\mathcal{R}$$

where

$$\mathcal{L} = \mathcal{D},\ \mathcal{R} = \mathcal{J}^{-1}\vec{B}, \text{ and } \mathcal{A} = -\mathcal{J}^{-1}\mathcal{C}. \tag{30}$$

Equation (30) can be used directly for reduced-order modeling by block-Krylov methods. We sketch the application of two popular such methods, Lanczos and Arnoldi.

1) Block Lanczos: Running the block-Lanczos algorithm [1], [12], [16], [17] with the quantities $\mathcal{L}$, $\mathcal{A}$, and $\mathcal{R}$ produces the matrices and vectors $L_q^{(L)}$ (of size $q \times N$), $R_q^{(L)}$ (size $q \times 1$), and $T_q^{(L)}$ (size $q \times q$). q is a small integer related to the number of iterations the algorithm is run. Define the qth-order approximant $\vec{H}_q^{(L)}(s)$ by

$$\vec{H}_q^{(L)}(s) = {L_q^{(L)}}^T[I_{q\times q} - sT_q^{(L)}]^{-1}R_q^{(L)}. \tag{31}$$

Then $\vec{H}_q^{(L)}(s) \approx \vec{H}(s)$, in the sense that a certain number of matrix-moments of the two quantities are identical—see [16] for a precise description of the approximation.

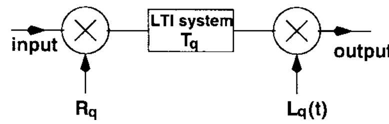


Fig. 1. Floquet from of LPTV system.

2) Block Arnoldi: The block Arnoldi algorithm, described in, e.g., [26], [37], uses $\mathcal{A}$ and $\mathcal{R}$ to produce matrices V_q (of size $Nn \times q$) and $T_q^{(A)}$ (size $q \times q$). V_q is orthogonal (i.e., $V_q^T V_q = I_q$), and $T_q^{(A)}$ block-Hessenberg. It can be shown that

$$\vec{H}_q^{(A)}(s) = \mathcal{L}^T V_q[I_{q\times q} - sT_q^{(A)}]^{-1}V_q^T\mathcal{R} \tag{32}$$

approximates $\vec{H}(s)$ [2].

C. The Reduced Model

Both (31) and (32), in the form

$$\vec{H}_q(s) = L_q^T[I_{q\times q} - sT_q]^{-1}R_q \tag{33}$$

approximate $\vec{H}(s)$. In typical applications, adequate approximations are obtained with fairly small q, ranging from 2 to 30.

Corresponding to (33), a time-domain system of size q can be obtained easily. We illustrate the procedure for the frequency-domain matrix form of Section II-A; the time-domain form of Section II-B is similar, differing simply in the choice of basis functions below. Define

$$\vec{L}_q(t_1) = \sum_{i=-\infty}^{\infty} L_{q,\,i}e^{\jmath i\omega_0 t_1} \tag{34}$$

where $L_{q,\,i}$ is the ith row of L_q. The approximate LTV transfer function $\tilde{H}^q(t_1, s)$ is given by

$$\tilde{H}^q(t_1, s) = \vec{L}_q(t_1)^T[I_{q\times q} - sT_q]^{-1}R_q. \tag{35}$$

Equation (35) is the time-varying transfer function of the following qth-order reduced system of time-domain equations

$$-T_q\frac{\partial\tilde{x}}{t} + \tilde{x} = R_q u(t); \qquad z(t) = \vec{L}_q(t)\tilde{x}(t) \tag{36}$$

where $\tilde{x}(t)$ is a vector of size q, much smaller than that of the original system (3).

D. Useful Features of TVP-K-Generated Macromodels

The TVP-K procedure in Section III-B has a number of notable properties, itemized below.

1) Note that (36) represents a linear *time-invariant* system, followed by a memoryless multiplication that appears only in the output equation. The reduced system is illustrated in Fig. 1. This feature makes the reduced model easy to incorporate as AHDL elements in existing tools, since no time-varying matrices are involved. Only LTI elements (resistors, capacitors, ideal controlled sources) and ideal multiplier elements are required to implement the macromodel.

2) In practice, only the baseband-referred transfer functions corresponding to harmonics of interest can be represented in (18), thereby reducing the number of columns of $\mathcal{D}$. Similarly, any postprocessing for averaging/Fourier analysis can be directly incorporated in (25), thereby reducing the number of time-domain outputs.
3) The form (35) can be shown to imply that $\tilde{H}^q(t_1, s)$ has a possibly infinite number of time-invariant poles, similar to $H(t_1, s)$. Further, the eigenvalues of T_q are the Floquet exponents of the reduced-order model, which approximate those of the original LTV system. The poles and residues of the reduced-order models of $H_i(s)$ can be easily calculated from the eigenvalues of T_q.
4) Krylov-subspace algorithms such as Lanczos and Arnoldi require only matrix-vector products with $\mathcal{C}$ and linear system solutions with $\mathcal{J}$. Though both these matrices can be large, dense or difficult to factor, exploiting structure and using iterative linear algebra techniques can make these computations scale almost linearly with problem size [15], [24], [29], [35], [38]. When these fast techniques are employed, the computation required by the TVP algorithm grows approximately linearly in circuit size and number of harmonics or time-points, making it usable for large problems.
5) The numerical ill-conditioning problem with explicit moment matching in Section III-A is eliminated using Krylov methods, hence TVP can be run up to large values of q if necessary.
6) A system with p_i inputs and p_o outputs can be handled easily, by stacking the extra outputs into $\vec{H}$ (resulting in $\mathcal{D}$ of size $nN \times p_o N$), and incorporating the inputs into $\vec{B}$ (to form a rectangular matrix of size $nN \times p_i$).

IV. Reduced-Order Modeling of Nonlinearities

In the section, we present an extension of TVP for modeling signal path nonlinearities described by Volterra series. Volterra series [25], [36], [40] are a generalization of Taylor series to systems with memory. Given a nonlinear system with input $x(t)$ and output $y(t)$, $y(t)$ can be represented in a Volterra series expansion as

$$y(t) = \sum_{i=0}^{\infty} y_i(t) \tag{37}$$

where

$$y_i(t) = \int \underset{i}{\cdots} \int h_i(t-\tau_1, \cdots, t-\tau_i) x(\tau_1) \cdots x(\tau_i) d\tau_1 \cdots d\tau_i. \tag{38}$$

Equation (37) reduces to a Taylor series if $h_n(t_1, \cdots, t_n) = c_n \delta(t_1, \cdots, t_n)$, i.e., an n-dimensional delta function

$$y(t) = \sum_{i=0}^{\infty} c_i x^i(t). \tag{39}$$

We observe that the $i = 0$ term is the constant term, $i = 1$ the linear term, $i = 2$ the quadratic term, and so on. Furthermore, we note that if $x(t) = \cos(\omega t)$, the ith Volterra term generates components at the ith and *lower* harmonics. For example, the $i = 3$ term of (39) is $c_i \cos^3(\omega t) = (c_i/4)(3\cos(\omega t) + \cos(3\omega t))$, consisting of both first and third harmonics. Thus, higher Volterra terms are useful not only for obtaining harmonic components, but also for modeling gain compression of the linear transfer function.

We outline the procedure for macromodeling nonlinearities by first considering time-invariant systems.

A. Reducing Time-Invariant Nonlinear Systems

We start by specializing (1) to the case of small perturbations about a dc operating point

$$\frac{\partial}{\partial t}[q(y((t))] + f(y(t)) = b^* + b\tilde{u}(t). \tag{40}$$

Let the dc solution of (40) (with $\tilde{u}(t) \equiv 0$) be y^*. Then, we can represent the perturbations $x(t)$ due to nonzero $\tilde{u}(t)$ as

$$\frac{\partial}{\partial t}[q(y^* + x(t))] + f(y^* + x(t)) = b^* + b\tilde{u}(t). \tag{41}$$

Expanding the nonlinear functions $q(\cdot)$ and $f(\cdot)$ in Taylor series, we obtain

$$\frac{\partial}{\partial t}\left[q(y^*) + C_1 x + C_2 x^{②} + C_3 x^{③} + \cdots\right] + f(y^*) + G_1 x + G_2 x^{②} + G_3 x^{③} + \cdots = b^* + b\tilde{u}(t). \tag{42}$$

Here $x^{②}$ represents the vector direct product $x \otimes x$. C_i and G_i represent the ith derivative matrices of $q(\cdot)$ and $f(\cdot)$, respectively. From these definitions, if the size of the original system (40) is n, we have $x^{ⓘ} \in \mathbb{R}^{n^i}$ and $C_i, G_i \in \mathbb{R}^{n \times n^i}$.

To obtain the Volterra formulation, we use a perturbational method. We express $\tilde{u}(t)$ as $\varepsilon u(t)$, where ε is a small scalar parameter. Since DAEs driven by smooth inputs have smooth solutions, $x(t)$ in (41) can be expressed in a Taylor series in ε

$$x(t) = \varepsilon x_1(t) + \varepsilon^2 x_2(t) + \varepsilon^3 x_3(t) + \cdots. \tag{43}$$

Substituting (43) in (42), and collecting the coefficients of powers of ε, the following equations for x_1, x_2, x_3, etc., are obtained:

$$\frac{\partial}{\partial t}[C_1 x_1] + G_1 x_1 = bu(t) \quad (\varepsilon \text{ terms}) \tag{44}$$

$$\frac{\partial}{\partial t}[C_1 x_2] + G_1 x_2 = -\frac{\partial}{\partial t}\left[C_2 x_1^{②}\right] - G_2 x_1^{②} \quad (\varepsilon^2 \text{ terms}) \tag{45}$$

$$\frac{\partial}{\partial t}[C_1 x_3] + G_1 x_3 = -\frac{\partial}{\partial t}\left[C_3 x_1^{③} + C_2(x_1 \otimes x_2 + x_2 \otimes x_1)\right] - G_3 x_1^{③} - G_2(x_1 \otimes x_2 + x_2 \otimes x_1) \quad (\varepsilon^3 \text{ terms}). \tag{46}$$

From (44)–(46), we observe that x_1 is the solution of the linearized system; x_2 is also a solution of the same linearized system but with different inputs (“distortion inputs”), which depend on x_1; and similarly, x_3 results from solving the linearized system with distortion inputs derived from x_1 and x_2.

Before investigating how to represent (45) and (46) by smaller systems, it is instructive to examine the mechanism by which a Krylov-subspace-based technique reduces the linearized system of (44) to a smaller one. Rewriting (44) first as $G_1^{-1}(\partial/\partial t)[C_1 x_1] + x_1 = G_1^{-1} b u(t)$, we obtain the Laplace-domain transfer function between $u(t)$ and $x(t)$ to be

$$X_1(s) = [s\underbrace{G_1^{-1}C_1}_{A} + I]^{-1}\underbrace{G_1^{-1}b}_{r}\,U(s). \tag{47}$$

A Krylov-subspace method simply generates a small set of basis vectors onto which the input and state spaces are projected [18], [19], resulting in the reduced model. We illustrate this projection concept using the Arnoldi method.[2] Run for q steps, Arnoldi generates a rectangular orthonormal matrix $V_q \in \mathbb{R}^{n\times q}$, such that $AV_q = V_q T_q$, where $T_q \in \mathbb{R}^{q\times q}$ is a small square Hessenberg matrix. The size-n linear system is now approximated as a size-q one

$$X_{1,q}(s) = (sT_q + I)^{-1} r_q U(s) \tag{48}$$

with

$$r_q = V_q^T r \tag{49}$$

and

$$X_1(s) \approx V_q X_{1,q}(s). \tag{50}$$

We observe that the reduction process consists simply of: 1) *projecting* the size-n input subspace r onto a size-q subspace (50); 2) using this as input to a size-q linear system (48) to obtain a size-q state-space $X_{1,q}$; and finally 3) representing (i.e., *embedding*) $X_{1,q}$ in the original size-n state–space (50). Equations (48)–(50) can be written in time-domain form as

$$\frac{\partial}{\partial t}\left[T_q x_{1,q}(t)\right] + x_{1,q}(t) = r_q u(t) \tag{51}$$

with

$$r_q = V_q^T r \tag{52}$$

and

$$x_1(t) \approx V_q x_{1,q}(t). \tag{53}$$

An approximation to any output $y_1(t) = d^T x_1(t)$ of the original system can thus be obtained directly from the reduced state–space as $y_1(t) \approx l^T x_{1,q}(t)$, where $l^T = d^T V_q$.

We can now apply the concept of projection and embedding to the nonlinear reduction problem. Observe that an essential difficulty in reducing (45) is that, potentially, the direct product of the entire size-n state space x_1 with itself is used as input. We can, however, reduce the dimensionality of this input by representing x_1 as the embedding in (53) from a q_1-sized subspace. We then have

$$\begin{aligned} x_1 \otimes x_1 &\approx (V_{q_1} x_{1,q_1}) \otimes (V_{q_1} x_{1,q_1}) \\ &= (V_{q_1} \otimes V_{q_1})(x_{1,q_1} \otimes x_{1,q_1}). \end{aligned} \tag{54}$$

Using (54), (45) becomes

$$\frac{\partial}{\partial t}\left[C_1 x_2\right] + G_1 x_2 \approx -\frac{\partial}{\partial t}\left[(C_2 V_{q_1}^{\textcircled{2}}) x_{1,q_1}^{\textcircled{2}}\right] - \left(G_2 V_{q_1}^{\textcircled{2}}\right) x_{1,q_1}^{\textcircled{2}}. \tag{55}$$

[2]We thank Alper Demir for pointing out the advantages of Arnoldi over Lanczos in this context.

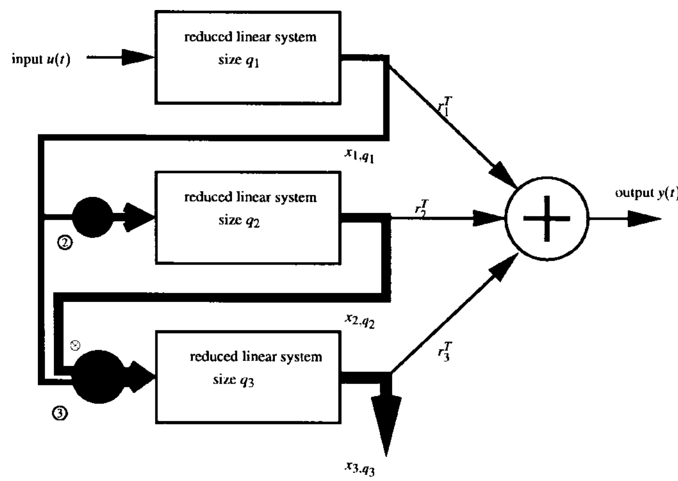


Fig. 2. Block structure of reduced system with nonlinearities.

Note that $x_{1,q_1}^{\textcircled{2}} \in \mathbb{R}^{q_1^2}$ and $G_2 V_{q_1}^{\textcircled{2}}$, $C_2 V_{q_1}^{\textcircled{2}} \in \mathbb{R}^{n\times q_1^2}$; in other words, the input to (55) is of size q_1^2.

For Krylov-based reduction, (55) can be reframed in block-matrix terms as

$$\frac{\partial}{\partial t}\left(\begin{bmatrix} C_1 & -I \\ 0 & 0 \end{bmatrix}\begin{bmatrix} x_2 \\ x_{2e} \end{bmatrix}\right) + \begin{bmatrix} G_1 & 0 \\ 0 & I \end{bmatrix}\begin{bmatrix} x_2 \\ x_{2e} \end{bmatrix} \approx -\begin{bmatrix} G_2 V_{q_1}^{\textcircled{2}} \\ C_2 V_{q_1}^{\textcircled{2}} \end{bmatrix} x_{1,q_1}^{\textcircled{2}} \tag{56}$$

Equation (56) is a LTI system with q_1^2 inputs; it can, therefore, be reduced to a smaller system, using Arnoldi with multiple starting vectors. Let the reduced size be q_2 and the corresponding subspace be $\tilde{V}_{q_2}$; define $V_{q_2} = [I_n \quad 0]\tilde{V}_{q_2}$, and let P_{q_2,q_1} be the permutation matrix that reorders $z_2 \otimes z_1$ to produce $z_1 \otimes z_2$, for any vectors $z_1 \in \mathbb{R}^{q_1}$ and $z_2 \in \mathbb{R}^{q_2}$. Similar to (56), (46) can be expressed as

$$\begin{aligned} &\frac{\partial}{\partial t}\left(\begin{bmatrix} C_1 & -I \\ 0 & 0 \end{bmatrix}\begin{bmatrix} x_3 \\ x_{3e} \end{bmatrix}\right) + \begin{bmatrix} G_1 & 0 \\ 0 & I \end{bmatrix}\begin{bmatrix} x_3 \\ x_{3e} \end{bmatrix} \\ &\quad \approx -\begin{bmatrix} G_3 V_{q_1}^{\textcircled{3}} & (G_2((V_{q_2} \otimes V_{q_1}) P_{q_2,q_1} + V_{q_1} \otimes V_{q_2})) \\ C_3 V_{q_1}^{\textcircled{3}} & (C_2((V_{q_2} \otimes V_{q_1}) P_{q_2,q_1} + V_{q_1} \otimes V_{q_2})) \end{bmatrix} \\ &\qquad \cdot \begin{bmatrix} x_{1,q_1}^{\textcircled{3}} \\ x_{1,q_1} \otimes x_{2,q_2} \end{bmatrix}. \end{aligned} \tag{57}$$

Equation (57) is a LTI system with $q_1^3 + q_1 q_2$ inputs. This system can, in turn, be reduced to a smaller one (of size q_3) using the Arnoldi method. The overall structure of the reduced system is shown in Fig. 2.

We note that the effectiveness of size reduction is limited by the rapidly increasing sizes of the distortion input sources to the higher order Volterra systems. The actual input sizes, however, are determined by the numerical rank of the input coefficient matrices, e.g.,

$$\begin{bmatrix} G_2 V_{q_1}^{\textcircled{2}} \\ C_2 V_{q_1}^{\textcircled{2}} \end{bmatrix}$$

for (56). Owing to the fact that higher order derivatives of typical circuit functions are typically very sparse, this rank can be lower than the nominal size of the input space.

B. Reducing Time-Varying Nonlinear Systems

The procedure outlined in Section IV-A can be extended to nonlinear terms of a time-varying system. We start with

$$\frac{\partial}{\partial t}\left[q(y((t))\right] + f(y(t)) = b^*(t) + b\tilde{u}(t) \tag{58}$$

where $u(t)$ is a small input perturbation. To analyze perturbations conveniently, we now switch to the MPDE form of the differential equation (2)

$$\left(\frac{\partial}{\partial t_1} + \frac{\partial}{\partial t_2}\right) q(\hat{y}(t_1, t_2)) + f(\hat{y}(t_1, t_2)) = b^*(t_1) + b\tilde{u}(t_2). \tag{59}$$

Let the unperturbed solution of (59) (with $\tilde{u}(t_2) \equiv 0$) be $y^*(t_1)$. Then, we can represent the perturbations $\hat{x}(t_1, t_2)$ due to nonzero $\tilde{u}(t_2)$ as

$$\begin{aligned}\left(\frac{\partial}{\partial t_1} + \frac{\partial}{\partial t_2}\right) &q(y^*(t_1) + \hat{x}(t_1, t_2)) + f(y^*(t_1) + \hat{x}(t_1, t_2))\\ &= b^*(t_1) + b\tilde{u}(t_2).\end{aligned} \tag{60}$$

Expanding the nonlinear functions $q(\cdot)$ and $f(\cdot)$ in Taylor series, we obtain

$$\begin{aligned}\left(\frac{\partial}{\partial t_1} + \frac{\partial}{\partial t_2}\right) &\left[q(y^*(t_1)) + C_1(t_1)\hat{x} + C_2(t_1)\hat{x}^{②}\right.\\ &\left. + C_3(t_1)\hat{x}^{③} + \cdots\right] + f(y^*(t_1)) + G_1(t_1)\hat{x}\\ &+ G_2(t_1)\hat{x}^{②} + G_3(t_1)\hat{x}^{③} + \cdots = b^*(t_1) + b\tilde{u}(t_2).\end{aligned} \tag{61}$$

Here, $C_i(t_1)$ and $G_i(t_1)$ represent the time-varying ith derivative matrices of $q(\cdot)$ and $f(\cdot)$, respectively, evaluated about $y^*(t_1)$. Next, we express $\tilde{u}(t_2)$ as $\varepsilon u(t_2)$, with ε a small scalar parameter. The solution $\hat{x}(t_1, t_2)$ can now be expressed in a Taylor series in ε

$$\hat{x}(t_1, t_2) = \varepsilon\hat{x}_1(t_1, t_2) + \varepsilon^2\hat{x}_2(t_1, t_2) + \varepsilon^3\hat{x}_3(t_1, t_2) + \cdots. \tag{62}$$

Substituting (62) in (61), and collecting the coefficients of powers of ε, the following equations for $\hat{x}_1$, $\hat{x}_2$, $\hat{x}_3$, etc., are obtained:

$$\left(\frac{\partial}{\partial t_1} + \frac{\partial}{\partial t_2}\right)[C_1(t_1)\hat{x}_1] + G_1(t_1)\hat{x}_1 = bu(t_2) \tag{63}$$

$$\begin{aligned}\left(\frac{\partial}{\partial t_1} + \frac{\partial}{\partial t_2}\right)&[C_1(t_1)\hat{x}_2] + G_1(t_1)\hat{x}_2\\ &= -\left(\frac{\partial}{\partial t_1} + \frac{\partial}{\partial t_2}\right)\left[C_2(t_1)\hat{x}_1^{②}\right] - G_2(t_1)\hat{x}_1^{②}\end{aligned} \tag{64}$$

$$\begin{aligned}\left(\frac{\partial}{\partial t_1} + \frac{\partial}{\partial t_2}\right)&[C_1(t_1)\hat{x}_3] + G_1(t_1)\hat{x}_3\\ = -&\left(\frac{\partial}{\partial t_1} + \frac{\partial}{\partial t_2}\right)\\ &\cdot\left[C_3(t_1)\hat{x}_1^{③} + C_2(t_1)(\hat{x}_1 \otimes \hat{x}_2 + \hat{x}_2 \otimes \hat{x}_1)\right]\\ &- G_3(t_1)\hat{x}_1^{③} - G_2(t_1)(\hat{x}_1 \otimes \hat{x}_2 + \hat{x}_2 \otimes \hat{x}_1).\end{aligned} \tag{65}$$

Equation (63) can be expressed in the operator form already encountered before in (7)

$$\hat{X}(t_1, s) = \left(\frac{D}{dt_1}[] + sC(t_1) + G(t_1)\right)^{-1}[b]U(s). \tag{66}$$

As discussed in Section III-B, (66) can be reduced using the Arnoldi method to the form

$$\frac{\partial}{\partial t_2}\left[T_q x_{1,q}(t_2)\right] + x_{1,q}(t_2) = R_q u(t_2) \tag{67}$$

with

$$R_q = V_q^T \mathcal{R} \tag{68}$$

and

$$\hat{x}_1(t_1, t_2) \approx V_q(t_1)x_{1,q}(t_2). \tag{69}$$

R_q, $\mathcal{R}$, and V_q are as defined in Section III-B; $V_q(t_1)$ is defined as

$$V_q(t_1) = \sum_{i=-\infty}^{\infty} V_{q,i}e^{ji\omega_0 t_1} \tag{70}$$

with $V_{q,i}$ being the ith block-row of V_q, corresponding to the ith output harmonic or time point.

As in the time-invariant case, we approximate $\hat{x}_1^{②}$ as

$$\begin{aligned}\hat{x}_1 \otimes \hat{x}_1 &\approx (V_{q_1}(t_1)x_{1,q_1}) \otimes (V_{q_1}(t_1)x_{1,q_1})\\ &= (V_{q_1}(t_1) \otimes V_{q_1}(t_1))(x_{1,q_1} \otimes x_{1,q_1}).\end{aligned} \tag{71}$$

Now, (64) becomes

$$\begin{aligned}\left(\frac{\partial}{\partial t_1} + \frac{\partial}{\partial t_2}\right)&[C_1(t_1)\hat{x}_2] + G_1(t_1)\hat{x}_2\\ &= -\left(\frac{\partial}{\partial t_1} + \frac{\partial}{\partial t_2}\right)\left[(C_2(t_1)V_{q_1}^{②}(t_1))x_{1,q_1}^{②}(t_2)\right]\\ &\quad - \left(G_2(t_1)V_{q_1}^{②}(t_1)\right)x_{1,q_1}^{②}(t_2).\end{aligned} \tag{72}$$

Equation (72) can now be expressed in block-matrix form as

$$\begin{aligned}\left(\frac{\partial}{\partial t} + \frac{\partial}{\partial t_2}\right)&\left(\begin{bmatrix} C_1(t_1) & -I \\ 0 & 0\end{bmatrix}\begin{bmatrix}\hat{x}_2 \\ \hat{x}_{2e}\end{bmatrix}\right) + \begin{bmatrix} G_1(t_1) & 0 \\ 0 & I\end{bmatrix}\\ &\cdot\begin{bmatrix}\hat{x}_2 \\ \hat{x}_{2e}\end{bmatrix} \approx -\begin{bmatrix} G_2(t_1)V_{q_1}^{②}(t_1) \\ C_2(t_1)V_{q_1}^{②}(t_1)\end{bmatrix} x_{1,q_1}^{②}(t_2).\end{aligned} \tag{73}$$

$$\begin{aligned}\left(\frac{\partial}{\partial t_1} + \frac{\partial}{\partial t_2}\right)&\left(\begin{bmatrix} C_1(t_1) & -I \\ 0 & 0\end{bmatrix}\begin{bmatrix}\hat{x}_3 \\ \hat{x}_{3e}\end{bmatrix}\right) + \begin{bmatrix} G_1(t_1) & 0 \\ 0 & I\end{bmatrix}\begin{bmatrix}\hat{x}_3 \\ \hat{x}_{3e}\end{bmatrix}\\ \approx -&\begin{bmatrix} G_3(t_1)V_{q_1}^{③}(t_1) & (G_2(t_1)((V_{q_2}(t_1) \otimes V_{q_1}(t_1))P_{q_2,q_1} + V_{q_1}(t_1) \otimes V_{q_2}(t_1))) \\ C_3(t_1)V_{q_1}^{③}(t_1) & (C_2(t_1)((V_{q_2}(t_1) \otimes V_{q_1}(t_1))P_{q_2,q_1} + V_{q_1}(t_1) \otimes V_{q_2}(t_1)))\end{bmatrix} \cdot \begin{bmatrix} x_{1,q_1}^{③}(t_2) \\ x_{1,q_1}(t_2) \otimes x_{2,q_2}(t_2)\end{bmatrix}.\end{aligned} \tag{74}$$

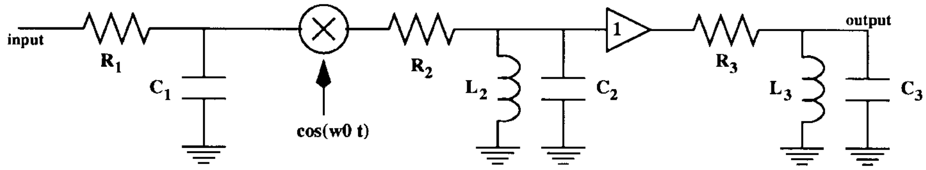


Fig. 3. Low-pass filter→mixer→two bandpass filters.

Equation (73) has q_1^2 inputs; it can therefore be reduced to a smaller system using the techniques of Section III-B for multiple inputs. Let the reduced size be q_2 and the corresponding Arnoldi subspace be $\tilde{V}_{q_2}$; define $V_{q_2} = [I_{Nn} \quad 0]\tilde{V}_{q_2}$. Following a procedure similar to that for obtaining (73), (65) can also be expressed in matrix form as shown in (74). Equation (74), shown at the bottom of the previous page, is an LTV system with $q_1^3 + q_1 q_2$ inputs, which can, in turn, be reduced to a smaller one (of size q_3) using the techniques of Section III-B.

V. Macromodeling Cyclostationary Noise

When a system is macromodeled, it is also desirable to replace all its noise contributions by a few equivalent noise sources at the inputs or outputs.[3] Usually, the power spectra of the equivalent sources have complicated frequency-dependence, unlike those of the relatively simple white and flicker noise models typically used for internal noise generators. At the macromodel level, representing this frequency dependence perfectly requires computations with the original system, thus defeating the purpose of macromodeling. Instead, it is preferable to find approximate, but computationally inexpensive, forms of this frequency dependence. Such a capability has already been obtained for LTI systems with stationary noise [13], [14]. In this section, we sketch the extension to cyclostationary noise in LTV systems, useful for capturing phenomena such as frequency-translation and mixing of noise. The extension is achieved by applying the noise reformulation technique in [13], [14] to a block-matrix relation for cyclostationary noise [35] to obtain the form (30), and then applying TVP.

We first recall the cyclostationary noise block-matrix relation [35]

$$S_{xx}(s) = [\mathcal{J}FD + s\mathcal{C}_{FD}]^{-1}\mathcal{A}S_{uu}(s)\mathcal{A}T[\mathcal{J}_{FD} + s\mathcal{C}_{FD}]^{-*} \tag{75}$$

where $\mathcal{A}$ is the incidence matrix of the systems internal noise sources,[4] S_{uu} is a block matrix of HPSDs (harmonic power spectral densities) from internal noise sources, and S_{xx} is the block matrix of noise HPSDs within the system, including the outputs. Analogous to (19) and without loss of generality, we can select the HPSDs at a single output by

$$\begin{aligned}\vec{F}(s) &= \mathcal{D}^T S_{xx}(s)\mathcal{D}\\ &= \mathcal{D}^T[\mathcal{J}_{FD} + s\mathcal{C}_{FD}]^{-1}\mathcal{A}S_{uu}(s)\mathcal{A}^T[\mathcal{J}_{FD} + s\mathcal{C}_{FD}]^{-*}\mathcal{D}.\end{aligned} \tag{76}$$

[3] Input- and output-referred noise sources are used extensively in circuit design.

[4] Not to be confused with $\mathcal{A}$ in (30).

Equation (76) is structurally similar to (21) in [14], with l replaced by the rectangular matrix $\mathcal{D}$. It is straightforward to apply the same reformulation steps as for LTI noise [14] to bring (76) to the form of (30), i.e.,

$$\vec{F}(s) = \tilde{\mathcal{D}}^T[s\tilde{\mathcal{C}} + \tilde{\mathcal{J}}]^{-1}\tilde{\mathcal{D}}. \tag{77}$$

TVP can now be applied to (77) to obtain a much smaller set of equations in the form of (36), which can be used to compute the noise contributions of the macromodeled system.

VI. Applications of TVP

In this section, we present four applications of TVP. The first application is to a small idealized example, for the purpose of verifying TVP against hand calculations. The second application is to a switched capacitor integrator block. The third is to a RF mixer subsystem from the Lucent W2030 RFIC chip. The final application is to a dc/dc power conversion system.

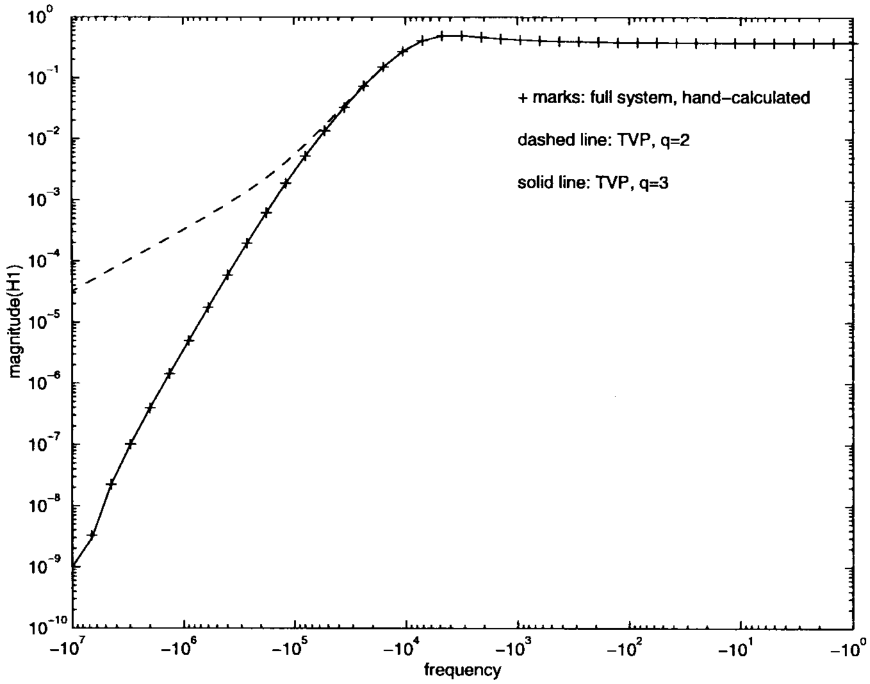
A. A Hand-Calculable Example

Fig. 3 depicts an upconverter, consisting of a low-pass filter, an ideal mixer, and two bandpass filter stages. The component values were chosen to be: $R_1 = 160\ \Omega$, $R_2 = 1.6$ kΩ, $R_3 = 500\ \Omega$, $C_1 = C_2 = C_3 = 10$ nF, and $L_2 = L_3 = 25.35$ nH. These values result in a low-pass filter with a pole at 100 kHz, and bandpass filters with a center frequency of 10 MHz and bandwidths of about 10 and 30 kHz, respectively. The LO frequency for the mixer was chosen to be 10 MHz.

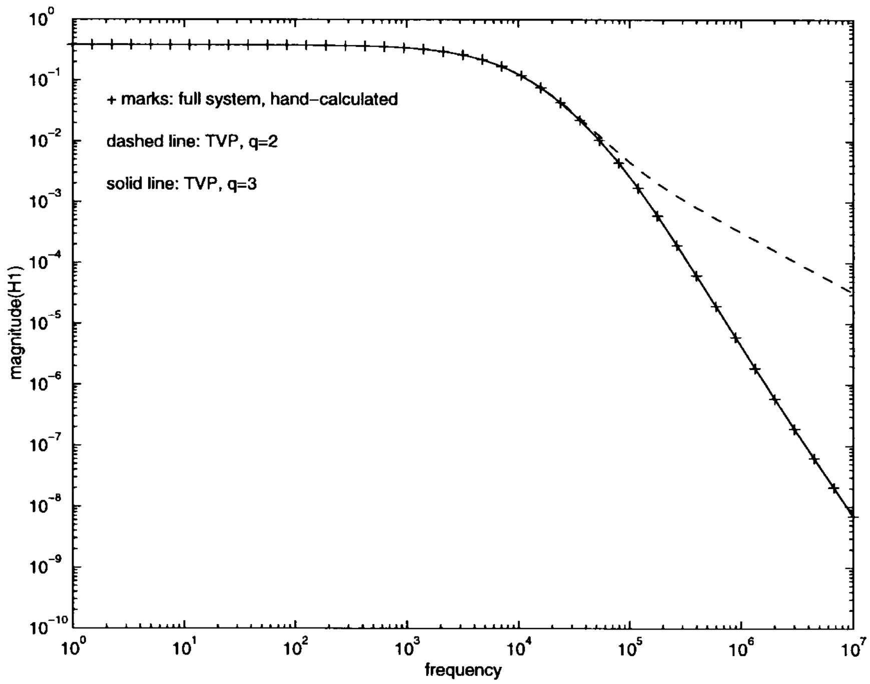
With reference to (17), the baseband-referred transfer functions of interest in this case are $H_1(s)$ and $H_{-1}(s)$, since they appear in the desired up- and down-conversion paths. It can be shown that $H_{-i}(s) = H_i^*(-s)$; hence, it suffices to consider only $H_1(s)$ here. The expression for $H_1(s)$ can be derived easily using intuitive frequency-translation concepts; it is

$$H_1(s) = \frac{0.5}{1 + sC_1R_1}\,\frac{\dfrac{(s+\jmath\omega_0)L_2}{1+(s+\jmath\omega_0)^2L_2C_2}}{R_2 + \dfrac{(s+\jmath\omega_0)L_2}{1+(s+\jmath\omega_0)^2L_2C_2}} \cdot \frac{\dfrac{(s+\jmath\omega_0)L_3}{1+(s+\jmath\omega_0)^2L_3C_3}}{R_3 + \dfrac{(s+\jmath\omega_0)L_3}{1+(s+\jmath\omega_0)^2L_3C_3}}. \tag{78}$$

Equation (78) is plotted for positive and negative frequencies in Fig. 4. Also plotted are the transfer functions obtained from TVP with $q = 2$ and $q = 3$. It can be seen that for $q = 2$, TVP produces a reasonable approximation, whereas for $q = 3$, the



(a)



(b)

Fig. 4. Simple circuit: $H_1(s)$ from TVP versus hand calculations. (a) $-$ve frequencies. (b) $+$ve frequencies.

match is perfect, even though the original system is of order five.

The poles of the original system and those from TVP are shown in Table I.

TABLE I
POLES (Hz) OF $H_1(s)$, ORIGINAL AND REDUCED SYSTEMS

Original system	TVP, $q = 2$	TVP, $q = 3$
-4.98e3 - j3.88e3	-4.96e3 - j3.91e3	-4.98e3 - j3.88e3
-1.59e4 - j3.90e3	-1.56e4 - j3.49e3	-1.59e4 - j3.90e3
-9.95e4		-9.95e4
-5.72e4 - j2.00e7		
-3.59e4 - j1.99e7		

B. Switched Capacitor Integrator Block

Our second application of TVP is to a lossy switched-capacitor integrator block. The circuit was designed in a 0.35-μ

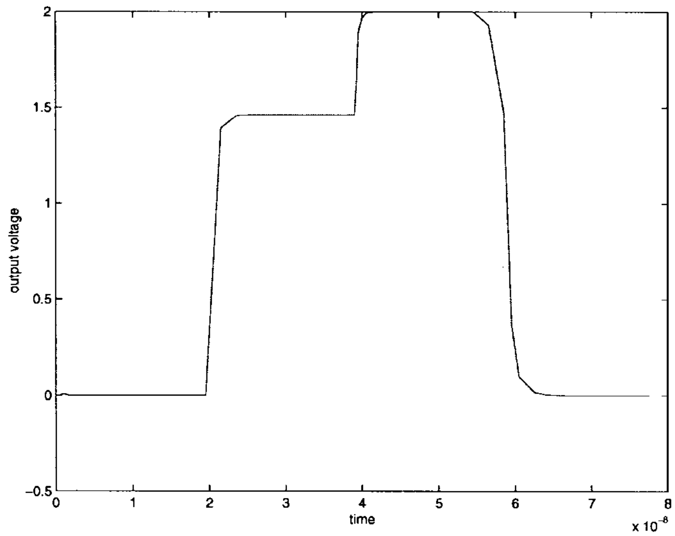


Fig. 5. Steady-state output of a switched-capacitor integrator (with zero input).

CMOS process, and modeled using a Lucent MOS model (ASIM3) specifically intended for high-accuracy analog simulations. Comprising more than 150 MOS devices, it includes biasing, common mode feedback and slew-rate enhancement sections.

The clock signal to the switched-capacitor filter had a time period of 78 ns (i.e., frequency $=$ 12.8 MHz), but some sections of the circuit operated at twice that frequency, i.e., 25.6 MHz. The steady-state waveform of the output node (in the absence of signal input) was obtained using shooting and is shown in Fig. 5.

The output node did not have switching activity filtered out. Fig. 6 depicts a multi-time scale plot of the waveform at the output node in the presence of a 10-kHz sinusoidal input. (For details on how to interpret multi-time plots of waveforms, see [31] and [34]. The signal envelope (riding on the switching variations) is obtained directly from the waveform along a cross-section parallel to the signal time scale. By shifting the point of cross-section to along the clock time-scale, the signal envelope at different points of the clock waveform can be seen. Note how the (sinusoidal) signal is transmitted in the region between 60 and 78 ns on the clock time scale, but is cut out (because switches are off) between about 0–20 and 40–60 ns. For macromodeling, we chose to sample the output at 70 ns on the clock time scale, i.e., in the middle of the clock phase in which the signal is being transmitted. In other words, the transfer function being modeled is that between the input and the waveform obtained by taking a cross-section, parallel to the signal time scale, at 70 ns on the clock time scale in Fig. 6.

A time-domain version of TVP was applied to reduce this transfer function. The macromodeling algorithm was run up to order 25. Fig. 7 depicts the input-to-output transfer functions from the full system ($\times$ marks), as well as from two macromodels of size $q = 3$ (dashed line) and $q = 25$ (solid line). As can be seen, even a tiny behavioral model of size 3 is sufficient to capture the response for input frequencies up to almost the switching frequency, while the size 25 model is accurate up to well beyond.

The poles and residues of the system were obtained by eigendecomposition of the T_q matrices, and used to construct expressions for the transfer function from the signal input to the signal output envelope. For $q = 3$, this resulted in the following analytical expression for the transfer function from the input to the output envelope:

$$H_{0,\,q=3}(f) = \frac{0.613}{\jmath 2\pi f - (-1.1e6)} + \frac{1.02e-4}{\jmath 2\pi f - (-1.68e5)} + \frac{9.81e-3}{\jmath 2\pi f - (-1.2e9)}. \tag{79}$$

From the fact that the poles have negative real parts, it is seen that the system is stable. Further, we also observe that the smallest pole (168 kHz) has a much smaller residue than the one at 1.1 MHz. Such expressions can be useful to incorporate the precise characteristics of real circuit blocks into simple spreadsheet-type system design tools. Note that this is a LTI macromodel that abstracts the underlying continuous filter from the switching. If detail about the effects of switching is desired in the macromodel, all the timepoints along the clock cycle need to be incorporated as outputs to TVP.

C. RF Buffer and Mixer Block

A portion of the W2013 RFIC from Lucent Microelectronics, consisting of an I-channel buffer and mixer, was reduced by TVP. The circuit consisted of about $n = 360$ nodes, and was excited by a local oscillator at 178 MHz driving the mixer, while the RF input was fed into the I-channel buffer. The time-varying system was obtained around a steady state of the circuit at the oscillator frequency; a total of $N = 21$ harmonics were considered for the time-variation.

Fig. 8 shows frequency plots of $H_1(s)$, the upconversion transfer function. The points marked "$+$" were obtained by direct computation of (17), while the lines were computed using the TVP-reduced models with $q = 2$ and $q = 10$, respectively. Even with $q = 2$, a size reduction of two orders of magnitude, the reduced model provides a good match up to the LO frequency. When the order of approximation is increased to ten, the reduced model is identical up to well beyond the LO frequency. Evaluating the reduced models was more than *three orders of magnitude faster* than evaluating the transfer function of the original system.

The poles of the reduced models for $H_1(s)$, easily calculated on account of their small size, are shown in Table II. These are useful in design because they constitute excellent approximations of the full system's poles, which are difficult to determine otherwise.

D. PWM DC/DC Converter

Our final application of TVP is to a boost-type dc/dc converter, featuring PWM feedback for output voltage stabilization. A simplified diagram of the circuit is shown in Fig. 9. When the switch closes, the inductor current rises linearly until the switch opens, after which the current is diverted through the diode into the load resistor. The peak current of the inductor is related to the amount of time the switch is

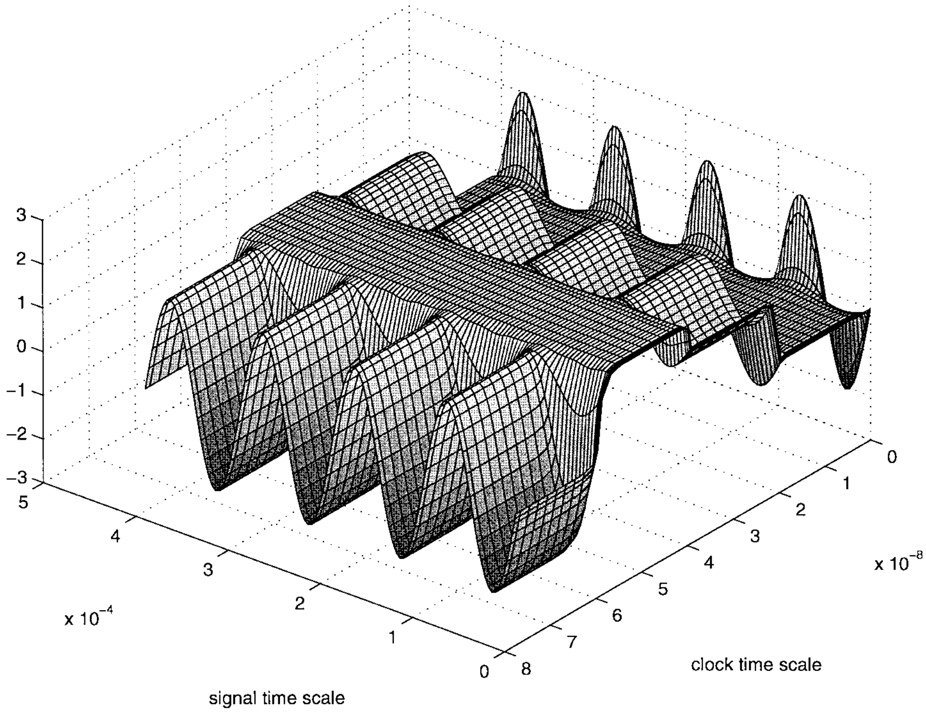


Fig. 6. Multitime plot of switched-capacitor output.

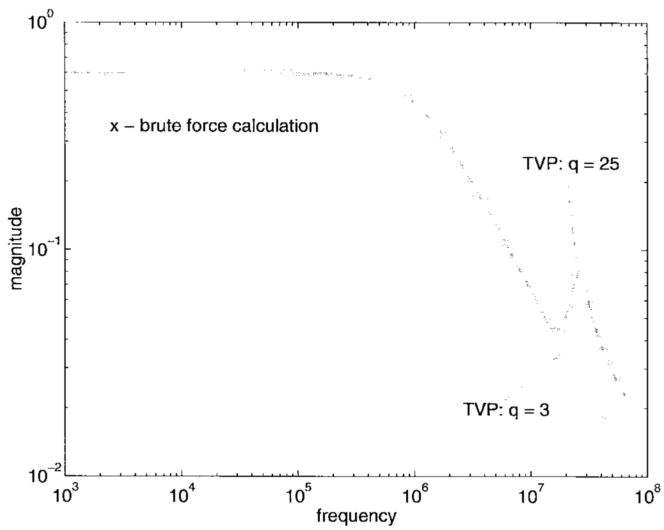


Fig. 7. Frequency response of a switched-capacitor filter.

TABLE II
POLES OF $H_1(s)$ FOR THE I-CHANNEL BUFFER/MIXER

TVP, $q=2$	TVP, $q=10$
-5.3951e+06	-5.3951e+06
-6.9196e+07 - j3.0085e+05	-9.4175e+06
	-1.5588e+07 - j2.5296e+07
	-1.5588e+07 + j2.5296e+07
	-6.2659e+08 - j1.6898e+06
	-1.0741e+09 - j2.2011e+09
	-1.0856e+09 + j2.3771e+09
	-7.5073e+07 - j1.4271e+04
	-5.0365e+07 + j1.8329e+02
	-5.2000e+07 + j7.8679e+05

closed, i.e., the duty cycle of the switch control. This peak current determines the maximum output voltage, at node 3.

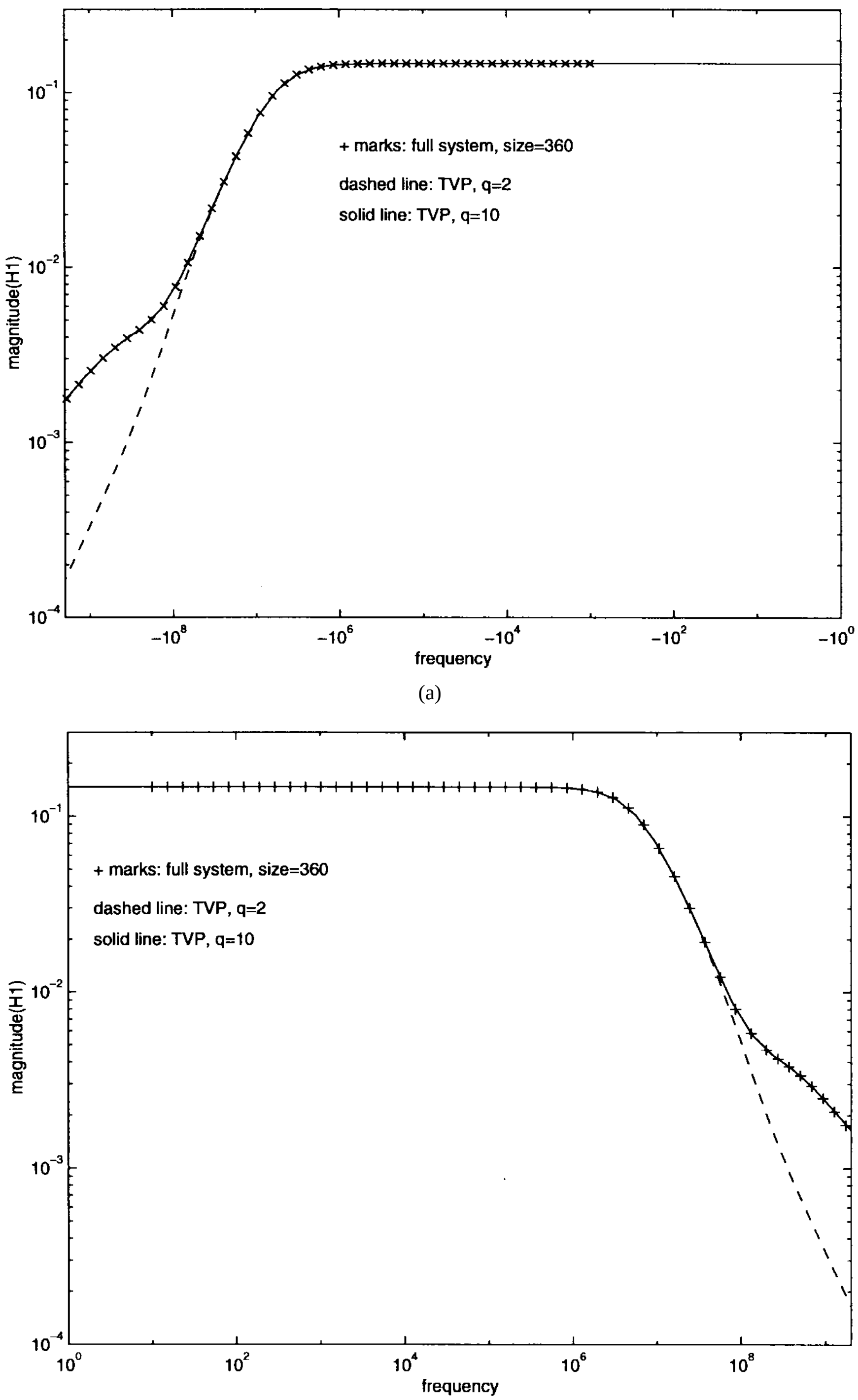
The negative feedback loop operates by comparing the output voltage at node 3 with a reference to obtain an error voltage, which is used to control the duty cycle of the control to the switch. If the output voltage is lower than the reference, the duty cycle is increased, and vice versa.

The nominal value of the input power source E was set at 1 V, while the reference voltage for the output was set to 1.4 V. The switching rate was 100 kHz. The resistance–capacitance (*RC*) pole formed at the load was at about 20 Hz.

Initially, the loop gain including the PCM unit was set to ten. The steady state of the system was obtained with shooting using about 100 timepoints. TVP (using time-domain steady-state matrices) was then run for ten steps. Fig. 10 shows plots of the transfer-function[5] from the input source E to the regulated voltage at node 3. The $\times$ marks were obtained from the full system, while the dashed and solid lines are from evaluations of the TVP-generated macromodels, as indicated. Observe that the size-4 macromodel is adequate to capture the system's behavior up to the switching frequency. From the plots, we note that for low frequencies, the ripple rejection of the system is of the same order as the loop gain. The rejection, however, deteriorates significantly as the frequency rises; in fact, a small *gain* is seen at about 80 Hz.

The transfer function corresponding to the $q = 4$ macromodel (using poles and residues obtained by eigendecompo-

[5] This is the 0th-harmonic transfer function, i.e., the average over the clock time scale.



(a)

(b)

Fig. 8. I-channel mixer $H_1(s)$: reduced versus full system.

sition) is

$$H_{0,\,q=4}(f) = \frac{-0.147 \pm 1.1j}{j2\pi f - (-24.66 \pm 38.36j)} + \frac{0.0366}{j2\pi f - (-250.74)} + \text{small term.} \quad (80)$$

Note that the real parts of the poles are negative, indicating a stable system.

To improve the supply rejection of the converter, the loop gain was increased to 1000, the steady-state recomputed using shooting, and TVP macromodels generated again. The new transfer plots are shown in Fig. 11. Note that, as expected, the rejection at dc has improved to a factor of about 1000. However, the TVP-generated analytic transfer function (for $q = 4$) is now

$$H_{0,\,q=4}(f) = \frac{-0.0124 \pm 0.0455j}{j2\pi f - (+80.32 \pm 773.4j)} + \frac{0.0239}{j2\pi f - (-2854.9)} + \text{small term.} \quad (81)$$

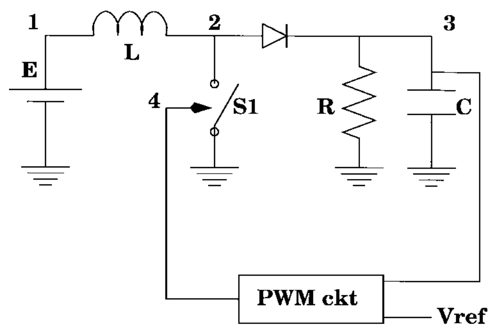


Fig. 9. A dc/dc switching power converter with PWM feedback.

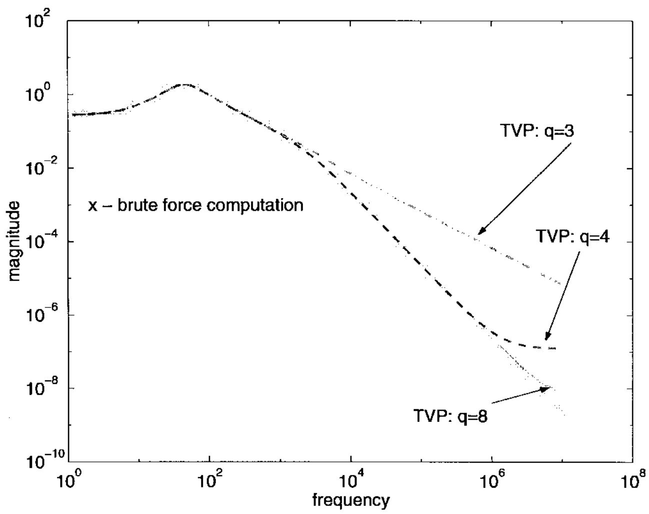


Fig. 10. A dc/dc converter: transfer function for loop-gain 10.

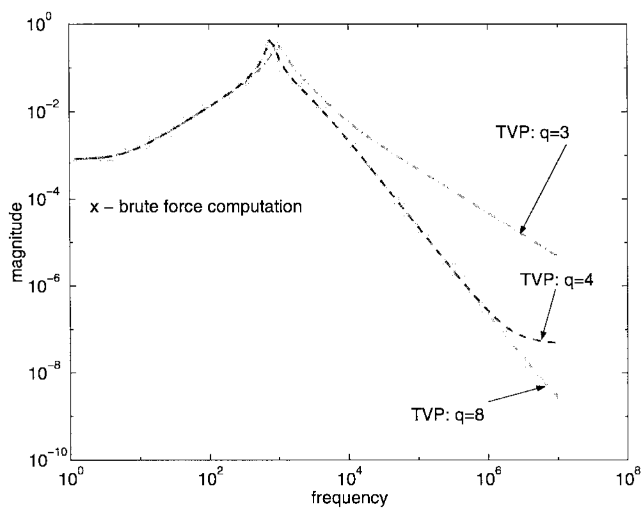


Fig. 11. A dc/dc converter: transfer function for loop-gain 1000.

Note that the complex pole pair now has a *positive* real part, showing that the system is in fact unstable. The instability is generated by a combination of excessive phase shift and gain in the PWM feedback look. Using TVP-generated macromodels, numerical values of such unstable poles are easily obtained. Note that steady-state methods like shooting and harmonic balance, on which TVP relies, are indeed able to find unstable periodic solutions, because they solve boundary-value problems rather than initial-value ("transient") problems.

VII. Conclusion

We have presented the TVP algorithm for reducing large LTV systems to much smaller ones with similar input–output transfer characteristics. The method is useful for automatic generation of accurate macromodels from SPICE-level descriptions, especially of communication system blocks. TVP has applications in system-level verification, producing analytical expressions for transfer functions, and intellectual property protection. We have illustrated TVP with several examples and obtained size reductions and computational speedups of orders of magnitude without loss of accuracy. We have also described extensions of TVP to incorporate signal path nonlinearities and for cyclostationary noise macromodeling.

Appendix
Floquet Parameters and LPTV Transfer Functions

It is well known that any LPTV system can be reduced to an LTI system and memoryless time-varying transformations. This result from Floquet theory (e.g., [10], [20]) implies that any LPTV system has n modes associated with it, the so-called Floquet parameters, corresponding to the eigenvalues of the underlying LTI system. In this section, we clarify the relationship between the Floquet parameters and the time-varying transfer function of the LPTV system.

We start from the following ordinary differential equation description of a linear periodic time-varying system[6]

$$\frac{d}{dt}x(t) = G(t)x(t) + bu(t) \tag{82}$$
$$z(t) = d^T x(t) \tag{83}$$

where $G(t)$ is periodic with period T_1. Floquet theory [10], [20] states that there exists a nonsingular T_1-periodic matrix $P(t)$ and a constant diagonal matrix D, such that (82) is equivalent to

$$\begin{aligned} \tilde{u}(t) &= P(t)^{-1}bu(t) \\ \frac{d}{dt}y(t) &= Dy(t) + \tilde{u}(t) \\ x(t) &= P(t)y(t). \end{aligned} \tag{84}$$

Hence, we obtain a system equivalent to (82) and (83)

$$\begin{aligned} \frac{d}{dt}y(t) &= Dy(t) + \underbrace{P(t)^{-1}bu(t)}_{r(t)} \\ z(t) &= \underbrace{d^T P(t)}_{l^T(t)} y(t). \end{aligned} \tag{85}$$

Equation (85) can be recognized to be an LTI system with the inputs and outputs multiplied by the periodic time-varying quantities $r(t)$ and $l^T(t)$. Since D is diagonal, the equations

[6]The general case of LPTV DAEs can be addressed similarly using Floquet theory for DAEs [8].

are decoupled into modes. The entries of D are the Floquet parameters. Following a procedure similar to that in Section III, the time-varying transfer function for (85) can be shown to be

$$H(t_1, s) = l^T(t_1)\left(\frac{d}{dt_1} + sI - D\right)^{-1}[r(t_1)]. \tag{86}$$

Equation (86) can be solved explicitly, because D is diagonal and time-invariant. The solution with T_1-periodic boundary conditions can be shown to be

$$H(t_1, s) = \sum_{m=-\infty}^{\infty} \underbrace{\left(\sum_{k=-\infty}^{\infty} L_{m-k}^T[s + j\omega_0 k - D]^{-1}R_k\right)}_{H_m(s)} \cdot e^{jm\omega_0 t_1} \tag{87}$$

where L_i and R_i are the Fourier coefficients of $l(t)$ and $r(t)$, respectively. Let the diagonal elements of D be ρ_i. Then $H_m(s)$ in (87) can be written as

$$H_m(s) = \sum_{k=-\infty}^{\infty} \sum_{i=1}^{n} \left(\frac{L_{(m-k),\,i}R_{k,\,i}}{s + j\omega_0 k - \rho_i}\right) \tag{88}$$

where $L_{(m-k),\,i}$ and $R_{k,\,i}$ are the ith elements of L_{m-k} and R_k, respectively.

Equation (88) shows that for each m, $H_m(s)$ can have an infinite number of poles, which are simply the Floquet parameters shifted by multiples of $j\omega_0$. Moreover, it is clear that these poles are not time-varying. When (88) is put into (87), it is also evident that the residues $L_{(m-k),\,i}L_{k,\,i}e^{jm\omega_0 t_1}$ of $H(t_1, s)$ are, in fact, time varying.

ACKNOWLEDGMENT

The author would like to thank A. Demir for many fruitful discussions, particularly on nonlinear extensions of TVP. D. Long provided assistance and code for the frequency-domain implementation. T.-F. Fang and J. Thottuvelil pointed out the usefulness of TVP for switched-capacitor filters, power conversion circuits and IP protection. The author would also like to acknowledge the LTI model-reduction community in general, and his colleagues P. Feldmann and R. Freund in particular, for developing effective methods for model-reduction and establishing it as a useful tool for circuit analysis.

REFERENCES

[1] J. Aliaga, D. Boley, R. Freund, and V. Hernandez, "A Lanczos-type algorithm for multiple starting vectors," Bell Labs, Murray Hill, NJ, Numerical Analysis no. 96-18, 1996.
[2] D. L. Boley, "Krylov space methods on state-space control models," *IEEE Trans. Circuits Syst. II,* vol. 13, pp. 733–758, June 1994.
[3] H. G. Brachtendorf, G. Welsch, R. Laur, and A. Bunse-Gerstner, "Numerical steady state analysis of electronic circuits driven by multi-tone signals," in *Electrical Engineering.* New York: Springer-Verlag, 1996, vol. 79, pp. 103–112.
[4] M. Celik and A. C. Cangellaris, "Simulation of dispersive multiconductor transmission lines by Padé approximation by the Lanczos process," *IEEE Trans. Microwave Theory Tech.,* vol. 44, pp. 2525–2535, 1996.
[5] ——, "Simulation of multiconductor transmission lines using Krylov subspace order-reduction techniques," *IEEE Trans. Computer-Aided Devices,* vol. 16, pp. 485–496, 1997.
[6] E. Chiprout and M. S. Nakhla, *Asymptotic Waveform Evaluation.* Norwell, MA: Kluwer, 1994.
[7] L. O. Chua and P.-M. Lin, *Computer-Aided Analysis of Electronic Circuits: Algorithms and Computational Techniques.* Englewood Cliffs, NJ: Prentice-Hall, 1975.
[8] A. Demir, "Floquet theory and nonlinear perturbation analysis for oscillators with differential-algebraic equations," in *Proc. ICCAD,* 1998.
[9] I. Elfadel and D. Ling, "A block rational Arnoldi algorithm for multipoint passive model-order reduction of multiport *RLC* networks," in *Proc. ICCAD,* Nov. 1997, pp. 66–71.
[10] M. Farkas, *Periodic Motions.* New York: Springer-Verlag, 1994.
[11] P. Feldmann and R. Freund, "Efficient linear circuit analysis by Pade approximation via the Lanczos process," *IEEE Trans. Computer-Aided Design,* vol. 14, pp. 639–649, May 1995.
[12] ——, "Reduced-order modeling of large linear subcircuits via a block Lanczos algorithm," in *Proc. IEEE DAC,* 1995, pp. 474–479.
[13] ——, "Circuit noise evaluation by Padé approximation based model-reduction techniques," in *Proc. ICCAD,* Nov. 1997, pp. 132–138.
[14] ——, "Circuit noise evaluation by Padé approximation based model-reduction techniques," Bell Laboratories, Murray Hill, NJ, Tech. Rep. ITD-97-31678G, 1997.
[15] P. Feldmann, R. C. Melville, and D. Long, "Efficient frequency domain analysis of large nonlinear analog circuits," in *Proc. IEEE CICC,* May 1996, pp. 461–464.
[16] R. Freund, "Reduced-order modeling techniques based on Krylov subspaces and their use in circuit simulation," Bell Labs, Tech. Rep. 11273-980217-02TM, 1998.
[17] R. Freund, G. H. Golub, and N. M. Nachtigal, "Iterative solution of linear systems," *Acta Numerica,* pp. 57–100, 1991.
[18] K. Gallivan, E. Grimme, and P. Van Dooren, "Asymptotic waveform evaluation via a lanczos method," *Appl. Math. Lett.,* vol. 7, pp. 75–80, 1994.
[19] E. J. Grimme, "Krylov projection methods for model reduction," Ph.D. dissertation, Elect. Eng. Dept., Univ. Illinois at Urbana-Champaign, 1997.
[20] R. Grimshaw, *Nonlinear Ordinary Differential Equations.* Oxford, U.K.: Blackwell, 1990.
[21] X. Huang, V. Raghavan, and R. A. Rohrer, "AWEsim: A program for the efficient analysis of linear(ized) circuits," in *Proc. ICCAD,* Nov. 1990, pp. 534–537.
[22] I. M. Jaimoukha, "A general minimal residual Krylov subspace method for large-scale model reduction," *IEEE Trans. Automat. Contr.,* vol. 42, pp. 1422–1427, 1997.
[23] T. Lenahan, "Analysis of linear periodically time-varying (LPTV) systems," Bell Laboratories, Tech. Rep. ITD-97-32813Q, Nov. 1997.
[24] R. C. Melville, P. Feldmann, and J. Roychowdhury, "Efficient multi-tone distortion analysis of analog integrated circuits," in *Proc. IEEE CICC,* pp. 241–244, May 1995.
[25] S. Narayanan, "Transistor distortion analysis using volterra series representation," *Bell System Tech. J.,* May/June 1967.
[26] A. Odabasioglu, M. Celik, and L. T. Pileggi, "PRIMA: Passive reduced-order interconnect macromodeling algorithm," in *Proc. ICCAD,* Nov. 1997, pp. 58–65.
[27] J. Phillips, "Model reduction of time-varying linear systems using approximate multipoint Krylov-subspace projectors," in *Proc. ICCAD,* Nov. 1998, pp.
[28] L. T. Pillage and R. A. Rohrer, "Asymptotic waveform evaluation for timing analysis," *IEEE Trans. Computer-Aided Design,* vol. 9, pp. 352–366, Apr. 1990.
[29] M. Rösch and K. J. Antreich, "Schnell stationäre simulation nichtlinearer Schaltungen im Frequenzbereich," *AEÜ,* vol. 46, no. 3, pp. 168–176, 1992.
[30] J. Roychowdhury, "A unifying formulation for analysing multi-rate circuits," in *Proc. NSF-IMA Workshop on Algorithmic Methods for Semiconductor Circuitry,* 1997.
[31] ——, "Efficient methods for simulating highly nonlinear multi-rate circuits," in *Proc. IEEE DAC,* 1997.
[32] ——, "MPDE methods for efficient analysis of wireless systems," in *Proc. IEEE CICC,* 1998, pp. 451–454.
[33] ——, "Reduced-order modeling of linear time-varying systems," in *Proc. ICCAD,* 1998, pp. 53–61.
[34] ——, "Analyzing circuits with widely-separated time scales using numerical PDE methods," *IEEE Trans. Circuits Syst. I,* vol. 46, Sept. 1999.
[35] J. Roychowdhury, D. Long, and P. Feldmann, "Cyclostationary noise analysis of large RF circuits with multitone excitations," *IEEE J. Solid-State Circuits,* vol. 33, pp. 324–336, Mar. 1998.
[36] J. S. Roychowdhury, "SPICE3 Distortion analysis," M.S. thesis, Elect. Eng. Comput. Sci. Dept., Univ. Calif. Berkeley, Elec. Res. Lab.,

Memorandum no. UCB/ERL M89/48, Apr. 1989.
[37] Y. Saad, *Iterative Methods for Sparse Linear Systems.* Boston, MA: PWS, 1996.
[38] R. Telichevesky, K. Kundert, and J. White, “Efficient steady-state analysis based on matrix-free Krylov subspace methods,” in *Proc. IEEE DAC,* pp. 480–484, 1995.
[39] ——, “Efficient AC and noise analysis of two-tone RF circuits,” in *Proc. IEEE DAC,* pp. 292–297, 1996.
[40] V. Volterra, *Theory of Functionals and of Integral and Integro-Differential Equations.* New York: Dover, 1959.

Jaijeet Roychowdhury received the B.Tech degree from IIT Kanpur in 1987, and the M.S. and Ph.D. degrees from the University of California at Berkeley in 1989 and 1993, all in electrical engineering.

From 1993 to 1995, he was with the CAD Laboratory of AT&T Bell Labs, Allentown, PA. Since 1995, he has been with the Communications Research Division of Bell Labs, Murray Hill, NJ. His research interests include design and analysis of communication systems and circuits.

Dr. Roychowdhury was awarded Distinguished Paper at ICCAD’91, and Best Papers at DAC’97, ASP-DAC’97, and ASP-DAC’99.