

Robust, Stable Time-Domain Methods for Solving MPDEs of Fast/Slow Systems

Ting Mei, Jaijeet Roychowdhury, *Member, IEEE*, Todd S. Coffey, Scott A. Hutchinson, and David M. Day

Abstract—In this paper, we explore the stability properties of time-domain numerical methods for multitime partial differential equations (MPDEs) in detail. We demonstrate that simple techniques for numerical discretization can lead easily to instability. By investigating the underlying eigenstructure of several discretization techniques along different artificial time scales, we show that not all combinations of techniques are stable. We identify choices of discretization method and step size, along fast and slow time scales, that lead to robust, stable time-domain integration methods for the MPDE. One of our results is that applying overstable methods along one time-scale can compensate for unstable discretization along others. Our novel integration schemes bring robustness to time-domain MPDE solution methods, as we demonstrate with examples.

Index Terms—Discretization, eigenstructure, envelope, multi-time partial differential equation (MPDE), stability, time-domain.

I. INTRODUCTION

MANY signals in analog, mixed-signal, and RF circuits contain both “fast” and “slow” components, i.e., they exhibit widely-separated time scales of variation. Examples of circuits that feature such signals include: mixers, phase-locked loops, voltage-controlled oscillators, automatic gain-control circuits, microwave amplifiers, etc. Simulating such circuits with conventional SPICE-like integration methods is usually computationally expensive, since time-steps must always be made small enough to accurately capture the fastest-varying component(s) in the solution. Since the information content of interest is typically in the slowest component (often termed the *envelope*), a large number of time-steps can be required. Performing such long simulations can be impractical, especially when many simulations are embedded within a comprehensive design methodology.

Recently, a class of time-domain techniques, arising from multitime partial differential equation (MPDE) circuit formulations [1]–[3], have shown promise in providing large simulation

speedups for such problems. In the MPDE formulation, multiple artificial time variables are introduced to decouple components with different rates of variation. This leads to a compact representation of the multiple-rate signals, with each disparate signal being represented by “its own” artificial time scale. By this transformation, it has been shown (e.g., [3]) that the circuit equations, originally ordinary differential equations (ODEs) or differential-algebraic equations (DAEs), are transformed into partial differential equation form, resulting in the MPDE. The utility of this transformation is that, when the circuit features signals with widely separated time scales with underlying periodic structure, discretizing and solving the MPDE numerically can, in fact, be far more efficient than solving the original circuit equations directly. Indeed, the MPDE is capable of generating different types of solutions by appropriate choice of boundary conditions [3], including quasiperiodic (also known as multi-tone) as well as *envelope* solutions, to be discussed later.

However, it has been observed empirically that there are often robustness issues related to numerical solution of the MPDE. For example, as illustrated in Section III, simple circuits can exhibit spurious oscillatory behavior when solved by MPDE discretization. While progress has been made on understanding MPDE robustness for *Fourier-envelope* methods (see Section II), the causes of undesirable numerical behavior for purely time-domain methods, and approaches to prevent them, have not been exposed to date. Clarifying these issues is crucial for enabling practical application of time-domain MPDE methods. Purely time-domain methods are necessary for solving strongly nonlinear circuits, since Fourier approaches are not best suited for representing strong nonlinearities efficiently.

In this paper, we explore the underlying cause of the time-domain MPDE robustness problem by investigating the eigenstructure of the discretized MPDE system. We find that time-domain discretization introduces fast-frequency components in the eigenvalues of the discretized system. This results in a change of stability properties from that of the original DAE.

We show that instability can be easily obtained for certain combinations of discretizations. We look into such stability changes in two steps. First, the stability impact of discretizing the fast time scale, and then, the additional effect of discretizing the slow time scale. We show that, in some cases, spurious instability introduced by the first discretization can be removed by the second, provided relative step-sizes are appropriately chosen. We also present a number of discretization schemes for which MPDE time-domain discretizations are stable, regardless of step size.

We thus achieve time-domain MPDE integration methods that work in a robust, predictable and stable manner. We

Manuscript received January 21, 2004; revised April 19, 2004. The work of T. Mei was supported by Sandia National Laboratories. The work of J. Roychowdhury was supported in part by Sandia National Laboratories, in part by the National Science Foundation under Award CCR-0312079 and Award CCR-0204278), in part by the Semiconductor Research Corporation, and in part by the Defense Advanced Research Projects Agency. This paper was recommended by Associate Editor L. M. Silveira.

T. Mei and J. Roychowdhury are with the Electrical and Computer Engineering Department, University of Minnesota, Minneapolis, MN 55455 USA (e-mail: meiting@ece.umn.edu).

T. S. Coffey, S. A. Hutchinson, and D. M. Day are with Sandia National Laboratories, Albuquerque, NM 87185 USA.

Digital Object Identifier 10.1109/TCAD.2004.841073

apply our novel methods to a down-conversion mixer and a fully differential opamp. The methods enable us to deliver the speedups, over traditional time-stepping DAE methods, that constitute one of the attractions of MPDEs.

The remainder of the paper is organized as follows. In Section II, we provide a brief discussion of relevant previous work. In Section III, we demonstrate the stability problem of time-domain MPDE techniques with certain discretization choices. This is followed by Section IV, where we develop the stability analysis of several combinations of discretization and integration methods, resulting in some predictably stable time-domain MPDE methods. In Section V, we apply our methods to circuits, demonstrating robustness and speedups.

II. RELEVANT PREVIOUS WORK

A number of efficient techniques are available for the numerical analysis of fast/slow systems. An underlying task that is basic to most such techniques is the determination of the (quasi-)periodic steady state of a given system. The techniques of shooting, finite-difference time-domain (FDTD) analysis, and harmonic balance (HB) are most commonly used for finding the steady state. Steady-state techniques are often used to handle fast components of a fast/slow waveform; building on these, variants of transient integration are applied to handle the slow components efficiently.

A. Steady-State Analysis

Finding the steady state of analog and mixed-signal circuits is an important task in itself as a means of determining distortion and intermodulation performance. Furthermore, as mentioned above, steady-state calculation methods often underlie more powerful slow/fast simulation techniques, such as those used to compute slow envelopes. Several methods are available for computing the periodic steady state. One such technique is the shooting algorithm (e.g., [4], [5]).

Shooting solves for an initial condition that leads to a perfectly periodic solution. Specifically, it starts with a guess for the initial condition, then integrates the system for one fast period using transient simulation. The solution at the end of the period is used to adjust the initial condition by any nonlinear solver, usually the Newton–Raphson method, until periodicity is satisfied. However, shooting is only suitable for one-tone problems. When signals involve widely separated rates, shooting can be very inefficient, since it uses transient simulation which can take a large number of time steps.

A related method for solving steady-state problems is the FDTD method, also known as time-domain collocation [6]. In this method, finite difference approximations are used to discretize differential equations over one period. This results in a system of algebraic equations, which are solved simultaneously to find solutions of all time points in the discretization. This may be considered to be an “unrolled” form of shooting. It was noted [7] that shooting usually converges faster than FDTD. In shooting, only the initial condition is guessed and then the system is integrated accurately by transient simulation. However, in FDTD, all discretization points are guessed simultaneously, some of which may not be close to solutions. In some

cases, this may cause slow convergence of Newton–Raphson solution methods for an FDTD system.

In the frequency domain, HB (e.g., [6], [8]–[11]) is the commonly used method for finding steady states. HB uses Fourier series to approximate the node voltages or branch currents in a circuit. A system of algebraic equations, in terms of Fourier coefficients, is formed and can be solved by the Newton-Raphson method. One significant advantage of HB is that it is capable of solving multitone, or quasiperiodic problems. Moreover, HB can be combined with time-domain methods to solve for slow envelopes, as discussed later.

For shooting, the size of the equation system is the same as that of the circuit DAE. However, the computation time can be considerable for large problems, since the Jacobian matrix is dense. The FDTD system, on the other hand, is sparse, although it is larger than the shooting system by a factor of the number of discretization points. However, it is still difficult to solve numerically, since fill-ins tend to appear in certain block-columns when LU factorization is applied to the Jacobian matrix. For HB, the system is both large (larger by a factor of the number of harmonics used when compared to shooting) in size and dense in structure. This limits traditional HB to relatively small circuits.

The superlinear computation limitation inherent in all the above methods was alleviated significantly with the advent of so-called fast methods [12]–[14], based on iterative techniques for linear matrix solution (e.g., [15], [16]). Unlike traditional linear solvers, which are typically based on LU factorization, iterative linear methods solve linear systems by applying a series of matrix-vector products. In fast variants of HB, the dense HB Jacobian matrix is decomposed into products and sums of sparse matrices or special matrices, such as DFTs, that have regular structure. These matrices can be applied or inverted quickly. Therefore, the multiplication of the Jacobian matrix with a vector can be performed in almost linear time; further, the (dense) Jacobian matrix is never computed or stored explicitly, thereby saving memory. These methods, combined with good preconditioners (e.g., [17]), significantly reduce the cost of solving the HB system, making steady-state computation practical for circuits with up to several thousand nodes. Shortly after, fast methods were proposed for HB [12], [13], they were also applied successfully to shooting and collocation/FDTD to improve speed and memory requirements [18]. Therefore, all steady-state methods, whether time- or frequency-domain have roughly the same computation complexity, when preconditioned iterative linear techniques are applied.

B. Solving for the Envelope

In addition to steady-state analysis, designers are often interested in finding slowly varying transient components (the “envelope”) of the system. The earliest time-domain envelope technique, to our knowledge, was proposed by Petzold [19]. This method exploits the fact that when the solution of highly oscillatory ODEs is sampled at every time-period T (the period of the fast oscillations), the resulting samples can be interpolated as a curve that varies slowly. This curve is defined as the “envelope” (dashed line in Fig. 1). Since the envelope varies orders of magnitude slower than the fast oscillation frequency, it can be followed by large steps. In Petzold’s method, a cycle of the

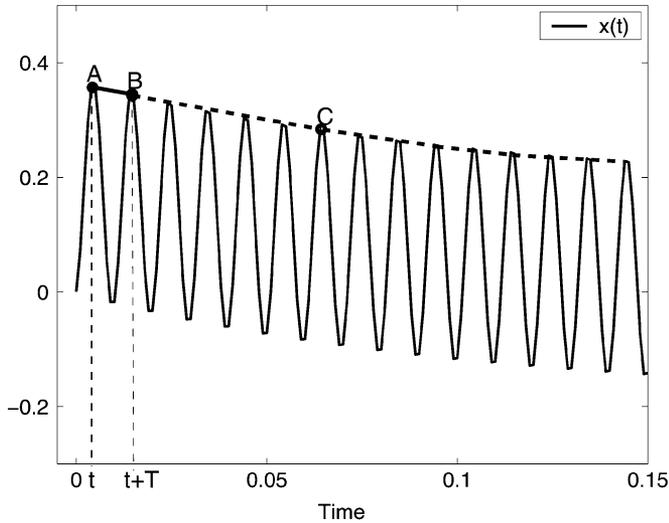


Fig. 1. Illustration of envelope and Petzold's method.

fast-varying oscillation is integrated accurately using a standard time-integration method, as shown in Fig. 1 (from point A to B). A secant line between A and B can then be extrapolated over a large time step, which may be many cycles, to obtain the solution at C. Starting from this new point, the process is repeated until the end of the time interval of interest is reached. Petzold's envelope method, variants of which apply explicit and implicit Adams formulae, was adapted and used for the simulation of switching power and filter circuits [20]. Similar ideas were presented in [6], [21], and [22], where it is assumed that the sequence A, B, C, \dots can be described by Fourier series with a few terms, thus can be computed by HB.

Another class of envelope methods, using a combination of HB and time-domain integration methods, was proposed in [1], [23], and [24]. These kinds of techniques are called Fourier envelope methods. In these methods, HB is used to solve the steady state of the fast component of signals. DAEs in terms of Fourier coefficients are then formed, which capture the slow behavior of circuits. Time-domain integration methods can be used to solve these DAEs to obtain the slow envelopes. However, these frequency-domain methods are limited to weakly nonlinear circuits since. For example, pulses or sharp edges, often generated by strong nonlinearities, cannot be represented well with only a few Fourier coefficients. Moreover, strong nonlinearities destroy the block-diagonal dominance of the HB Jacobian matrix, which makes preconditioned iterative techniques [13], [25] ineffective. This can limit HB to small circuits, since preconditioned iterative linear solution techniques are needed for fast HB techniques to be applied to large circuits.

Recently, a novel class of methods based on MPDEs have shown promise in further reducing computation times for fast/slow systems, while being applicable to a wider range of circuits encompassing both strong and weak nonlinearities, as well as oscillators [26]. MPDEs form a framework in which previous envelope methods can be interpreted, and have also led to new methods in both time and frequency domains [2], [3], [27].

Although these novel methods have a number of advantages, they can sometimes suffer from instabilities when solved using

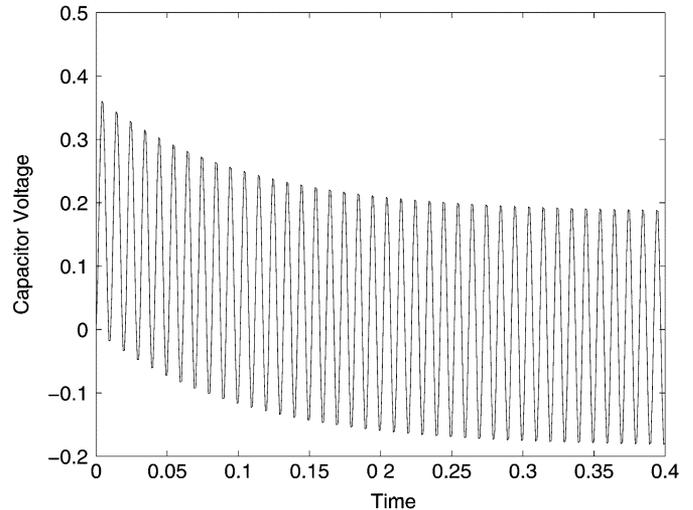


Fig. 2. Transient simulation of the standard test problem.

time-domain techniques. The advantage of these methods can be utilized only when they work stably and robustly for practical circuits and systems. Whereas stability results have long been known for DAEs [28]–[30], and have led to robust numerical integration methods (e.g., Gear methods), the same has not so far been true of MPDE-based methods.

Recently, in [31], the lack of robustness in Fourier envelope methods was noted. Fourier envelopes use HB along the fast time scale; for such techniques, [31] proposed solutions to address robustness issues. For fully time-domain methods, however, the MPDE stability question has so far remained open. It has been observed in practice that, instead of generating a slow-varying envelope, time-domain MPDE methods often show nonphysical oscillatory behaviors, typical of instability.

III. ILLUSTRATION OF STABILITY PROBLEMS IN EXISTING TIME-DOMAIN MPDE METHODS

Consider the standard test problem for multistep integration methods [30]

$$\dot{x} + \lambda x = b(t), \quad (1)$$

Without loss of generality, we use $\lambda = 10$ and $b(t) = \sin(2\pi 100t)$ here. Physically, this corresponds to an RC network, with time constant $\tau = 100$ ms and an sinusoidal voltage source at 100 Hz, where $x(t)$ is the voltage across the capacitor.

Fig. 2 shows the transient simulation result of the system. In addition to the fast-varying component resulting from the 100 Hz input, there is also a slow transient envelope caused by the slow RC time constant.

Following [3], the MPDE corresponding to (1) is

$$\frac{\partial}{\partial t_1} \hat{x}(t_1, t_2) + \frac{\partial}{\partial t_2} \hat{x}(t_1, t_2) + \lambda \hat{x} = \sin(2\pi 100t_2). \quad (2)$$

In this equation, we choose t_1 to be the *slow* time scale and t_2 to be the *fast* time scale. Periodic boundary conditions are assumed on the fast time scale [3]. The time steps along the slow and fast time scales are denoted by h_1 and h_2 , respectively. To solve the MPDE in the time domain, we first discretize the fast

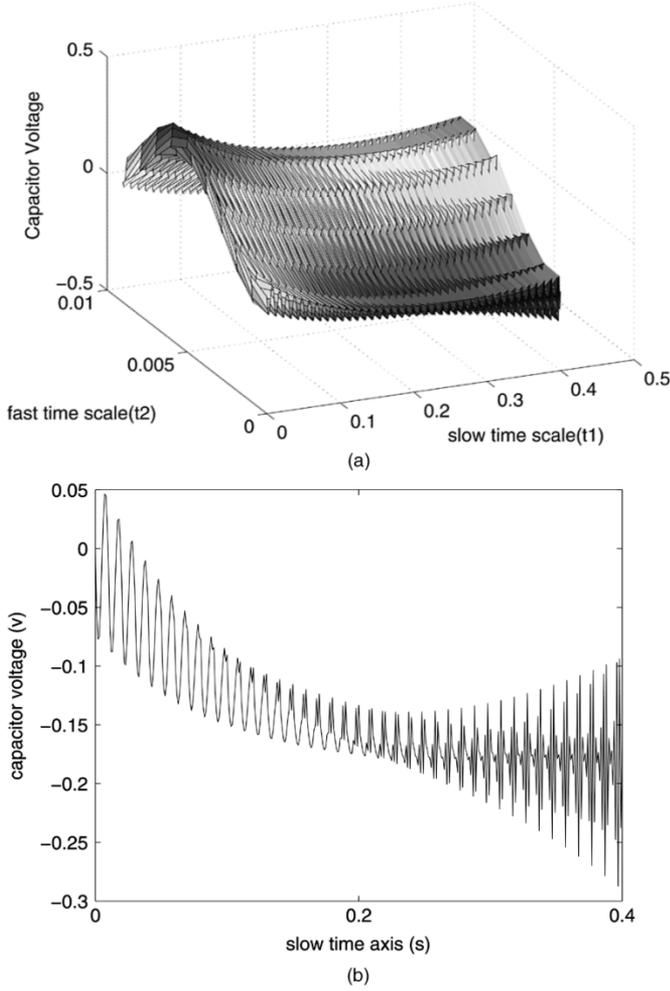


Fig. 3. Test ODE: MPDE solutions. The fast-varying component resulting from 100 Hz input has a period of 10 ms. The fast time scale was discretized with 10 points, i.e., the time step along the fast time scale, h_2 , is 1 ms. The time step along the slow time scale, h_1 , is also 1 ms. (a) Multitime solution, (b) slice at $t_2 = 0$.

time scale with n_2 points, using a finite difference approximation such as forward differences (FDs), backward differences (BDs), or centered differences (CDs). Note that, *due to the periodicity of the fast time scale, there does not appear to be a strong reason to choose BDs over FDs*. This results in the transformation of the MPDE into an ODE (or a DAE, in the general case). Then, the differential equation is integrated along the slow time scale using conventional time-integration methods such as backward Euler (BE), trapezoidal, or Gear's methods. The transient solution can then be recovered from the MPDE solution by interpolating along the characteristics, in this case the diagonal line $t_1 = t_2 = t$.

From Fig. 2, the envelope for this system is a slowly decaying curve. However, depending on the choice of discretization method for the time scales, the MPDE approach can fail to find this slowly-varying envelope. Fig. 3 shows a multitime solution of this system with FD and BE applied in the fast and slow time scales, respectively, with $h_1 = h_2 = 1$ ms. The slow envelope, which is the variation of $\hat{x}(t_1, t_2)$ in t_1 [shown in 3(b)], not only contains large oscillations, but eventually becomes unbounded for this case.

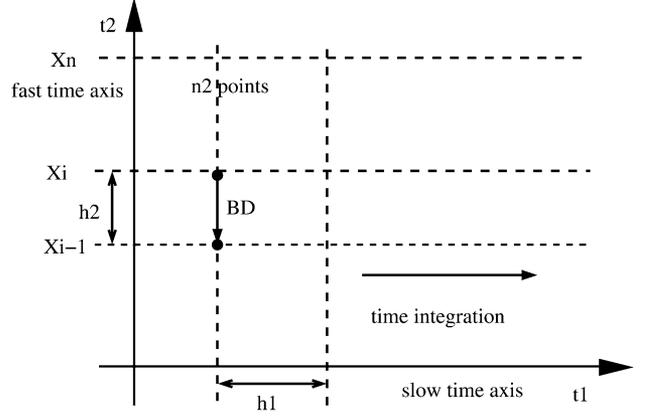


Fig. 4. Discretize the fast time scale by BD.

In the remainder of this paper, we first develop a clear understanding of such undesirable phenomena by investigating the stability properties of the system, and then establishing methods to circumvent instabilities.

IV. STABILITY ANALYSIS OF TIME-DOMAIN MPDE METHODS

In time-domain MPDE methods, different discretization methods can be applied to each time scale, but, as we will show, not all combinations are stable. To find the underlying cause of instability, we investigate the eigenstructure of the system after the fast time scale is discretized. We then examine the effect of discretization on the slow time scale.

A. Effect of Discretization Along the Fast Time Scale on Poles and Eigenvalues

When the BD method is applied along the fast time axis, the ODE corresponding to (2) becomes

$$\frac{d}{dt_1} \hat{x}_i(t_1) = -\frac{\hat{x}_i(t_1) - \hat{x}_{i-1}(t_1)}{h_2} - \lambda \hat{x}_i(t_1) + b(t_2) \quad \forall i \in \{1, \dots, n_2\}. \quad (3)$$

Here, \hat{x}_i is the solution along the line $t_2 = (i - 1)h_2$, as shown in Fig. 4. Recall that for the linear problem $\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{b}$, if $\mathbf{X}(t)$ is the fundamental solution ($\dot{\mathbf{X}} = \mathbf{A}\mathbf{X}$), then

$$\mathbf{x}(t) = \mathbf{X}(t) \left[\mathbf{c} + \int_0^t \mathbf{X}^{-1}(s) \mathbf{b}(s) ds \right] \quad (4)$$

where $\mathbf{c} = \mathbf{x}(0)$. So, the question of the stability of \hat{x}_i relates to the stability of its homogeneous problem (i.e., with $b(t_2) = 0$) [32], [33].

Collecting the (3) along all $t_2 = (i - 1)h_2$ slices leads to n_2 equations and the corresponding homogeneous problem

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x}, \quad \text{where } \mathbf{x} = [\hat{x}_1, \dots, \hat{x}_{n_2}]^T. \quad (5)$$

Here, the matrix $\mathbf{A} \in \mathfrak{R}^{n_2 \times n_2}$ is sparse with the structure

$$\mathbf{A} = \begin{bmatrix} -\frac{1}{h_2} - \lambda & & & & \\ & -\frac{1}{h_2} - \lambda & & & \\ & & \ddots & & \\ & & & \frac{1}{h_2} & \\ & & & & -\frac{1}{h_2} - \lambda \end{bmatrix}$$

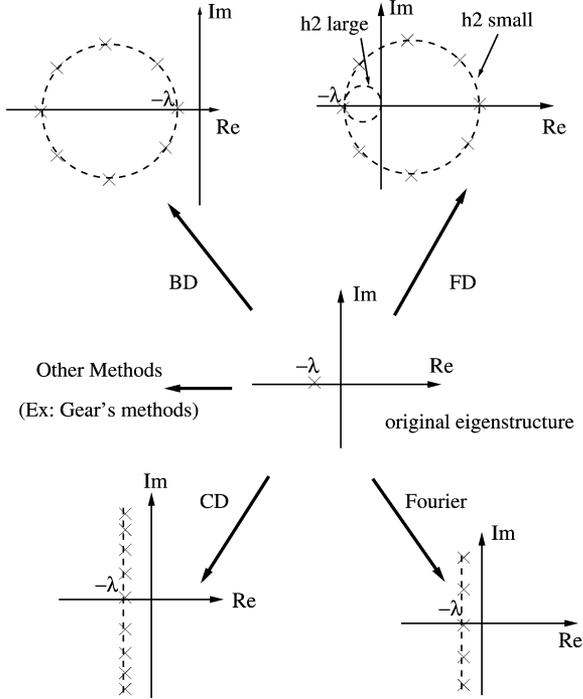


Fig. 5. Effect of discretization along fast time scale on poles/eigenvalues.

$$= \left(-\frac{1}{h_2} - \lambda \right) \mathbf{I} + \frac{1}{h_2} \mathbf{P}. \quad (6)$$

Because of periodicity conditions along the fast time scale, the matrix \mathbf{A} has a circular structure. Here, \mathbf{I} is the identity matrix and \mathbf{P} is a permutation matrix. The identity matrix only shifts the eigenvalues by $-(1)/(h_2) - \lambda$ and the permutation matrix has its eigenvalues around a circle, i.e., $e^{j\theta}$. Therefore, \mathbf{A} has n_2 distinct eigenvalues (as shown in Fig. 5)

$$\hat{\lambda}_i = -\frac{1}{h_2} - \lambda + \frac{1}{h_2} e^{j\theta_i}, \quad \theta_i = \frac{2\pi}{n_2}(i-1) \quad \forall i \in \{1, \dots, n_2\}. \quad (7)$$

\mathbf{A} is diagonalizable and a similarity transformation can transform \mathbf{A} into $\mathbf{\Lambda}$, a diagonal matrix of the eigenvalues of \mathbf{A}

$$\mathbf{M}^{-1} \mathbf{A} \mathbf{M} = \mathbf{\Lambda}. \quad (8)$$

With the change of variables $\mathbf{y} = \mathbf{M}^{-1} \mathbf{x}$, we have a decoupled system of equations

$$\dot{\mathbf{y}} = \mathbf{\Lambda} \mathbf{y}. \quad (9)$$

Here, for each y_i , there is a test equation $\dot{y}_i = \hat{\lambda}_i y_i$. Thus, the stability for \mathbf{y} , and for \mathbf{x} as well, is determined by the eigenvalues; stability is guaranteed if $\text{Re}(\hat{\lambda}_i) \leq 0$, for all $i = 1, \dots, n_2$.

Following the same procedure, we obtain the eigenstructure of the system when discretizing the fast time scale by FD

methods (see the Appendix for further details), as indicated in Fig. 5

$$\hat{\lambda}_i = \frac{1}{h_2} - \lambda - \frac{1}{h_2} e^{j\theta_i}, \quad \theta_i = \frac{2\pi}{n_2}(i-1) \quad \forall i \in \{1, \dots, n_2\}. \quad (10)$$

If CDs are used, the corresponding matrix \mathbf{A} has n_2 eigenvalues (see the Appendix):

$$\hat{\lambda}_i = -\lambda - \frac{1}{h_2} j \sin \theta_i, \quad \theta_i = \frac{2\pi}{n_2}(i-1), \quad \forall i \in \{1, \dots, n_2\}. \quad (11)$$

Fig. 5 summarizes the transformation of the eigenvalues due to discretization (note that the original ODE system has one negative eigenvalue). For comparison, we have also shown the eigenstructure of Fourier envelope methods [31] for this problem. When BD, CD, or Fourier methods are used along the fast time scale, all eigenvalues of the resulting DAE are located on the negative half plane. Therefore, discretization of the fast time scale by these methods results in a stable system. FD, on the other hand, can result in some eigenvalues on the positive half plane when h_2 is small while it keeps all eigenvalues negative only when h_2 is large. In actuality, when the two time scales are widely separated, $h_2 \ll (1)/(\lambda)$, so $(2)/(h_2) \gg \lambda$. In this case, discretization by FD leads to an unstable system.

An interesting result of this analysis is apparent if one considers the original system (1) for which the solution is always stable. When an FD method is applied to the fast time scale, the *intermediate* system may become unstable. The reason for this change in the stability is due to the introduction of the term $1/h_2$, which is usually much bigger than λ , to the eigenvalues ($\hat{\lambda}$) of the new system. Therefore, the stability is now dominated by this term instead of λ as in the original system (1).

Discretization by BD also has a similar effect; however, note that the difference between these two methods is a sign change of $1/h_2$ [compare (7) and (10)]. While all eigenvalues from BD discretization are located on the left of $-\lambda$, the eigenvalue of the original ODE, they locate on the right of $-\lambda$ for FD case.

However, unlike the cases with FD and BD, when CD is applied on the fast time scale, the term $1/h_2$ does not have any effect on stability because it only appears in imaginary parts of the eigenvalues. Stability in this case is solely determined by the original λ . The same is true for Fourier envelope methods. Note that eigenvalues are equally spaced for Fourier methods, while they are sparse at the center but dense at the two ends for the CD case.

We could also consider higher order methods, such as k th order Gear's methods ($k \geq 2$). Here, we derive the eigenstructure for the case when the second-order Gear's method is used on the fast time scale. Higher order Gear's methods have similar eigenstructures. The corresponding DAE is

$$\frac{d}{dt_1} \hat{x}_i(t_1) = -\frac{3\hat{x}_i(t_1) - 4\hat{x}_{i-1}(t_1) + \hat{x}_{i-2}(t_1)}{2h_2} - \lambda \hat{x}_i(t_1) + b(t_2). \quad (12)$$

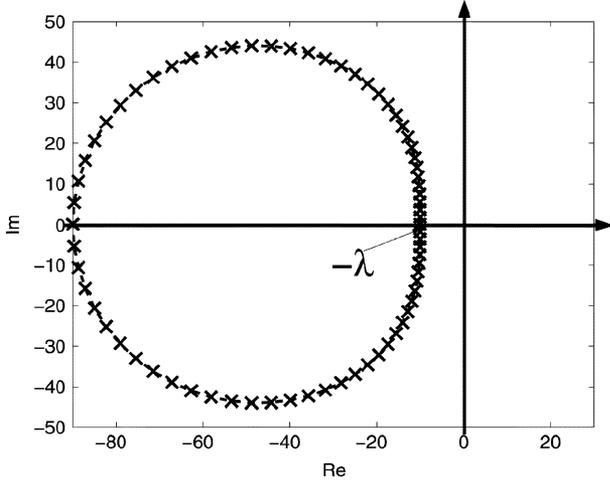


Fig. 6. Locations of eigenvalues after fast time discretization by Gear's second-order method ($(1)/(h_2) = 20, \lambda = 10$).

The matrix \mathbf{A} has the circulant structure

$$\mathbf{A} = \begin{bmatrix} a & & & c & b \\ b & a & & & c \\ c & b & a & & \\ & \ddots & \ddots & \ddots & \\ & & & c & b & a \end{bmatrix} = a\mathbf{I} + b\mathbf{P} + c\mathbf{P}^2 \quad (13)$$

where

$$a = -\frac{3}{2h_2} - \lambda, \quad b = \frac{2}{h_2}, \quad c = -\frac{1}{2h_2}.$$

It has n_2 distinct eigenvalues (as shown in Fig. 6)

$$\hat{\lambda}_i = a + be^{j\theta_i} + ce^{j2\theta_i}, \quad \theta_i = \frac{2\pi}{n_2}(i-1) \quad \forall i \in \{1, \dots, n_2\}. \quad (14)$$

All eigenvalues are located on the negative half plane. Thus, the resulting system, after discretizing the fast time scale by the second-order Gear's method, is a stable one. For the third-order method, there may appear positive eigenvalues, depending on the values of both h_2 and λ , as shown in Fig. 7 (see the Appendix for the eigenstructure of this case). In fact, when the two time scales are widely separated, $(1)/(h_2) \gg \lambda$, and there can be positive eigenvalues when third or higher order methods are used. The trend is that the higher the order of the method, the less stable the resulting system becomes, with positive eigenvalues becoming more likely. In other words, there is a tradeoff between accuracy and stability.

B. Effect of Discretization on the Slow Time Scale

Continuing this analysis, the slow time scale is discretized next by a conventional DAE integration method (e.g., the BE method or the trapezoidal rule). When the fast time scale is discretized by methods which only generate eigenvalues/poles on the negative half plane (such as BD, CD, Fourier, or second-order Gear), using any A-stable method along the slow time scale results in a stable solution. (Recall that a method is said to be A-stable if its stability region includes the entire left half

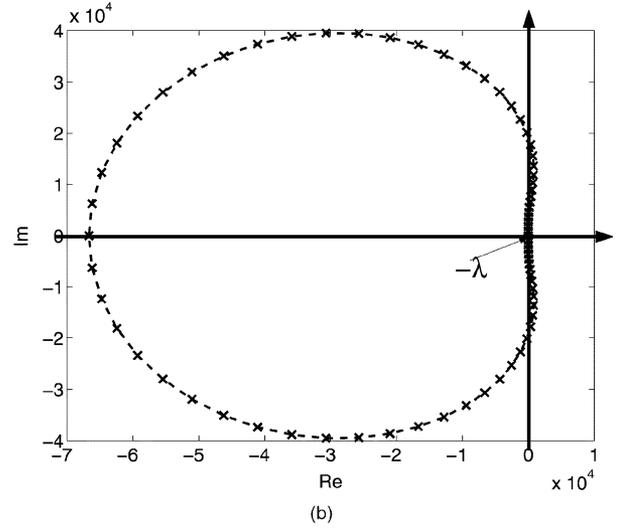
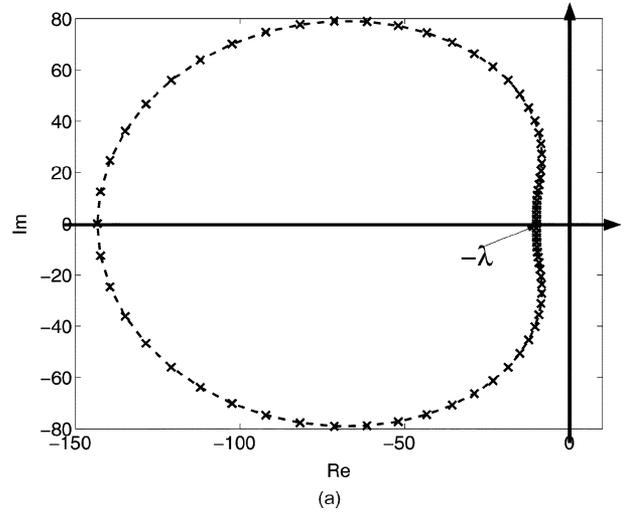


Fig. 7. Locations of eigenvalues after fast time discretization by 3rd Gear's method. (a) case 1: $(1)/(h_2), \lambda$ are not widely separated ($(1)/(h_2) = 20, \lambda = 10$), (b) case 2: $(1)/(h_2) \gg \lambda$ ($(1)/(h_2) = 10^4, \lambda = 10$).

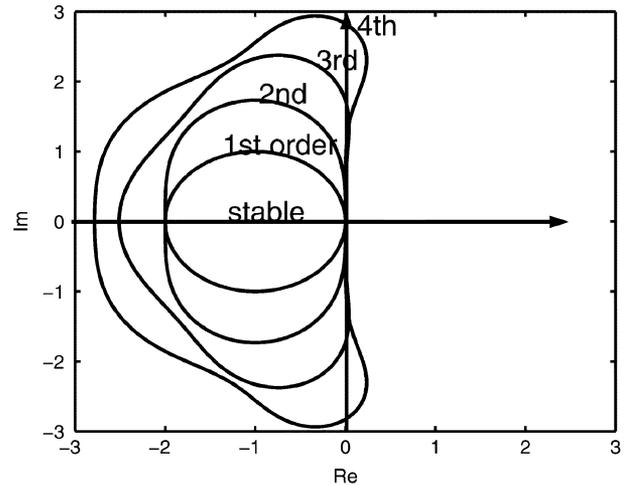


Fig. 8. Stability regions of RK methods.

plane). Several popular implicit methods, such as BE, the trapezoidal rule, second-order Gear and implicit Runge–Kutta (RK) methods (e.g., [34], [35]), are A-stable. The third-order Gear's

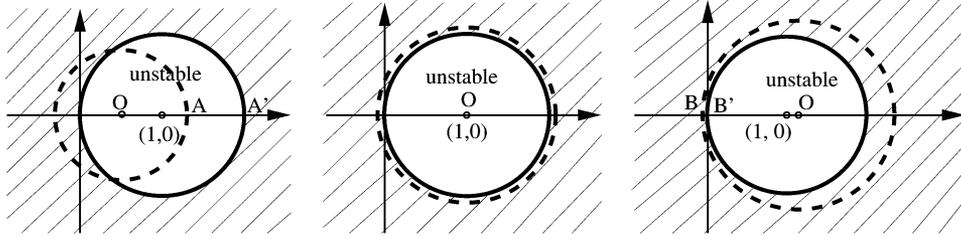


Fig. 9. Relative locations of $\sigma = h_1 \hat{\lambda}$, and the unstable circle of the BE method. The stability region of BE is the shaded area [32], [33].

algorithm, although not A-stable, also could be an option since its stability region almost covers the entire left-half plane.

Explicit methods, like the forward Euler method, are usually not preferred, since their stability regions only include small portions of the left half plane. Explicit RK methods, which have both higher accuracy and better stability regions than many other explicit methods, are also not good choices for the slow time discretization. The stability regions for i th order RK methods ($i = 1, 2, 3, 4$) are shown in Fig. 8. From the previous section, note that the imaginary part of $\sigma = h_1 \hat{\lambda}$ could be as large as $\pm(h_1/h_2)$. When the two time scales are widely separated, typically $h_1 \gg h_2$. However, from Fig. 8, even for the fourth-order RK method, the stability region cannot be beyond ± 3 along the imaginary coordinate; therefore, it results in unstable solutions.

Some implicit methods, such as BE and the second-order Gear's method, are overly stable; they are stable even for ODEs with positive eigenvalues. This suggests a way to obtain a stable solution, even when the system has positive eigenvalues. For instance, with suitable time step sizes along the slow time scale (h_1), BE can damp out artificial instabilities introduced by FD discretization along the fast time scale. As a result, nonexhaustive empirical experiments with such combinations of methods can easily reach the erroneous conclusion that FDs are not different from BDs for MPDE discretization.

We next look into how to choose h_1 in order to obtain a stable solution in this case. From (10), we know that $\sigma = h_1 \hat{\lambda}$ is located on a circle with center $(h_1((1)/(h_2) - \lambda), 0)$ and radius h_1/h_2 . With fixed h_2 , a change in h_1 not only moves the center but also changes the radius. Fig. 9 illustrates the relative locations of $\sigma = h_1 \hat{\lambda}$ (dashed line) and the unstable circle of the BE method (solid line). The stability region of BE is the area outside this solid line circle (shaded area). In order to obtain a stable solution, we only need to choose the step size h_1 to ensure that $\sigma = h_1 \hat{\lambda}$ (dashed circle) lies outside the solid circle.

When h_1 is small, the center of $\sigma = h_1 \hat{\lambda}$ is located to the left of the instability circle of the BE method, i.e., $h_1((1)/(h_2) - \lambda) < 1$. To ensure that all $\sigma = h_1 \hat{\lambda}$ are located outside the instability circle, the radius of the dashed circle $|OA|$ must be $\geq |OA'|$, i.e.,

$$\frac{h_1}{h_2} \geq 2 - h_1 \left(\frac{1}{h_2} - \lambda \right). \quad (15)$$

Hence,

$$h_1 \geq \frac{2}{\frac{2}{h_2} - \lambda}. \quad (16)$$

TABLE I
TEST-PROBLEM: BEHAVIOUR FOR DIFFERENT h_1 VALUES

h_2	h_1	$h_1 \geq \frac{2}{\frac{2}{h_2} - \lambda}$	stability
1 ms	0.9 ms	No	unstable, blows up quickly
1 ms	1 ms	No	unstable, blows up slowly
1 ms	1.1 ms	Yes	damps slowly
1 ms	10 ms	Yes	damps quickly

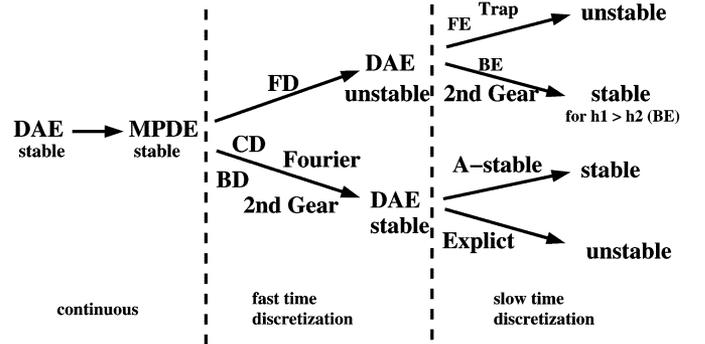


Fig. 10. Discretization flow.

TABLE II
STABILITY SUMMARY

	slow	FE	BE	Trap	2 nd -order Gear	explicit RK
fast	FD	BAD	ok	BAD	ok	BAD
BD	BAD	Good	Good	Good	Good	BAD
CD	BAD	Good	Good	Good	Good	BAD
2 nd -order Gear	2 nd -order Gear	BAD	Good	Good	Good	BAD
Fourier	BAD	Good	Good	Good	Good	BAD

As h_1 increases, the center of the dashed circle moves to the right. At some point, $h_1((1)/(h_2) - \lambda) = 1$ and the two centers overlap. In this situation, the radius of dashed circle $(h_1)/h_2 = 1 + h_1 \lambda > 1$ and the solution is always stable.

As h_1 grows, $h_1((1)/(h_2) - \lambda) > 1$ and the center of the dashed circle moves to the right of $(1, 0)$. To ensure that all $\sigma = h_1 \hat{\lambda}$ are outside the instability circle, the radius of the dashed line $|OB|$ must be $\geq |OB'|$, i.e.,

$$\frac{h_1}{h_2} \geq h_1 \left(\frac{1}{h_2} - \lambda \right). \quad (17)$$

It is obvious that this equation always holds. Considering these three situations, we conclude that h_1 must satisfy (16) to ensure stability of the solution.

To illustrate this point, Table I depicts the above analysis for different h_1 values. (These results are backed by simulations in

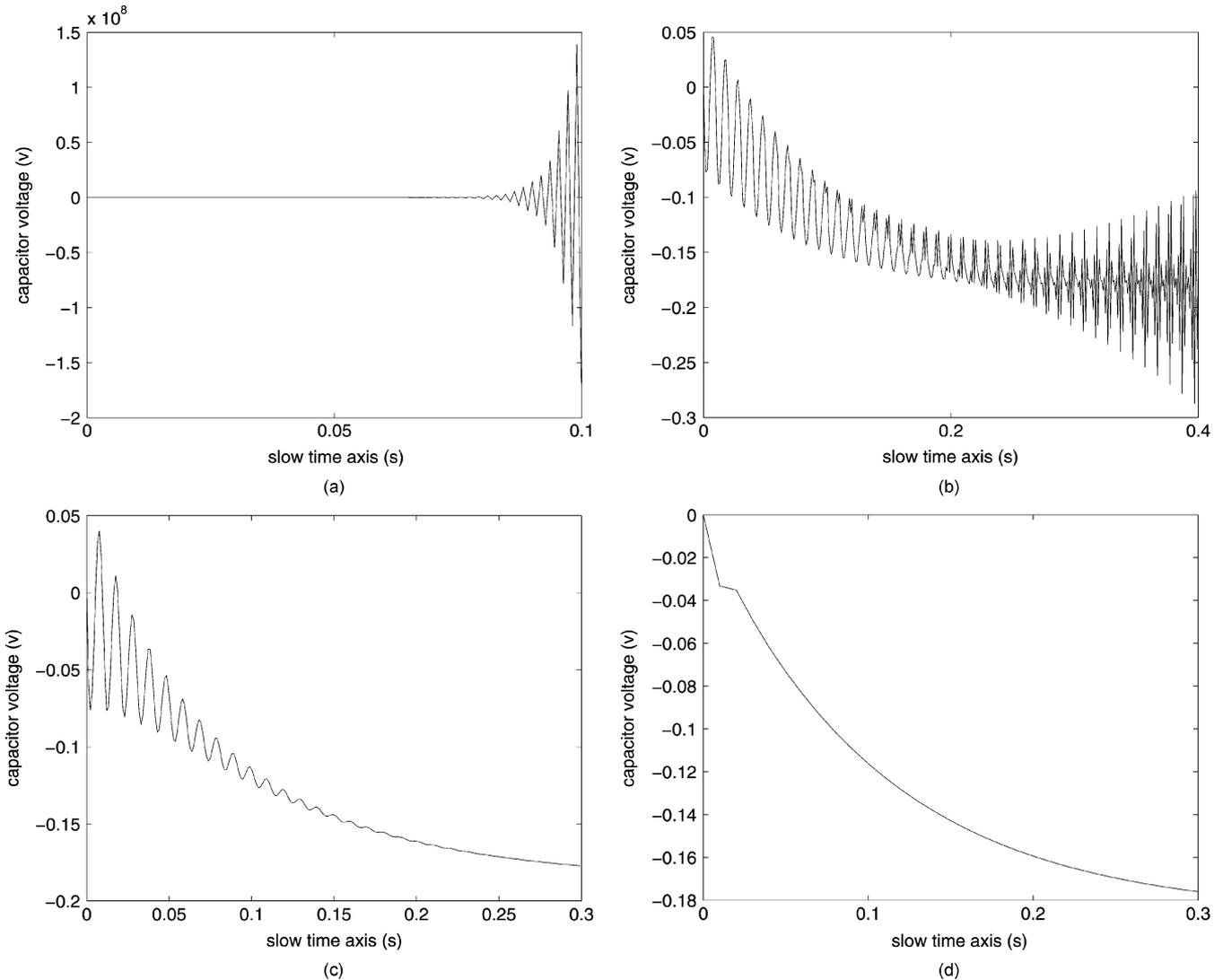


Fig. 11. Test problem: FD and BE along fast and slow time scale, respectively. (a) $h_1 = 0.9$ ms, (b) $h_1 = 1$ ms, (c) $h_1 = 1.1$ ms, (d) $h_1 = 10$ ms.

Section V-A, which match perfectly). For the same h_2 , even a small change in h_1 can totally change the stability of the solution. For example, if $h_2 = 1$ ms, with $h_1 = 1$ ms, the solution is unstable, while with $h_1 = 1.1$ ms, the solution will be stable.

Notice here that h_1 only needs to be approximately larger than h_2 to ensure a stable solution with stability being proportional to the size of h_1 . In reality, when the two time scales are widely separated, we usually choose $h_1 \gg h_2$. Therefore, the stable solution is usually obtained, without uncovering the potential instability inherent in this combination.

C. Stable Time-Domain MPDE Methods

To summarize the above analysis, we consider the case of interest ($(2)/(h_2) - \lambda > 0$). We begin with a stable DAE and transform it to a MPDE form. At this point, prior to discretization, the equation is still stable. However, when this MPDE form is discretized along the fast time scale, the stability condition of the resulting slow-time DAE changes. With the FD method, the result is an unstable system while with BD, CD, or the second-order Gear's method, the result is a stable system. Then, when numerically integrating the slow time scale, the stability properties change once more. For the former case, only using the

“overly” stable methods will result in a stable solution provided that $h_1 > h_2$ (for BE). For the latter case, any A-stable method will result in a stable solution, but explicit methods should be avoided. The results are summarized in Table II; the discretization flow is shown in Fig. 10.

To ensure the stability of the time-domain MPDE solution, one should always use the “Good” combinations in Table II. Note that if one chooses to use FD and BE (or second-order Gear) along the fast and slow time scales, respectively, the resulting combination will yield a stable overall solution but an unstable form was created after the discretization of the fast time scale. In other words, *applying overly stable methods along the slow time scale can compensate for the unstable discretization along the fast time scale*. The results obtained by using these “ok” combinations are correct since they match those obtained by applying “Good” combinations, as shown in Figs. 11(d) and 12 in Section V-A.

D. Projection-Based Perspective

Broadly speaking, the instability of time-domain MPDE methods can be thought to stem from the difficulty of enforcing the slowly-varying constraint along the slow time scale. By

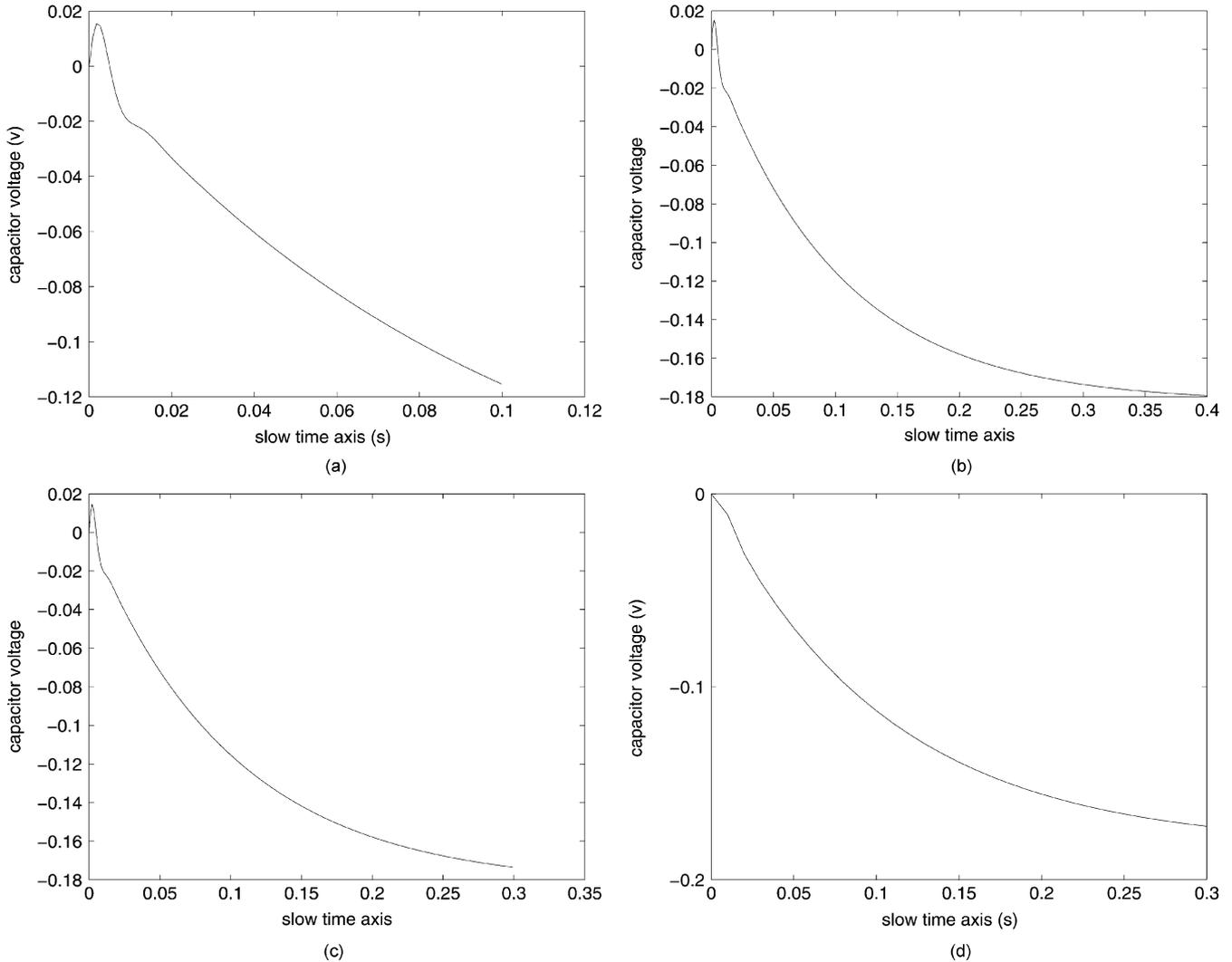


Fig. 12. Test problem: BD and BE along fast and slow time scale, respectively. (a) $h_1 = 0.9$ ms. (b) $h_1 = 1$ ms. (c) $h_1 = 1.1$ ms. (d) $h_1 = 10$ ms.

transforming circuit equations (usually ODEs/DAEs) into MPDEs, one introduces extra degrees of freedom, to balance which extra constraints must be imposed that require that the solution along the slow time scale vary slowly.

One way of discarding unwanted degrees of freedom might be to apply projection-based model reduction methods [36], [37] to reduce the fast time discretized MPDE systems to smaller ones that abstract out the slow envelope components. This concept can be concretized for linear time invaring (LTI) systems using standard projection methodologies to abstract only the nonshifted eigenvalues. However, the MPDE is most useful only for non-LTI systems. For linear time varying (LTV) systems, projecting the system to a smaller one that abstracts only the envelope components is a natural and useful concept. This idea is, in fact, implicit in [38], where one can select one or a few harmonics of Fourier envelopes of interest as outputs. Unfortunately, for nonlinear systems in general, the projection idea appears more difficult to apply, though it remains an interesting subject for further study. The results of this paper may be interpreted as a practical way to achieve roughly the same goal (i.e., abstracting the envelope) implicitly, via numerical methods that, by nature, reject fast-varying components.

V. RESULTS

We have implemented time-domain MPDE methods, using the discretization schemes described above, in quick analog prototyping platform (QAPP) and Myce, which are MATLAB-based packages for prototyping and testing analog-simulation algorithms. All simulations were performed using MATLAB on a 2.4 GHz, Athlon XP-based PC running Linux (kernel 2.4 series).

A. Simple Linear Test Problem

Simulation results for the simple linear test problem of (1) are shown in Figs. 11 and 12. Here, we plot the envelope solution along the $t_2 = 0$ slice. Note that the scales are different. Fig. 11 demonstrates how BE along the slow time scale damps out instabilities introduced by the FD discretization of the fast time scale. Fig. 11(b) is the unstable example of Section III [also shown previously in Fig. 3(b)].

For comparison, we use the same values in Table I for the “Good” combinations. In the interest of brevity, we only present results for the BD + BE combination. Other “Good” combinations show the same stability properties. Fig. 12 indicate that

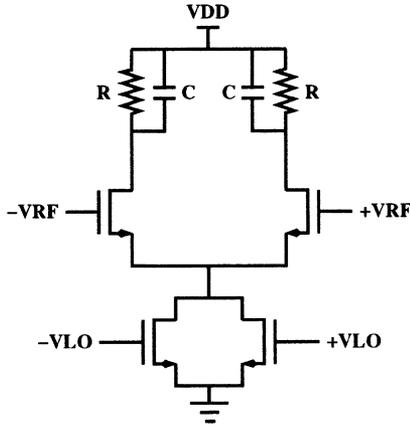
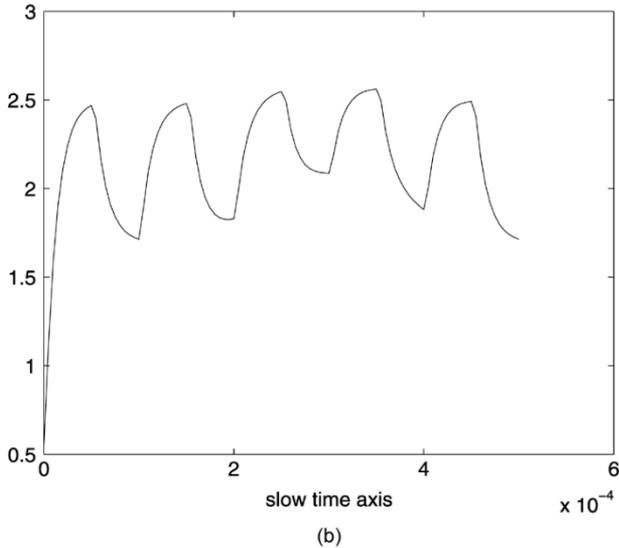
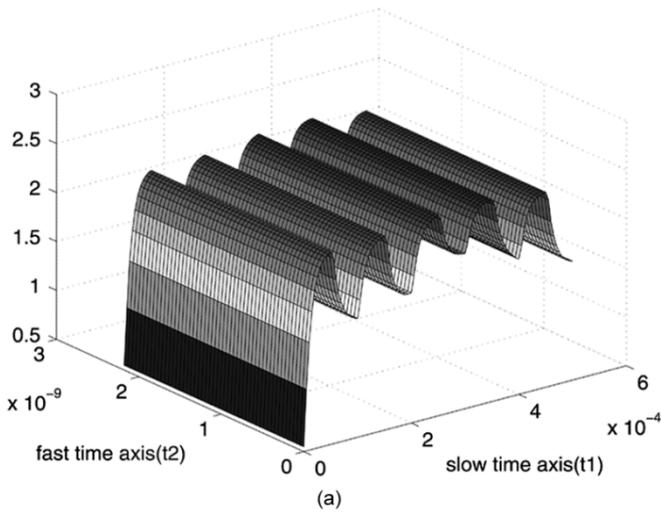
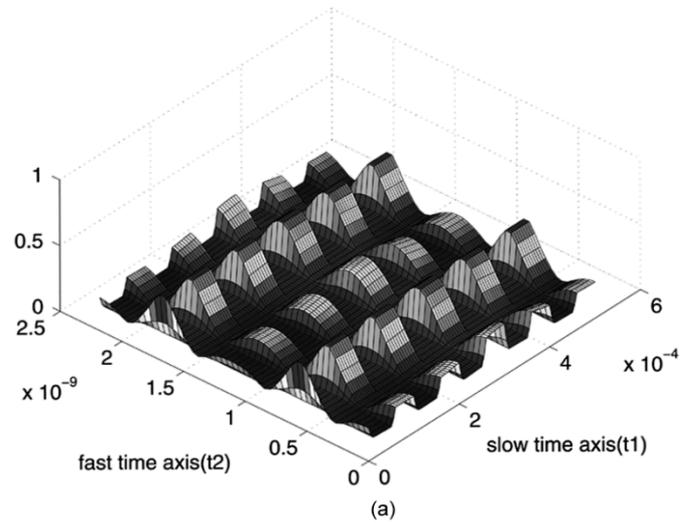
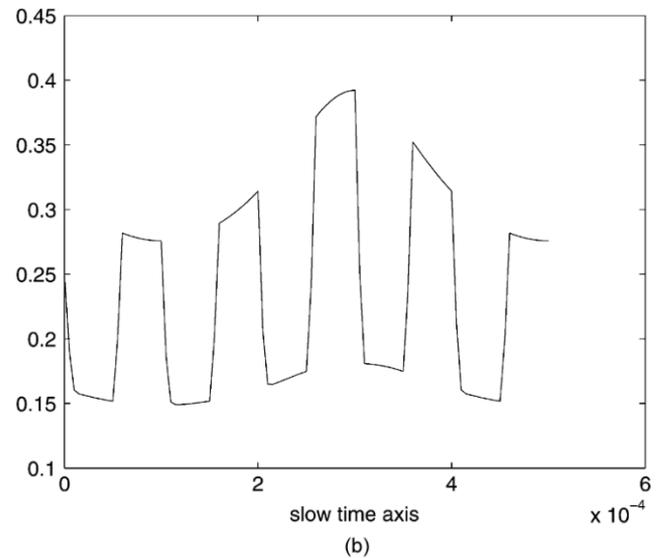


Fig. 13. Balanced CMOS direct-downconversion mixer.


 Fig. 14. Simulation results: at output. (a) Multitime solution. (b) Slice at $t_2 = 0$.

the envelope solution is always stable for any step size along the slow time scale (h_1).


 Fig. 15. Simulation results: at drains of lower MOSFETs. (a) Multitime solution. (b) Slice at $t_2 = 0$.

B. CMOS Direct-Downconversion Mixer (Four Active Transistors)

A balanced CMOS direct-downconversion mixer (based on [39]) is shown in Fig. 13. The lower pair of MOSFETs generates a current that doubles the local oscillator (LO) frequency while the upper pair form a differential pair. This circuit implements a multiplication of the RF and LO signals, with the high-frequency component of the product filtered out by the RC network. In our example, the LO signal is a 450-MHz sinusoid modulated by a 2.5-kHz sinusoid. The RF signal is a 900 MHz carrier modulated by a bit-stream at 10 Kb/s.

The simulation results are illustrated in Figs. 14 and 15, using BD and BE in the fast and slow time scales, respectively. As pointed out in Table II, this is always a stable strategy. The figures show both the multitime solution and a slice through the solution at $t_2 = 0$. Fig. 14 shows the voltage at one output node. The voltage at the drains of the lower MOSFETs is shown in Fig. 15. As can be seen, they double the frequency of the LO signal. The envelope, as expected, is a slowly-varying curve. In

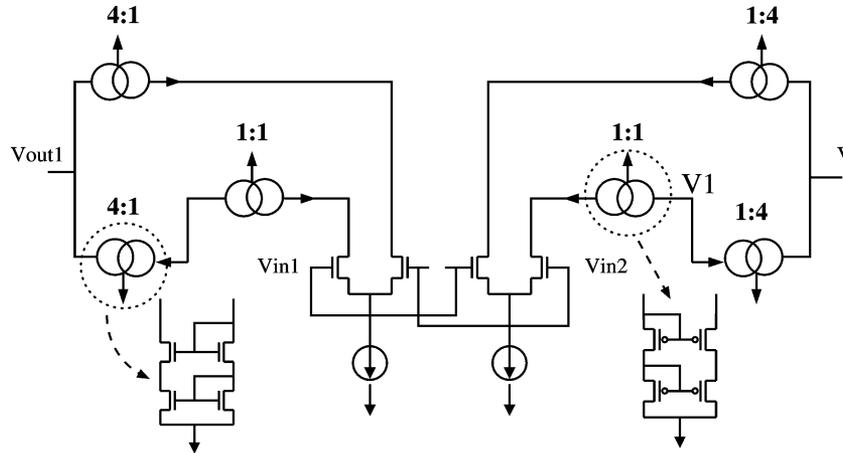


Fig. 16. Fully differential OPAMP.

addition, the slowly changing amplitude of bits illustrates the change of the downconversion gain, which is caused by amplitude changes in the 450-MHz LO signal.

As this example demonstrates, more than two orders of magnitude of speedup can be obtained from robust and stable time-domain MPDE methods compared to traditional DAE integration (such as transient analysis in SPICE). Furthermore, the wider the separation between fast and slow time scales, the greater the speedup.

C. Fully Differential OPAMP (28 Transistors)

A fully differential opamp is shown in Fig. 16, [40]. It is composed of two single-ended output current-mirror opamps with their inputs connected in parallel and each of their outputs being one of the fully differential circuit outputs. Here we use the sum of a fast sinusoid and a slow bit stream as the input. The intent of this simulation is to investigate nonlinearities in the amplifier that can cause intermodulations. The simulation results of the differential output and one internal node (V1) are shown in Figs. 17 and 18. Time-domain MPDE envelope integration on this example runs about 100 times faster than ordinary DAE integration.

VI. CONCLUSION

In this paper, we have used eigenstructure analysis to investigate the stability properties of a variety of discretization methods for time-domain MPDE solution. We have proposed robust and stable methods to circumvent instabilities based on the insight provided by this analysis, which has been confirmed by numerical simulations. We have applied our new robust time-domain MPDE techniques to mixed-signal circuits with strong nonlinearities and demonstrated speedups of two–three orders of magnitude over transient simulations. We expect the adoption of our methods to lead to significant improvements in simulation speed for practical applications with fast/slow signal characteristics.

Although our stability analysis has been for MPDEs with two time scales, similar analysis is possible for more than two time scales, provided that only the slowest time scale is potentially nonperiodic (all other time scales are periodic). Block-circular

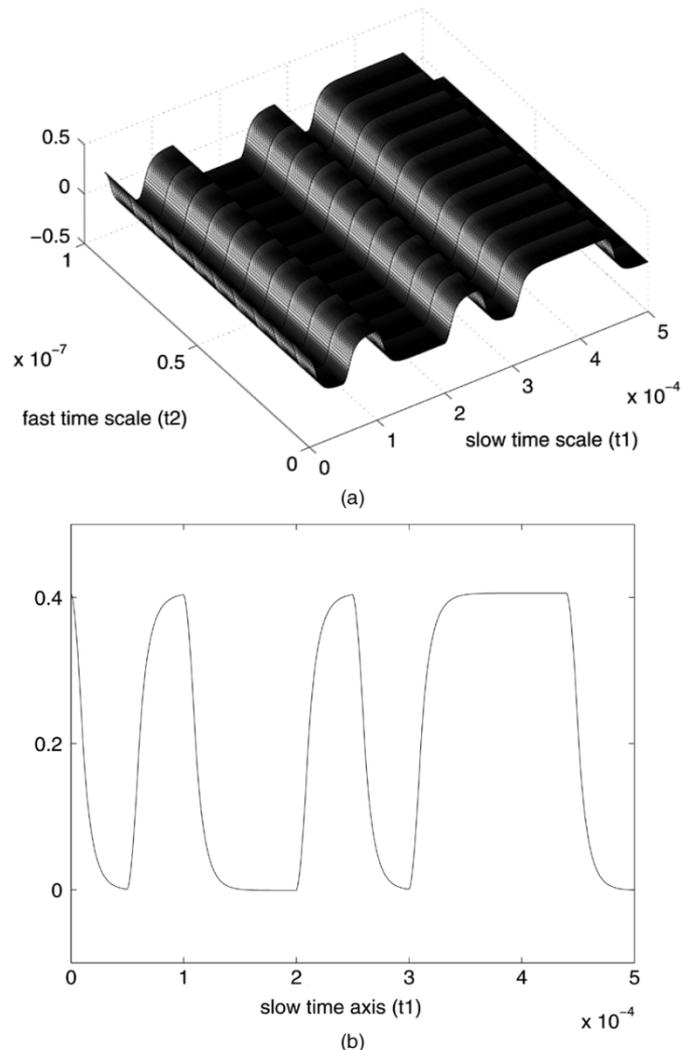


Fig. 17. Simulation results: at differential output ($V_{out1} - V_{out2}$). (a) Multitime solution. (b) Slice at $t_2 = 0$.

matrix structures arise for three or more time scales; preliminary work indicates that the broad conclusions obtained here will hold true for more time scales as well. Further, these results for ODE-based MPDEs, analogous to classical results on stability analysis of ODEs, can also be extended to DAE-based MPDEs, though the DAE case has not been considered here.

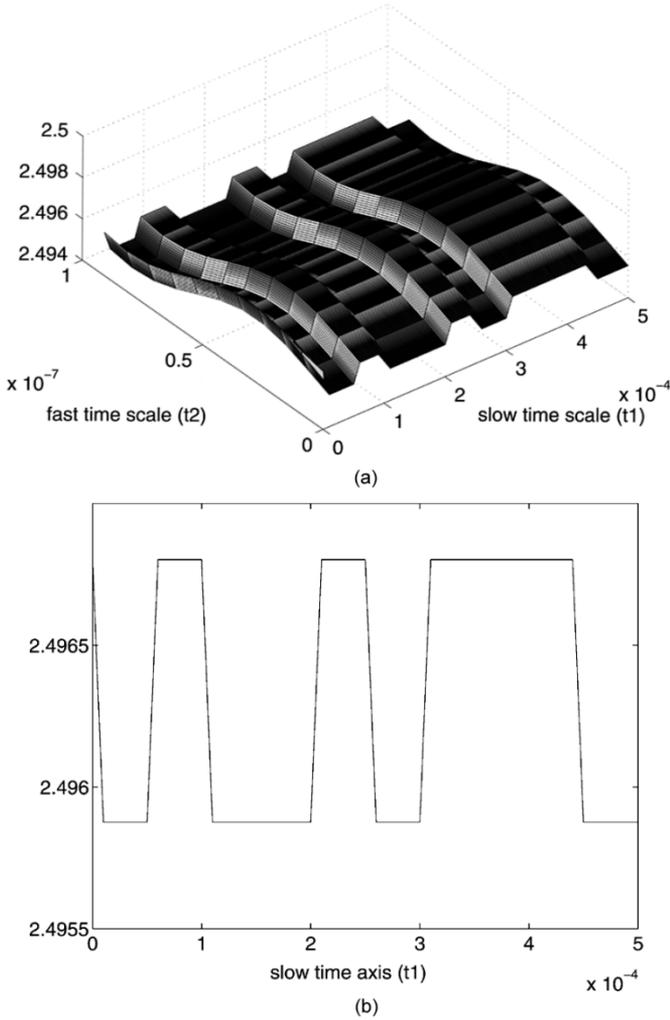


Fig. 18. Simulation results: at one internal node. (a) Multitime solution. (b) Slice at $t_2 = 0$.

APPENDIX

A. Eigenstructure After Fast Time Scale Discretization by FD

When the fast time scale is discretized by the FD method, the DAE corresponding to (2) becomes

$$\frac{d}{dt_1} \hat{x}_i(t_1) = -\frac{\hat{x}_{i+1}(t_1) - \hat{x}_i(t_1)}{h_2} - \lambda \hat{x}_i(t_1) + b(t_2) \quad \forall i \in \{1, \dots, n_2\}. \quad (18)$$

The matrix \mathbf{A} has the structure

$$\mathbf{A} = \begin{bmatrix} \frac{1}{h_2} - \lambda & -\frac{1}{h_2} & & & \\ & \frac{1}{h_2} - \lambda & -\frac{1}{h_2} & & \\ & & \ddots & \ddots & \\ -\frac{1}{h_2} & & & \frac{1}{h_2} - \lambda & \\ \left(\frac{1}{h_2} - \lambda\right) \mathbf{I} - \frac{1}{h_2} \mathbf{P}, & & & & \end{bmatrix} \quad (19)$$

with n_2 distinct eigenvalues

$$\hat{\lambda}_i = \frac{1}{h_2} - \lambda - \frac{1}{h_2} e^{j\theta_i}, \quad \theta_i = \frac{2\pi}{n_2}(i-1) \quad \forall i \in \{1, \dots, n_2\}. \quad (20)$$

B. Eigenstructure After Fast Time Scale Discretization by CD

When CDs are used to discretize the fast time scale, the DAE corresponding to (2) becomes

$$\frac{d}{dt_1} \hat{x}_i(t_1) = -\frac{\hat{x}_{i+1}(t_1) - \hat{x}_{i-1}(t_1)}{2h_2} - \lambda \hat{x}_i(t_1) + b(t_2) \quad \forall i \in \{1, \dots, n_2\}. \quad (21)$$

The matrix \mathbf{A} has the structure

$$\mathbf{A} = \begin{bmatrix} -\lambda & -\frac{1}{2h_2} & & \frac{1}{2h_2} \\ \frac{1}{2h_2} & -\lambda & -\frac{1}{2h_2} & \\ & & \ddots & \\ -\frac{1}{2h_2} & & \frac{1}{2h_2} & -\lambda \end{bmatrix} \quad (22)$$

with n_2 eigenvalues

$$\hat{\lambda}_i = -\lambda - \frac{1}{h_2} j \sin \theta_i, \quad \theta_i = \frac{2\pi}{n_2}(i-1) \quad \forall i \in \{1, \dots, n_2\}. \quad (23)$$

C. Eigenstructure After Fast Time-Scale Discretization by the Third-Order Gear Method

For Gear's third-order method, the corresponding DAE after fast time scale discretization is

$$\frac{d}{dt_1} \hat{x}_i(t_1) = -\frac{11\hat{x}_i(t_1) - 18\hat{x}_{i-1}(t_1) + 9\hat{x}_{i-2}(t_1)}{6h_2} - \frac{-2\hat{x}_{i-3}(t_1)}{6h_2} - \lambda \hat{x}_i(t_1) + b(t_2). \quad (24)$$

Similar to second-order Gear's method, the matrix \mathbf{A} has the structure

$$\mathbf{A} = a\mathbf{I} + b\mathbf{P} + c\mathbf{P}^2 + d\mathbf{P}^3 \quad (25)$$

where

$$a = -\frac{11}{6h_2} - \lambda, \quad b = \frac{3}{h_2}, \quad c = -\frac{3}{2h_2}, \quad d = \frac{1}{3h_2}.$$

It has n_2 distinct eigenvalues (as shown in Fig. 7)

$$\hat{\lambda}_i = a + be^{j\theta_i} + ce^{j2\theta_i} + de^{j3\theta_i}, \quad \theta_i = \frac{2\pi}{n_2}(i-1) \quad \forall i \in \{1, \dots, n_2\}. \quad (26)$$

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers for their helpful suggestions, particularly Reviewer 4, whose comments formed the basis of the discussion in Section IV-D. Computational resources used include those from the Digital Technology Center and the Supercomputing Institute, University of Minnesota.

REFERENCES

- [1] E. Ngoya and R. Larchevéque, "Envelop transient analysis: A new method for the transient and steady-state analysis of microwave communication circuits and systems," in *Proc. IEEE MTT Symp.*, vol. 3, Jun., 17–21 1996, pp. 1365–1368.
- [2] H. G. Brachtendorf, G. Welsch, R. Laur, and A. Bunse-Gerstner, "Numerical steady state analysis of electronic circuits driven by multi-tone signals," *Elect. Eng.*, vol. 79, pp. 103–112, 1996.
- [3] J. Roychowdhury, "Analyzing circuits with widely separated time scales using numerical PDE methods," *IEEE Trans. Circuits Syst. I, Fundam. Theory Appl.*, vol. 48, no. 5, pp. 578–594, May 2001.
- [4] T. J. Aprille and T. N. Trick, "Steady-state analysis of nonlinear circuits with period inputs," *Proc. IEEE*, vol. 60, no. 1, pp. 108–114, Jan. 1972.

- [5] S. Skelboe, "Computation of the periodic steady-state response of nonlinear networks by extrapolation methods," *IEEE Trans. Circuits Syst.*, vol. CAS-27, pp. 161–175, Mar. 1980.
- [6] K. S. Kundert, J. K. White, and A. Sangiovanni-Vincentelli, *Steady-State Methods for Simulating Analog and Microwave Circuits*. Norwell, MA: Kluwer, 1990.
- [7] K. S. Kundert, private communication, Apr. 1992.
- [8] M. S. Nakhla and J. Vlach, "A piecewise harmonic balance technique for determination of periodic responses of nonlinear systems," *IEEE Trans. Circuits Syst.*, vol. CAS-23, no. 2, pp. 85–91, Feb. 1976.
- [9] S. A. Maas, *Nonlinear Microwave Circuits*. Norwood, MA: Artech House, 1988.
- [10] V. Rizzoli and A. Neri, "State of the art and present trends in nonlinear microwave CAD techniques," *IEEE Trans. Microwave Theory Tech.*, vol. 36, no. 2, pp. 343–365, Feb. 1988.
- [11] R. J. Gilmore and M. B. Steer, "Nonlinear circuit analysis using the method of harmonic balance—A review of the art. Part I. Introductory concepts," *Int. J. Microwave Millimeter Wave CAE*, vol. 1, no. 1, 1991.
- [12] M. Rösch, "Schnell Simulation des stationären Verhaltens nichtlinearer Schaltungen," Ph.D. dissertation, Tech. Univ. Munich, Munich, Germany, 1992.
- [13] R. C. Melville, P. Feldmann, and J. Roychowdhury, "Efficient multitone distortion analysis of analog integrated circuits," in *Proc. IEEE CICC*, May 1995, pp. 241–244.
- [14] D. Long *et al.*, "Full chip harmonic balance," in *Proc. IEEE CICC*, May 1997, pp. 379–382.
- [15] R. W. Freund, G. H. Golub, and N. M. Nachtigal, "Iterative solution of linear systems," *Acta Numerica*, pp. 57–100, 1991.
- [16] Y. Saad, *Iterative Methods for Sparse Linear Systems*. Boston, MA: PWS, 1996.
- [17] 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.
- [18] R. Telichevesky, K. Kundert, and J. White, "Efficient steady-state analysis based on matrix-free krylov subspace methods," in *Proc. IEEE Design Automation Conf.*, 1995, pp. 480–484.
- [19] L. Petzold, "An efficient numerical method for highly oscillatory ordinary differential equations," *SIAM J. Numer. Anal.*, vol. 18, pp. 455–479, 1981.
- [20] K. Kundert, J. White, and A. Sangiovanni, "An envelope-following method for the efficient transient simulation of switching power and filter circuits," in *Proc. Int. Conf. Computer-Aided Design*, 1988, pp. 446–449.
- [21] K. Kundert, J. White, and A. Sangiovanni-Vincentelli, "A mixed frequency-time approach for distortion analysis of switching filter circuits," *IEEE J. Solid-State Circuits*, vol. 24, no. 2, pp. 443–451, Apr. 1989.
- [22] L. O. Chua and A. Ushida, "A switching-parameter algorithm for finding multiple solutions of nonlinear resistive circuits," *Int. J. Circuits. Theory Appl.*, vol. 4, pp. 215–239, 1976.
- [23] D. Sharrit, "New method of analysis of communication systems," presented at the MTTT WMFA: Nonlinear CAD Workshop, Jun. 1996.
- [24] P. Feldmann and J. Roychowdhury, "Computation of circuit waveform envelopes using an efficient, matrix-decomposed harmonic balance algorithm," in *Proc. Int. Conf. Computer-Aided Design*, 1996, pp. 295–300.
- [25] M. Rösch and K. J. Antreich, "Schell stationäre simulation nicht-linearer schaltungen im frequenzbereich," *AEÜ*, vol. 46, no. 3, pp. 168–176, 1992.
- [26] O. Narayan and J. Roychowdhury, "Multitime simulation of voltage-controlled oscillators," in *Proc. IEEE Design Automation Conf.*, 1999, pp. 629–634.
- [27] J. Roychowdhury, "Efficient methods for simulating highly nonlinear multirate circuits," in *Proc. IEEE Design Automation Conf.*, 1997, pp. 269–274.
- [28] C. William Gear, *Numerical Initial Value Problem in Ordinary Differential Equations*. Englewood Cliff, N.J.: Prentice-Hall, Inc., 1971.
- [29] R. K. Brayton and C. H. Tong, "Stability of dynamical systems: A constructive approach," *IEEE Trans. Circuits Syst.*, vol. CAS-26, no. 4, pp. 224–234, Apr. 1979.
- [30] L. O. Chua and P.-M. Lin, *Computer-Aided Analysis of Electronic Circuits: Algorithms and Computational Techniques*. Englewood Cliff, NJ: Prentice-Hall, 1975.
- [31] J. Roychowdhury, "Making Fourier-envelope simulation robust," in *Proc. Int. Conf. Computer-Aided Design*, 2002, pp. 240–245.
- [32] K. E. Brennan, S. L. Campbell, and L. R. Petzold, *Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations, Classics in Applied Mathematics*, 1st ed. Philadelphia, PA: SIAM, 1996, vol. 14.
- [33] U. M. Ascher and L. R. Petzold, *Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations*. Philadelphia, PA: SIAM, 1998.
- [34] J. C. Butcher, *The Numerical Analysis of Ordinary Differential Equations*. New York: Wiley, 1987.
- [35] J. D. Lambert, *Numerical Methods for Ordinary Differential Systems: The Initial Value Problem*. New York: Wiley, 1991.
- [36] E. J. Grimme, "Krylov projection methods for model reduction," Ph.D. dissertation, Dept. Elect. Eng., Univ. Illinois, Urbana–Champaign, 1997.
- [37] R. W. Freund, "Reduced-order modeling techniques based on Krylov subspaces and their use in circuit simulation," *Appl. Computat. Control, Signals, Circuits*, vol. 1, pp. 435–498, 1999.
- [38] J. Roychowdhury, "Reduced-order modeling of time-varying systems," *IEEE Trans. Circuits Syst. II: Analog Digit. Signal Process.*, vol. 46, no. 10, pp. 1273–1288, Nov. 1999.
- [39] Z. Zhang, Z. Chen, and J. Lau, "A 900-mHz CMOS balanced harmonic mixer for direct conversion receiver," in *Proc. IEEE Radio Wireless Conf.*, 2000, pp. 219–222.
- [40] D. A. Johns and K. Martin, *Analog Integrated Circuit Design*. New York: Wiley, 1997.



Ting Mei received the B.S. and M.S. degrees in electrical engineering from the Huazhong University of Science and Technology, Wuhan, China, in 1998 and 2001. She is currently pursuing the Ph.D. degree at the University of Minnesota, Twin City.

Her research interests are in circuit- and system-level analog, RF, and mixed-signal verification.



Jaijeet Roychowdhury (S'85–M'87) received the Bachelor's degree in electrical engineering from the Indian Institute of Technology, Kanpur, India in 1987, and the Ph.D. degree in electrical engineering and computer science from the University of California, Berkeley, in 1993.

He was with the CAD Lab, AT&T Bell Laboratories, Allentown, PA from 1993 to 1995. From 1995 to 2000, he was with the Communication Sciences Research Division, Lucent Bell Laboratories, Murray Hill, NJ. From 2000 to 2001, he was with CeLight, Inc., Silver Spring, MD, an optical networking startup. Since 2001, he has been with the Department of Electrical and Computer Engineering and the Digital Technology Center, University of Minnesota, Minneapolis. He currently holds ten patents. His professional interests include the design, analysis, and simulation of electronic, electrooptical, and mixed-domain systems, particularly for high-speed and high-frequency communications.

Dr. Roychowdhury was named an IEEE Circuits and Systems (IEEE CAS) Society Distinguished Lecturer for 2003–2004 and currently serves as the Officer-at-Large in IEEE's CANDE committee. He was cited for Extraordinary Achievement by Bell Laboratories in 1996. He was awarded four Best Papers Awards at international computer-aided design conferences.



Todd S. Coffey received the B.S. degree in pure and applied mathematics from Oregon State University, Corvallis, in 1995 and the Ph.D. degree in applied mathematics from North Carolina State University, Raleigh, in 2002.

He is currently a John von Neumann Fellow of computational science at Sandia National Laboratories, Albuquerque, NM. His professional interests include nonlinear solvers, time integration, and electrical circuit simulation.



Scott A. Hutchinson received the B.S.E.E., M.S.E.E., and Ph.D. degrees from New Mexico State University, Las Cruces, in 1988, 1989, and 1993, respectively.

He joined Sandia National Laboratories, Albuquerque, NM, as a Research Fellow in 1993 and became a Senior Member of the Technical Staff in 1994. At Sandia, he has worked on a variety of computational projects including inverse problems for electrocardiography, parallel Krylov iterative solution methods (Aztec), parallel finite-element

methods for reacting flows (MPSalsa), finite-element modeling of continuum plasmas, and parallel circuit modeling (Xyce).

Dr. Hutchinson is the recipient of the 1997 Research and Development 100 Award for the Aztec Library and was the Gordon Bell Prize finalist in 1994 and 1997.



David M. Day received the Ph.D. degree in mathematics from the University of California, Berkeley, in 1993.

He spent one year with the Mathematics Department, University of Kentucky. In 1995, he joined Sandia National Laboratories, Albuquerque, NM, where he is a member of the Technical Staff in the Computational Mathematics and Algorithms Department.

Dr. Day was a co-recipient of the 2002 Gordon Bell Award for Salinas, a massively parallel structural dynamics simulation code.