A Time-Domain Oscillator Envelope Tracking Algorithm Employing Dual Phase Conditions

Ting Mei, *Student Member, IEEE*, and Jaijeet Roychowdhury, *Senior Member, IEEE*

***Abstract*—Envelope-following methods face special challenges when applied to oscillators because of their fundamental property of dynamically changing frequencies. In this paper, we present a novel and robust approach for oscillator envelope following. Our method combines, unifies, and extends ideas from two prior oscillator envelope-following approaches, namely, Petzold's method and the warped multitime partial differential equation. Our technique uses two extra system unknowns, as well as two extra "phase condition" equations, to track quantities related to dynamical frequency/time-period changes. These advances confer significant robustness, without appreciable computational overhead. We validate our method on LC, ring, and crystal oscillators, accurately predicting frequency and amplitude modulations, as well as transient startup envelopes. Speedups of one to two orders of magnitude are obtained over traditional alternatives.**

***Index Terms*—Amplitude modulations, envelope following, frequency modulations (FMs), oscillators.**

I. Introduction

OSCILLATORS are important in many engineering and communication systems. For example, they are often used as time references in digital circuits and for information encoding in communication systems. As is well known, oscillator simulation presents challenges that traditional SPICE-like simulation (e.g., [1] and [2]) has difficulty in addressing effectively. Due to their marginal stability [3], small phase errors accumulate unboundedly during transient simulation. This leads to a much worse tradeoff between simulation timestep size and accuracy for oscillators than for nonautonomous circuits. This is especially true for high-Q oscillators, which often feature very slow and sensitive amplitude responses. To obtain even reasonably accurate results, extremely small step sizes can be required during simulation.

A. Envelope

The envelope of a highly oscillatory signal refers to its slowly varying characteristics such as gradual amplitude or frequency modulation (FM) of fast oscillations. When very slow envelopes are present in an oscillator, predicting waveforms by conventional time-stepping simulation can extremely be inefficient because of the widely separated time scales of the fast and slow components [4]. Simulation timesteps are constrained to remain small (especially for oscillators, as noted earlier) by the fast undulations of the signal. At the same time, the presence of slow envelopes requires that thousands or millions of these cycles be simulated, resulting in a long simulation time. Such problems are encountered in many practical design situations, e.g., when simulating startup/shutdown of oscillators (especially high-Q ones), FM in voltage-controlled oscillators (VCOs), phase-locked loops, injection pulling/locking by external signals, etc. It should be noted that, in such situations, designers are often directly interested in the slow envelopes themselves. The envelopes are often used to encode information: For instance, variations in the envelope of oscillator startup transients (i.e., changes in envelope shape) can be used to detect weak antenna signals in ultralow-energy transceiver design (e.g., [5]).

B. Previous Work

Over the past few decades, a variety of methods have been devised to solve for envelopes more efficiently than transient simulation. The majority of such techniques have focused on circuits that are not oscillators, i.e., where the circuit's frequency does not change. The earliest of such a technique, to our knowledge, is the time-domain envelope-following method proposed by Petzold in [6], which was later adapted for circuit simulation (e.g., [7]–[11]), with application to both transient and steady-state simulations. Envelope-following methods operate by utilizing the assumed smoothness of the envelope, together with the circuit's differential equations, to mathematically relate and then solve for faraway envelope samples, without the need for computing in detail the skipped cycles. Another class of techniques (Fourier-envelope methods [12]–[14]) combines frequency-domain harmonic balance (HB) and time-domain integration methods. These techniques solve for the slowly varying Fourier coefficients of fast oscillations. Recently, a family of methods based on multitime partial differential equations (MPDEs) [4], [15], [16] has also emerged. These methods rely on separating slow and fast variations by employing several artificial time variables.

The fundamental idea behind all time-domain envelope-following methods is to sample consecutive cycles of the oscillatory waveform in a phase-synchronized manner and to interpolate between these samples to obtain a waveform called

Manuscript received May 13, 2006; revised March 20, 2007. This work was supported by the National Science Foundation under Awards 0515227, 0204278, and 0312079, by the Sandia National Laboratories, by the Semiconductor Research Corporation, and by MARCO/FCRP. This paper was recommended by Associate Editor C.-J. R. Shi.

T. Mei was with the Electrical and Computer Engineering Department, University of Minnesota, Minneapolis, MN 55455 USA. She is now with Sandia National Laboratories, Livermore, CA 94551-0969 USA (e-mail: meiting@ece.umn.edu).

J. Roychowdhury is with the Electrical and Computer Engineering Department, University of Minnesota, Minneapolis, MN 55455 USA (e-mail: jr@umn.edu).

Digital Object Identifier 10.1109/TCAD.2007.907259

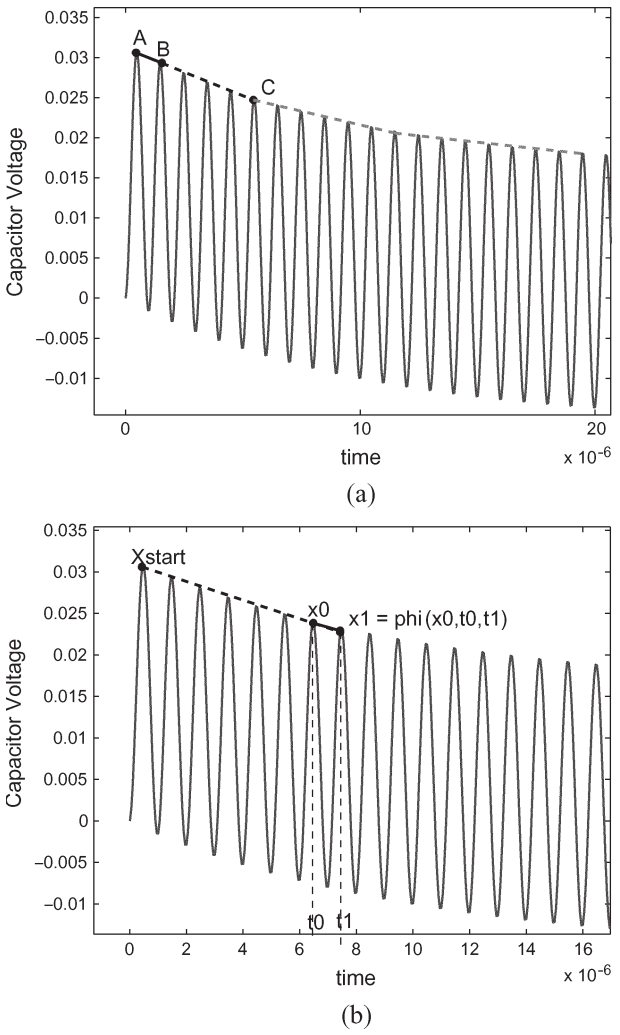

Fig. 1. Illustration of envelope and Petzold's method. (a) Forward-Euler-based envelope-following method. (b) Backward-Euler-based envelope-following method.

the envelope. For example, a popular choice is to sample and connect the peaks of each fast cycle [see Fig. 1(a)]. A key feature for envelope following to be useful and meaningful is that the envelope should be smooth and slowly varying, in which case efficiency potentially can be gained by skipping a large number of fast cycles between samples of the envelope. To abstract the smoothness correctly, it is critical that the points sampled for the envelope will all be "at the same point" within the fast cycles. For nonautonomous systems (i.e., the time period T of all fast cycles is the same and known *a priori*), the issue of where to sample the envelope within each cycle is easy to resolve—the sampling points can simply be integer multiples of T (plus any constant phase shift), i.e., an integer number of cycles are skipped between envelope samples.

For autonomous systems (oscillators), however, correct phase-synchronized sampling of the fast cycles is a much more complex problem. The reason is that (unlike for nonautonomous problems) the period T of the fast cycles is not known *a priori*, but is an unknown that needs to be solved—this is a fundamental property of oscillators. Furthermore, the period T is subject to change continuously during circuit operation as the oscillator is disturbed by external influences—indeed, this feature of oscillators, called FM, is extremely important in applications. Because of these reasons, separating envelope samples by an integer number of cycles becomes more complicated. The problem has two components:

1) finding the unknown (and changing) period of the few fast cycles that are chosen for sampling during envelope simulation;
2) correctly accounting for the cumulative effect of changing periods of the cycles skipped during envelope simulation so as to still skip an integer number of cycles between envelope samples.

To our knowledge, a prior work on envelope following for oscillators has concentrated only on the aforementioned first component.[1] In [6], Petzold proposed a technique in tracking the changing period by minimizing the difference between values at the beginning and at the end of a period. Since the cost of such minimization (via optimization) can be high, Gear and Gallivan [17] proposed a heuristic to alleviate this issue, identifying periods using zero crossings.

The second component of the envelope-sample synchronization problem for oscillators noted earlier, however, appears not to have been clearly recognized or addressed in prior work. As shown in this paper, estimating the correct phase within the cycle to be sampled is important. Any errors rapidly compound and lead to loss of smoothness during envelope computation, thereby destroying the robustness and effectiveness of prior envelope simulation methods for oscillators.

C. Contribution of This Paper

The contribution of this paper is to identify this crucial issue clearly and to propose an effective method to resolve it. Our method introduces two extra system unknowns to help find the correct sampling points. One unknown T represents the time period of the next cycle to be envelope-sampled, which is a concept similar to those in prior envelope methods [6], [17], [18]. In addition to this, however, a second unknown $T_{\rm env}$ represents the envelope timestep. This envelope timestep is calculated implicitly as part of the envelope-following algorithm such that each envelope step exactly spans an integral number of skipped cycles, in spite of the fact that their periods are unknown and changing. It is this feature that ensures that, even in the presence of dynamic FM effects, the envelope is sampled smoothly and correctly.

Addition of two extra unknowns leads to an underdetermined system of envelope-following equations. To obtain a square system and solve uniquely for these unknowns, we add two "phase condition" equations,[2] which directly enforce that the points being sampled for the envelope are phase-synchronized. We will call this method "Multiple-Phase Condition-based ENVelope following" (MPCENV).

[1]For Fourier and MPDE-based envelope methods, they do not have the problem of accounting for the cumulative effect of changing periods. However, they face the problem of finding good envelope initial conditions.

[2]We use derivative- and value-based phase conditions in this paper, but our method can also use any other phase conditions.

Because it does not rely on minimization, MPCENV's computational efficiency is similar to that of nonoscillatory envelope methods. By automatically taking good care of the problem of choosing envelope steps correctly via the extra variables, MPCENV achieves a considerable robustness while retaining the computational efficiency expected of envelope methods. We report results from validating MPCENV on several types of LC and ring oscillators, analyzing FM and amplitude modulation, as well as slow startup transients. Our simulations confirm excellent matches between MPCENV and carefully conducted traditional transient runs, with speedups of one to two orders of magnitude.

It is also interesting to compare MPCENV with another family of oscillator envelope methods, based on generalizing MPDE approaches using the notion of "warping" time to even out the changing periods of fast cycles. This approach, termed as warped MPDE (WaMPDE) [18], is appealing not only because of its theoretical elegance but also on account of its use of an explicit unknown variable, the local frequency, that is solved along with all other waveforms to capture the slowly changing time period (or frequency) of the oscillator. A problem with the WaMPDE, however, is that of choosing envelope initial conditions (ICs) [19]; unsuitable choices can severely compromise computational efficiency, which is the primary motivation for envelope simulation in the first place.

MPCENV can be viewed as combining the advantages of Petzold-style methods and the WaMPDE while eliminating their disadvantages. From the Petzold point of view, MPCENV automatically chooses correct envelope steps regardless of dynamic frequency changes; from the WaMPDE point of view, the envelope IC problem is sidestepped, while the notion of using system unknowns to capture changing frequencies is retained and generalized.

D. Organization of This Paper

The remainder of this paper is organized as follows. In Section II, we briefly review Petzold-type envelope-following methods and discuss issues for oscillator envelopes. In Section III, we provide related background on the MPDE and WaMPDE methods. In Section IV, we present the MPCENV method and discuss its relation to the WaMPDE. In Section V, we report results from applying MPCENV to several oscillators and VCOs and investigate startup transients, amplitude modulation, and FM.

II. Envelope-Following Methods

In this section, we first review existing envelope-following methods for nonoscillatory circuits and then discuss extensions to oscillators [6], [17].

A. Nonautonomous Envelope-Following Methods

A circuit can be described by the system of differential equations

$$\dot{q}(x) + f(x) = b(t) \tag{1}$$

where x is a vector of state variables (such as node voltages and branch currents), q contains capacitor charge or inductor flux terms, and f represents resistive terms [16].

We assume that the solution of the circuit to be simulated has fast oscillations, whose amplitude changes much more slowly than the oscillations themselves. When this solution is sampled at every fast oscillation period T, the resulting samples can be interpolated using a slowly varying curve, termed the envelope. The basic idea of envelope-following methods is shown in Fig. 1(a) using a scalar differential algebraic equation (DAE) for illustration. We start our simulation at time t_0, at which the state variable has a value x_0 (point A in the figure). Transient simulation is performed accurately for one cycle of the fast oscillation, and the state variable is now at point B shown in the figure. A secant line between A and B is drawn and then extrapolated over a large "envelope timestep," which can comprise many fast cycles, to reach the solution at point C. This process is repeated until the end of the simulation interval.

The process described earlier is analogous to solving the envelope by conventional forward-Euler integration. It uses the slope between points A and B to approximate the derivative of the envelope solution at point A. This is a valid approximation if the envelope varies much more slowly than the fast oscillation. If the envelope step is $(m-1)T$ (time interval between points B and C), the process of envelope following can be described by the difference equation

$$x(t+mT) = x(t) + mT\frac{x(t+T) - x(t)}{T}. \tag{2}$$

Although this forward-Euler-based envelope following is easy to illustrate, it is not very useful in practice since the envelope step cannot be very large due to stability issues, just as with the normal forward-Euler integration method [20]. A more stable backward-Euler-based envelope-following method is given by

$$x(t+mT) = x(t) + mT\frac{x(t+mT) - x\left(t+(m-1)T\right)}{T}. \tag{3}$$

When implicit methods are used, the only unknown is $x(t+(m-1)T)$; the state at $t+mT$ can be evaluated by integrating (1) for one fast cycle and can be written using the state transition function ϕ [21]–[25] as

$$x(t+mT) = \phi\left(x\left(t+(m-1)T\right),\ t+(m-1)T, t+mT\right). \tag{4}$$

Equation (3) is a boundary-value problem, shown in Fig. 1(b), which can be solved by any nonlinear solution method such as Newton–Raphson [26]. An initial value of the unknown x_0 is guessed, and then, a cycle of transient simulation is performed to obtain the state x_1. This information, together with the monodromy matrix $\partial x_1/\partial x_0$ [22], is used to update x_0 using Newton's method until the boundary condition $x_1 - x_{\text{start}} = m(x_1 - x_0)$ is satisfied.

The outlined nonautonomous envelope-following method has met with considerable success in various disciplines, including tracking of circuit envelopes, as reported in [7]–[11].

B. Extensions to Oscillators

Envelope-following methods involve integrating (1) for one cycle T. For oscillators, however, T is not known *a priori*, and it is important that it is accurately calculated. All prior work on oscillator envelope methods has focused on this problem.

Petzold [6], motivated by the observation that $\|y(t+T) - y(t)\| = 0$ for T-periodic $y(t)$, proposed that T should be estimated for nonperiodic $y(t)$ by minimizing $\int \|y(t+T) - y(t)\|_2 dt$. However, as noted by Gear and Gallivan [17], performing this minimization is not only expensive but also prone to error buildup, presumably due to the explicit estimation of the period. An alternative approach toward estimating the period was proposed in [17] using zero crossings of the waveform levels or their derivatives.

However, estimation of the envelope step is left to the time-stepping algorithm, just as in normal numerical solution of differential equations, in both aforementioned methods. This can be a cause of reduced robustness when the time period is varying, because the envelope step may not exactly be an integer number of cycles. It is critical that this envelope step be accurately estimated, for essentially the same reason that T needs to be found accurately, i.e., in order to properly "line up" points A, B, and C in Fig. 1(a) and to obtain a smooth envelope. For this reason, prior envelope methods have been limited to taking relatively small envelope timesteps, since not accounting for the cumulative phase error appears as a lack of smoothness in the envelope calculated.

III. MPDE-Based Envelope Methods

In this section, we review envelope methods based on multitime concepts, including the nonautonomous (MPDE) and autonomous (WaMPDE) cases, and touch on the problem of finding good ICs.

A. MPDE Methods for Widely Separated Time Scales

In the MPDE formulation, artificial time scales are introduced to decouple slow and fast time scales [4], [16]. Each rate of variation is represented by its "own" time scale so that it can be solved using its "own" timestep size. Thus, the envelope, i.e., the slow component in a signal, can efficiently be solved since large timesteps can be used for its time scale. Consider two time scales: one fast and one slow. The MPDE form corresponding to (1) is

$$\frac{\partial q(\hat{x})}{\partial t_1} + \frac{\partial q(\hat{x})}{\partial t_2} + f(\hat{x}) = \hat{b}(t_1, t_2) \tag{5}$$

where $\hat{x}(t_1, t_2)$ and $\hat{b}(t_1, t_2)$ are the bivariate forms of $x(t)$ and $b(t)$, respectively. Here, without loss of generality, we choose t_1 to be the slow time scale and t_2 to be the fast time scale.

The envelope equation that results from the MPDE [19] is essentially a DAE in the slow envelope time scale. Taking a timestep along the envelope time scale involves solving a steady-state problem with periodic boundary conditions along the fast time scale [16]. To start the DAE solution along the envelope time scale, however, a periodic solution along the fast time scale needs to be given at $t_1 = 0$ (the initial starting point of the envelope simulation) along the envelope time scale. It has been shown [19] that proper choice of this envelope IC, which involves heuristics, is crucial to the efficiency of the MPDE-based and Fourier envelope solution.

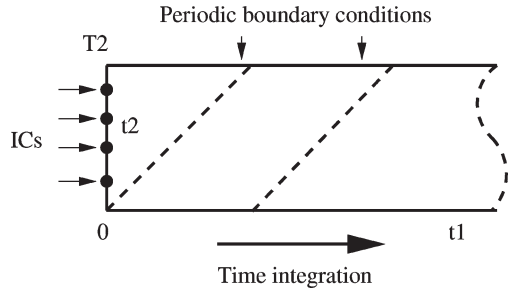

Fig. 2. Solution process of the MPDE.

In the numerical solution of the MPDE, the fast time scale is first discretized by either time-domain methods, such as finite difference (FD) time-domain method, or frequency-domain methods, such as HB. A DAE in the slow time scale is then formed and can be solved using any time integration method such as backward-Euler. Single time solutions can be recovered from MPDE solutions by interpolating solutions along the "diagonal" lines, i.e., solutions at $t_1 = t_2$. The envelope solutions are the solutions along the slow time scale.

Fig. 2 shows the solution process of the MPDE. As can be seen, N (the number of points used to discretize the fast time scale) ICs (along $t_1 = 0$) are required to solve the slow time DAE. Only the IC at $t_1 = t_2 = 0$ is "free" since it is the IC for the transient simulation. Other $N - 1$ ICs need to be chosen properly; otherwise, envelope solutions (along the slow time scale) will show rapid oscillations which compromise the efficiency of the MPDE methods [19]. Techniques to alleviate this problem are also proposed in [19].

B. Warped MPDE (WaMPDE)

The MPDE is not well suited for analyzing FM in oscillators. To remedy this situation, the WaMPDE formulation was devised in [18]. In the WaMPDE, the fast time scale is dynamically rescaled (or warped) to undo FM and make the fast undulations uniform. The resulting warped multivariate waveforms can compactly be represented (as for circuits without FM), while the rescaled fast time scale captures the effects of FM. At the equation level, the WaMPDE is formed by adding an extra unknown (representing the instantaneous frequency) to the MPDE as [18]

$$\frac{\partial q(\hat{x})}{\partial t_1} + \omega(t_1)\frac{\partial q(\hat{x})}{\partial t_2} + f(\hat{x}) = \hat{b}(t_1, t_2) \tag{6}$$

where t_1 is the unwarped (slow) time scale, and t_2 is the warped (fast) time scale.

To solve the WaMPDE, an extra equation must be added to the system since there is one more unknown than there are equations. We note that (6) is autonomous in the t_2 scale, i.e., a time shift in t_2 from any solution $x(t_1, t_2)$ is also a valid solution. Uniqueness of the solution is enforced by adding

a phase condition, which fixes the phase of one variable at, e.g., $t_2 = 0$. For example, the phase condition can be specified to be

$$\left.\frac{d\hat{x}_l(t_1, t_2)}{dt_2}\right|_{t_2=0} = 0 \tag{7}$$

where $\hat{x}_l(t_1, t_2)$ is one of the state variables. The WaMPDE can then be solved with numerical methods similar to those for the MPDE.

The DAE solution can be recovered using the following relations:

$$x(t) = \hat{x}\left(t, \phi(t)\right) \tag{8}$$

where

$$\phi(t) = \int_0^t \omega(t_1) dt_1. \tag{9}$$

By adding one unknown and equation, the WaMPDE achieves efficiency as in the MPDE without much extra computational expense.

The WaMPDE also faces the same problem of finding good envelope ICs, just as for the MPDE. Finding good envelope ICs for oscillators is, indeed, typically more difficult due to the unknown period of the oscillator. The techniques in [19] are not immediately extensible to oscillators.

IV. MPCENV Method

In this section, we propose an MPCENV for oscillators. The connection between MPCENV and WaMPDE is also discussed.

A. Extra Unknowns and Phase-Condition Equations

In MPCENV, we introduce two extra unknowns: T to represent the changing period of the oscillator and $T_{\rm env}$ (the envelope step) to capture the effect of FM—since periods of small cycles will typically vary, although slowly, within an envelope step. In other words, the envelope step will, in general, no longer remain an integer number of the period T (at any given time point). With these two extra unknowns, the envelope-following equation system [for example, the backward-Euler-based one (3)] becomes under-determined since the number of unknowns becomes more than that of equations.

Our solution to this is directly motivated by our goal of keeping the phases at the beginning and at the end of a fast cycle (also the beginning of the next cycle) the same. Recall that, for the nonoscillatory case, this is automatically satisfied because the period of the fast oscillation is known (and a fixed constant over the simulation). For the oscillator case, since the period is unknown and changing, these conditions must be enforced. For example, we can use a phase condition similar to (7) at both the beginning and the end of the fast cycle, over which a standard transient simulation is performed during envelope following. Only the phase of one of the state variables needs to be fixed. Using the backward-Euler-based envelope-following method as an example, we add these two phase conditions to x_0 and x_1, as shown in Fig. 1(b).

We now have a well-determined system with equal numbers of equations and unknowns, which can be solved by nonlinear solvers such as Newton's method. The augmented backward-Euler-based envelope-following equation system is

$$\begin{aligned} \frac{x(t + T_{\rm env} + T) - x(t + T_{\rm env})}{T} &= \frac{x(t + T_{\rm env}) - x(t)}{T_{\rm env}} \\ \left.\frac{dx_l}{dt}\right|_{t+T_{\rm env}} &= 0 \\ \left.\frac{dx_l}{dt}\right|_{t+T_{\rm env}+T} &= 0 \end{aligned} \tag{10}$$

where x_l is the state variable to which the phase constraints are applied.

Note that the IC $x_l(0)$ should be chosen such that it also satisfies the phase condition applied. This is not difficult to achieve; for example, a transient simulation can be run for a few initial cycles to choose an appropriate point for this IC. In our implementation, we first use transient simulation to locate the time interval on which signs of the derivative of x_l change. We then halve the timestep and perform transient simulation on this short time interval. This procedure is repeated until the IC satisfies the phase condition with the chosen accuracy.

B. Numerical Solution of MPCENV

MPCENV is a boundary value problem with two more phase constraints and can be solved by using Newton's method. At each MPCENV step, the unknowns are $x(t + T_{\rm env})$, T, and $T_{\rm env}$. We denote the unknown state variables $x(t + T_{\rm env})$ with x_0, the corresponding time point as t_0 ($t_0 = t + T_{\rm env}$), and the starting states $x(t)$ (which are either ICs, given at $t = 0$, or known from the previous step) as x_s. We can rewrite (10) using a state transition function notation as

$$\begin{aligned} \frac{\phi(x_0, t_0, t_0 + T) - x_0}{T} &= \frac{x_0 - x_s}{T_{\rm env}} \\ \frac{dx_{0l}}{dt} &= 0 \\ \frac{d\phi(x_0, t_0, t_0 + T)_l}{dt} &= 0. \end{aligned} \tag{11}$$

Note that the phase-condition equations in (10) and (11) are based on derivatives, which detect peaks/troughs in a particular circuit waveform, and use them to aid in the determination of T and $T_{\rm env}$. Derivative-based phase conditions are not the only ones possible; indeed, value-based or other phase conditions can also be used for MPCENV and may be better suited for certain kinds of circuits. For example, if the waveform of a circuit variable (normally the inductor current in an oscillator) is symmetric about a certain value (e.g., zero), we can use the

phase condition $x_l|_{t=t_c} = C$, where C is a constant. Then, the backward-Euler-based MPCENV system becomes

$$\begin{aligned} \frac{\phi(x_0, t_0, t_0+T) - x_0}{T} &= \frac{x_0 - x_s}{T_{\text{env}}} \\ x_{0l} &= C \\ \phi(x_0, t_0, t_0+T)_l &= C. \end{aligned} \tag{12}$$

Indeed, value-based phase conditions can work better than derivative-based ones when waveforms of interest feature large "flat" sections (e.g., for relatively ideal ring oscillators). The value-based phase condition is also somewhat easier to solve numerically than derivative-based ones because derivative-based phase constraints (11) are approximated by FD methods. In our implementation, since the phase constraints are applied to the derivatives of both ends of a period, it is natural to use the backward difference method at $t_0 + T$ and the forward difference method at t_0. The phase constraints approximated by FD can be difficult to satisfy, making the whole system difficult to solve. One can also approximate both phase constraints by centered difference, with a little change in the first equation of (11). However, the phase constraints approximated by centered difference are still difficult to satisfy. In our experiment, only very simple circuits (e.g., circuit in Section V-A) can be solved using (11). Better numerical handling of phase constrains (11) and their corresponding stability property are currently under investigation.

We can reorganize the MPCENV system (12) in order to apply Newton's method

$$\begin{aligned} (\phi(x_0, t_0, t_0+T) - x_0)\, T_{\text{env}} - (x_0 - x_s)T &= 0 \\ x_{0l} - C &= 0 \\ \phi(x_0, t_0, t_0+T)_l - C &= 0. \end{aligned} \tag{13}$$

This is a system of $n+2$ equations and $n+2$ unknowns, where n is the number of circuit variables. $\phi(x_0, t_0, t_0+T)$ can be evaluated by integrating one cycle T starting from x_0.

The Jacobian matrix is given by

$$J = \begin{pmatrix} \frac{\partial \phi()}{\partial x_0} T_{\text{env}} - (T_{\text{env}}+T) I_n & \frac{\partial \phi()}{\partial T} T_{\text{env}} - (x_0 - x_s) & \phi() - x_0 \\ I_n|_l & 0 & 0 \\ \frac{\partial \phi()}{\partial x_0}|_l & \frac{\partial \phi()}{\partial T}|_l & 0 \end{pmatrix} \tag{14}$$

where I_n is an identity matrix of size $n \times n$, $I_n|_l$ is the lth row of I_n, and $[\partial \phi()/\partial x_0]|_l$ is the lth row of $\partial \phi()/\partial x_0$.

The calculation of the Jacobian matrix involves the evaluation of the derivative of the state transition function, also known as the sensitivity matrix. It represents the sensitivity of $\phi(x_0, t_0, t_0+T)$ to changes of both x_0 and T. The evaluation of the sensitivity matrix is performed during transient simulation, with a little additional computation, just as for the shooting method [28]. Indeed, if we assume that the sampled envelope does not change with time, then the boundary condition in (10) becomes a simple periodic one, resulting in the shooting method.

We apply the trapezoidal integration method (better suited for oscillators since overly stable methods like backward-Euler can damp out oscillations) to integrate (1) from t_0 to $t_0 + T$. Both $d\phi/dx_0$ and $d\phi/dT$ can be derived as described in [23]. If x_n is the state at t_n ($t_0 \le t_{n-1} < t_n \le t_0 + T$), then

$$\frac{dx_n}{dx_0} = \left(\frac{C_n}{h_n} + \frac{G_n}{2}\right)^{-1} \left(\frac{C_{n-1}}{h_n} - \frac{G_{n-1}}{2}\right) \frac{dx_{n-1}}{dx_0} \tag{15}$$

where $C_i = dq(x_i)/dx_i$, $G_i = df(x_i)/dx_i$, and $h_i = t_i - t_{i-1}$. Note that $C_n/h_n + G_n/2$ is the Jacobian matrix of (1) during the integration and is already available from transient simulation. Starting from $dx_0/dx_0 = I$, $d\phi/dx_0$ can be found by repeatedly applying (15) until $t_0 + T$ is reached.

Similarly, $d\phi/dT$ can be found by repeatedly applying

$$\begin{aligned} \frac{dx_n}{dT} = &\left(\frac{C_n}{h_n} + \frac{G_n}{2}\right)^{-1} \\ &\times \left[\left(\frac{C_{n-1}}{h_n} - \frac{G_{n-1}}{2}\right) \frac{dx_{n-1}}{dT} + \frac{q(x_n) - q(x_{n-1})}{T}\right] \end{aligned} \tag{16}$$

starting from $dx_0/dT = 0$.

C. Relationship With WaMPDE

The relation between MPCENV and WaMPDE is shown in Fig. 3. As mentioned in Section III-B, the solution of DAE (1) is available along the characteristics, i.e., $x(t) = \hat{x}(t, \phi(t))$. Note that the WaMPDE solution is periodic along t_2 time scale. The shape of $\phi(t)$ represents the effect of FM in oscillators. If $\phi(t) = t$, then the oscillator frequency is the free-running frequency of an oscillator. If $\phi(t)$ becomes sharp, then the oscillator frequency is high. For illustration purposes, we still use backward-Euler-based MPCENV. We start from x_0 and then perform a transient simulation for one fast cycle to obtain solution x_1. This is equivalent to tracking the solution along $t_1 = t, t_2 = \phi(t)$ starting from $(t_0, \phi(t_0))$ to $(t_0 + T, \phi(t_0 + T))$ in the WaMPDE solution. Due to the periodic boundary condition along t_2, $x_1' = x_1$. In the WaMPDE, we apply a phase constraint on all $t_2 = 0$. This automatically applies to both x_0 and x_1. We achieve the same effect by adding two phase constraints on both x_0 and x_1.

As shown in Fig. 3, the solution obtained from MPCENV is actually the WaMPDE solution along $t_2 = 0$. That is the reason why we only need ICs at $t_1 = t_2 = 0$ in MPCENV. On the other hand, by introducing more information in the system, the WaMPDE has the freedom to take any length of step size along t_1, with phase conditions applied on all $t_2 = 0$. MPCENV takes the envelope step size of integer number of cycles, with phase conditions explicitly added on some specific points.

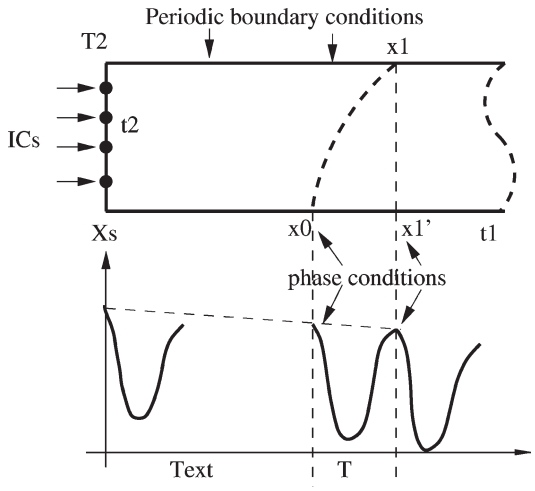

Fig. 3. Relation between MPCENV and WaMPDE.

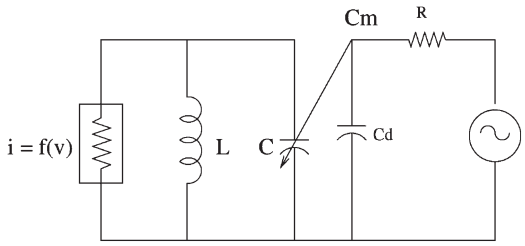

Fig. 4. Circuit schematic of an LC VCO.

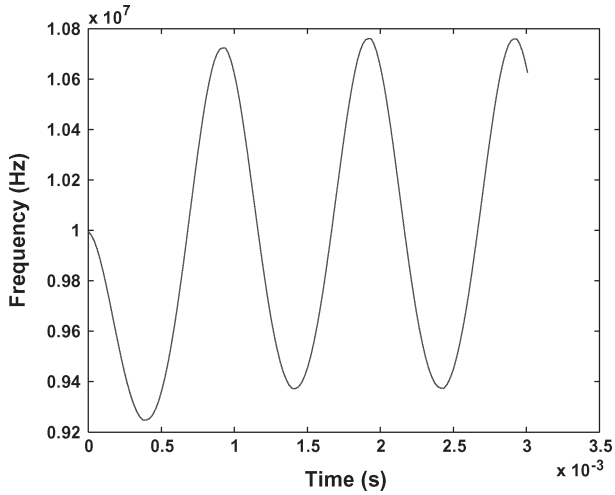

Fig. 5. LC VCO: FM.

V. Applications and Validation

In this section, we apply and validate MPCENV on a variety of oscillators. We investigate FM, as well as amplitude envelopes, and also simulate startup transient envelopes in a high-Q crystal-based oscillator. MPCENV results show excellent matches with those from traditional SPICE-like transient simulation while delivering speedups of one to two orders of magnitude. All simulations were performed using MATLAB on a 2.4-GHz PC running Linux.

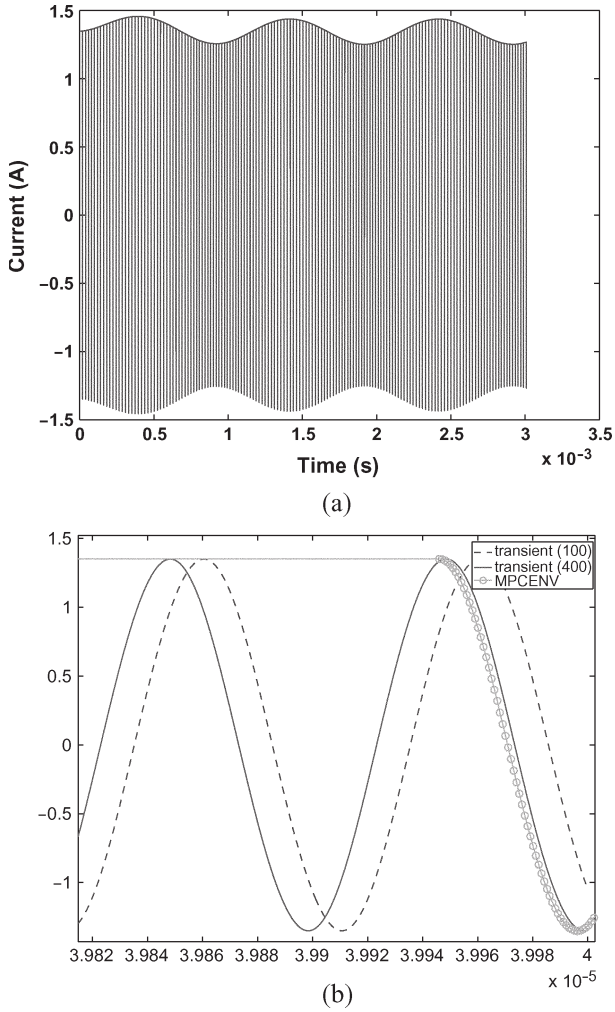

Fig. 6. LC VCO: Solution of the inductor current from MPCENV and comparison with transient simulation. (a) MPCENV solution. (b) Detailed comparison with transient simulation.

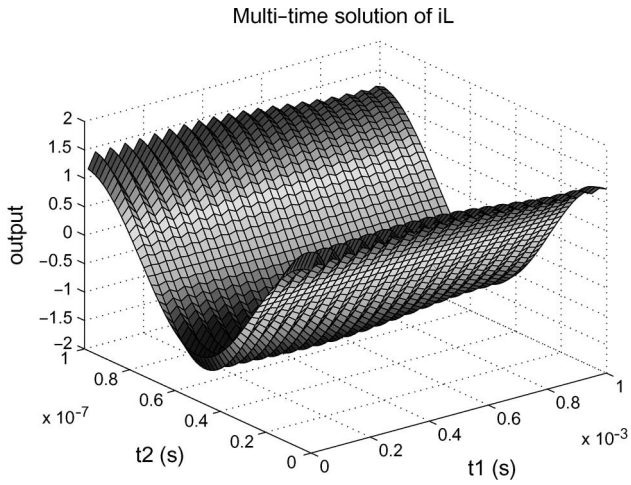

Fig. 7. WaMPDE solution of the inductor current.

A. LC VCO

A simple LC VCO from [18] is shown in Fig. 4 and is simulated using MPCENV. The oscillator contains an LC tank, with the capacitance controlled by a voltage source. The

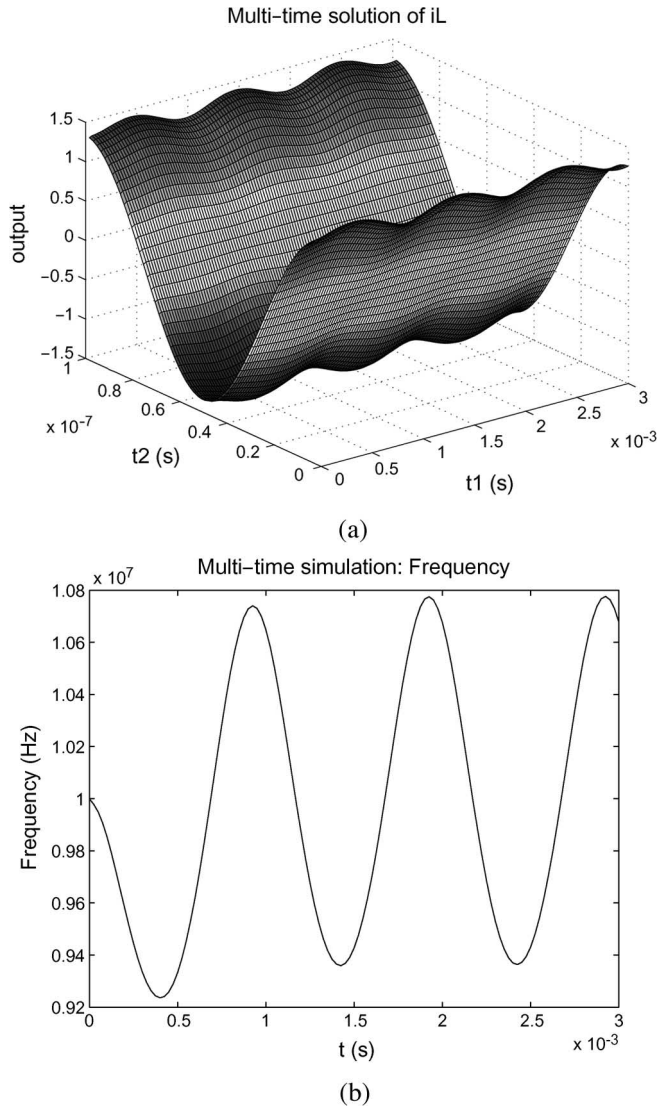

Fig. 8. WaMPDE solution from good ICs. (a) Inductor current. (b) FM.

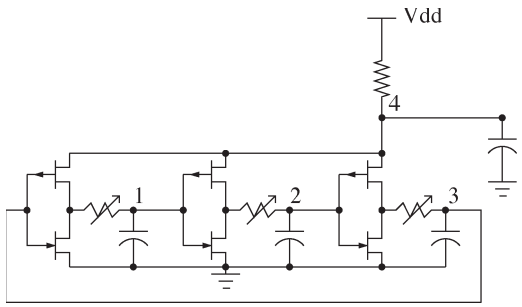

Fig. 9. Circuit schematic of a ring VCO.

element values are $R = 1$ kΩ, Cd $= 0.3$ uF, $C = 1/2\pi 10^{-7}$ F, $L = 1/2\pi 10^{-7}$ H, and Cm $= 1/4\pi 10^{-7}$ F. The nonlinear negative resistor characteristic is given in [18] as

$$i = f(\nu) = (G_0 - G_\infty)V_k \tanh\left(\frac{\nu}{V_k}\right) + G_\infty \nu \qquad (17)$$

where $G_0 = -0.1, G_\infty = 0.25$, and $V_k = 1$. The oscillator has a nominal frequency of 10 MHz.

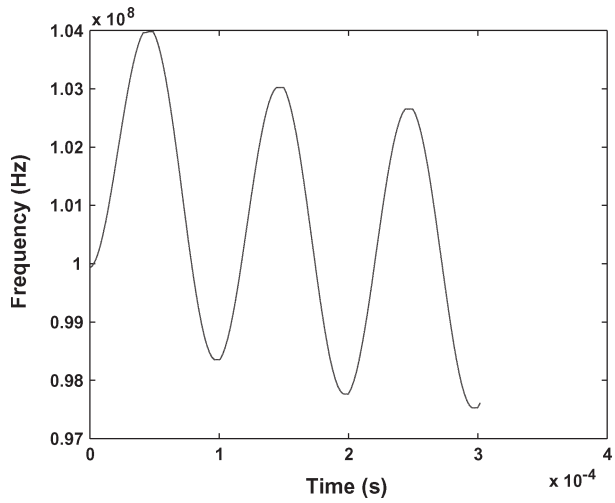

Fig. 10. Ring VCO: FM.

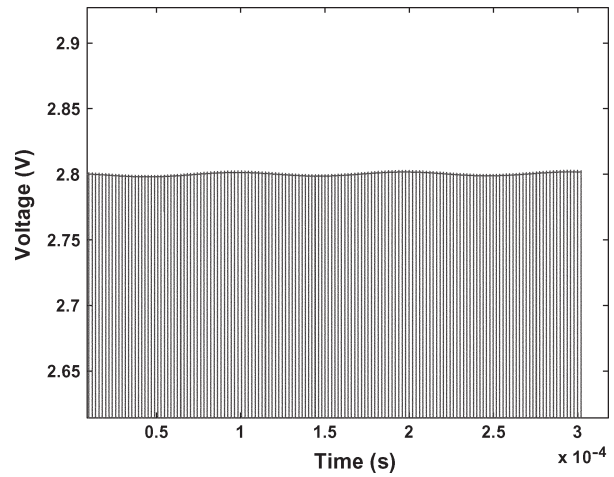

Fig. 11. Ring VCO: Envelope solution of the output of the third stage.

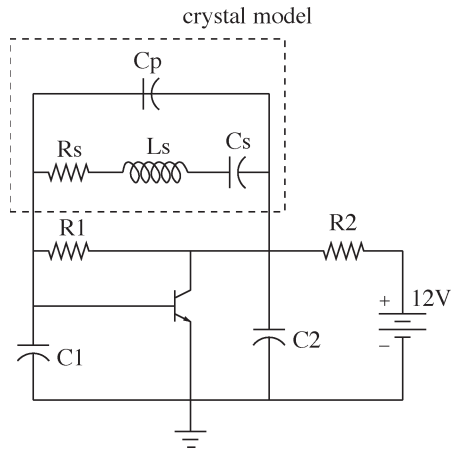

Fig. 12. Circuit schematic of a Pierce crystal oscillator.

We start the envelope simulation from the oscillator's steady state. ICs are chosen so that they satisfy the phase condition in (10). The controlling voltage is a sinusoid with the frequency 10^4 times slower than the oscillator's free-running frequency ($V_c = 0.6\sin(2\pi 10^3 t)$). The main purpose of this simulation is to illustrate strong FM in VCOs and show how MPCENV

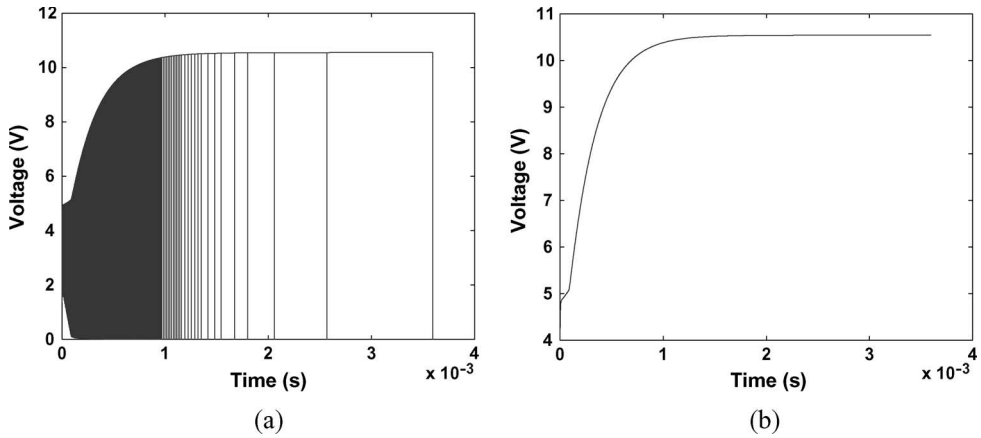

Fig. 13. Waveform at the collector of the BJT in the pierce crystal oscillator. (a) MPCENV solutions. (b) Envelope solution.

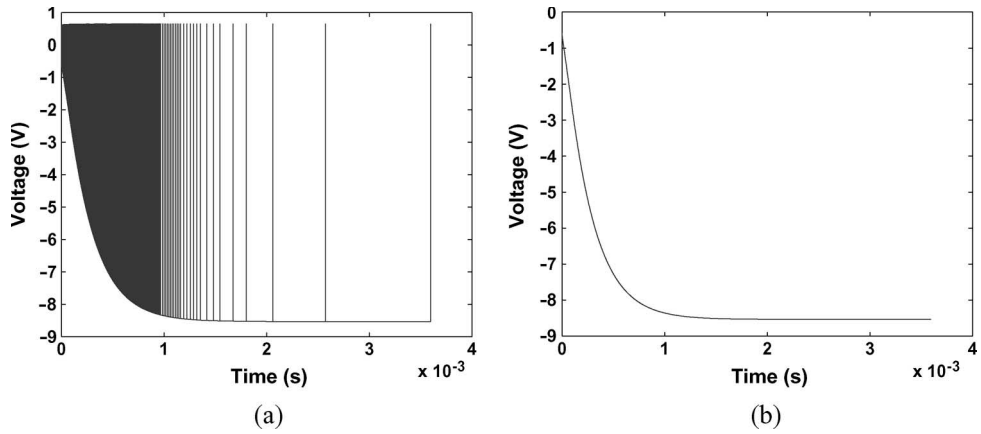

Fig. 14. Waveform at the base of the BJT in the Pierce crystal oscillator. (a) MPCENV solutions. (b) Envelope solution.

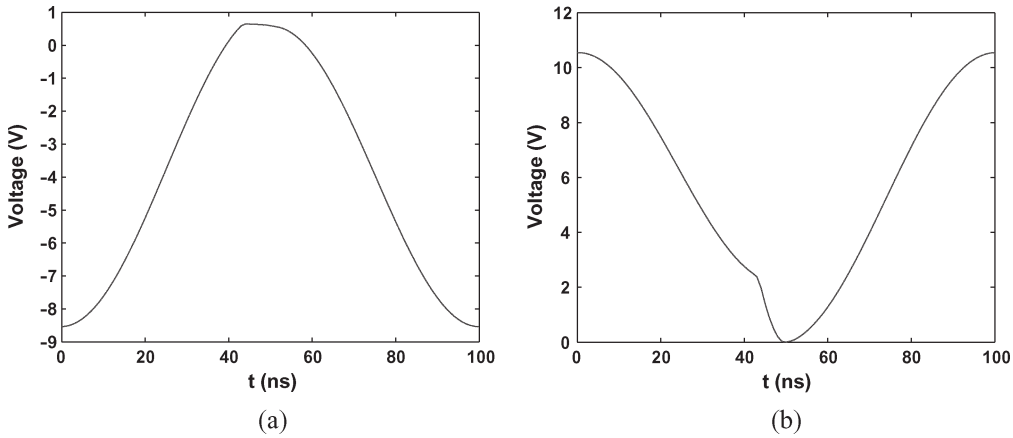

Fig. 15. Steady-state solutions. (a) Base of the BJT. (b) Collector of the BJT.

captures it. Fig. 5 shows the frequency change due to the variation of the capacitance in an LC tank. The solution of the inductor current from MPCENV is shown and compared with full transient simulation result in detail in Fig. 6. As can be seen, the MPCENV result with 100 timesteps per fast cycle matches the transient simulation result with 400 timesteps per fast cycle. This is because our MPCENV only performs transient simulation for a few selected fast cycles and skips a large number of fast cycles in each envelope step, thus avoiding phase error accumulation, which could have happened in oscillator transient simulation. In this example, MPCENV takes envelope steps of about 200 fast cycles each, solving the Newton equations with only two to three iterations at each step. A speedup of more than 160$\times$ over full transient simulation with similar accuracy is obtained for this example.

For comparison, we also solve the circuit using the WaMPDE method [18]. In our implementation, HB is used to discretize the fast time scale. This is equivalent to a Fourier envelope type of method [12]–[14]. As mentioned in Section III-B, the WaMPDE faces the problem of finding good ICs [19], as shown

in Fig. 7. In this experiment, we start from ICs which are very close to the good ICs, as we will show later. However, we obtain a wrong envelope solution with many undulations along the slow time scale. In comparison, we also show the WaMPDE results from slightly different envelope ICs. As shown in Fig. 8, the solutions from these ICs provide correct envelopes, as shown along the slow time scale. However, as we demonstrated, the WaMPDE methods are sensitive to envelope ICs, and it is not always easy to find good ICs.

B. Three-Stage Ring VCO

A three-stage ring VCO is shown in Fig. 9. Each stage is identical in this VCO. The resistance is varied by changing the controlling voltage; this changes the period/frequency of the oscillator. The oscillator has a free-running frequency of 100 MHz.

We choose a point in the oscillator's steady state as the IC for MPCENV, i.e., we start the simulation from steady state. The VCO's controlling voltage is $10 + 2\sin(2\pi 10^4 t)$. The simulation interval is 0.3 ms. A nominal envelope timestep of 200 cycles is used in this example. Fig. 10 shows the variation of the oscillation frequency as it responds to the controlling voltage. The MPCENV solution of the amplitude variation at the output of the third stage is shown in Fig. 11 and compared against transient simulation results. They are in good agreement. We do not show the transient simulation result here due to the density of the waveform. We obtain speedups of about $135\times$ for this example.

C. Startup-Transient Pierce Crystal Oscillator

Fig. 12 shows a Pierce crystal oscillator from [29] and [30]. The element values are $R_1 = 100$ KΩ, $R_2 = 2.2$ KΩ, $C_1 = 100$ pF, $C_2 = 100$ pF, $C_p = 25$ pF, $C_s = 99.5$ fF, $R_s = 6.4\ \Omega$, and $L_s = 2.55$ mH, resulting in a high quality factor Q about 2.5×10^4. The bipolar transistor has a current gain $\beta = 100$, and the oscillator's nominal frequency is around 10 MHz. Due to its high Q, the crystal oscillator takes many oscillatory cycles to reach its steady state from power-on startup.

In the simulation, we use variable envelope step sizes based on a very simple convergence criterion: If the envelope Newton converges in a few iterations, we increase the envelope step; otherwise, if Newton takes too many iterations, we shorten the step. Figs. 13 and 14 show waveforms obtained by MPCENV at the base and the collector of the bipolar junction transistor (BJT). As can be seen, the envelope step is small at the beginning due to a relatively fast-changing envelope. The envelope step gets larger as the oscillator approaches its steady state and the waveform stabilizes. Over the simulation, MPCENV takes an average envelope step of about 91 fast cycles. For a full transient simulation, it takes about 1 ms (10 000 cycles) to approach the steady state and about another 2 ms (20 000 cycles) to actually reach the steady state. We obtained speedups of 45 over the transient for this simulation.

Thus, we note in passing that MPCENV can also be used as a means for finding the steady-state solution of high-Q oscillators (e.g., [11]) since it accelerates the simulation of startup transients—although this is of course not its only capability. The steady-state solutions at both the base and the collector of the BJT are shown in Fig. 15; the frequency found by MPCENV is 9.995×10^6 Hz. Both the startup and the steady state solved by MPCENV perfectly match with the full transient simulation.

VI. Conclusion and Future Directions

We have presented MPCENV, a robust and efficient algorithm for oscillator envelope following, which is able to capture dynamic frequency changes with unprecedented accuracy and robustness compared to prior methods. MPCENV's flexibility with regard to different types of phase-condition equations enables it to work effectively on many different classes of oscillators. We have obtained speedups of one to two orders of magnitude over transient simulation.

At this point, what type of phase condition to use is decided manually, taking into account the type of oscillator being simulated. We are actively investigating extensions that remove the need for the user to specify phase-condition types. To better solve systems with flat waveforms, which can cause problems in solving MPCENV if derivative-based phase conditions are used, we are also investigating extensions that remove the need for phase conditions [27]. In addition, although we have not observed any stability-related problems with MPCENV in practice, rigorous theoretical analysis of MPCENV's stability properties remains a future topic for investigation.

Acknowledgment

The authors would like to thank the Digital Technology Center and the Supercomputing Institute, University of Minnesota, for the computational and infrastructural resources.

References

[1] L. W. Nagel, "SPICE2: A computer program to simulate semiconductor circuits," Ph.D. dissertation, EECS Dept., Univ. California, Berkeley, Electron. Res. Lab., Berkeley, CA, 1975. Memorandum no. ERL-M520.

[2] T. L. Quarles, *SPICE 3C.1 User's Guide*. Berkeley, CA: EECS Ind. Liaison Program, Univ. California, Berkeley, Apr. 1989.

[3] A. Demir and J. Roychowdhury, "A reliable and efficient procedure for oscillator PPV computation, with phase noise macromodeling applications," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 22, no. 2, pp. 188–197, Feb. 2003.

[4] J. Roychowdhury, "Efficient methods for simulating highly nonlinear multi-rate circuits," in *Proc. IEEE DAC*, 1997, pp. 269–274.

[5] B. Otis, Y. H. Chee, and J. Rabaey, "A 400μ-rx, 1.6mw-tx super-regenerative transceiver for wireless sensor networks," in *Proc. IEEE ISSCC Dig. Tech. Papers*, Feb. 2005, pp. 6–7.

[6] L. Petzold, "An efficient numerical method for highly oscillatory ordinary differential equations," *SIAM J. Numer. Anal.*, vol. 18, no. 3, pp. 455–479, Jun. 1981.

[7] K. Kundert, J. White, and A. Sangiovanni, "An envelope-following method for the efficient transient simulation of switching power and filter circuits," in *Proc. ICCAD*, 1988, pp. 446–449.

[8] J. White and S. B. Leeb, "An envelope-following approach to switching power converter simulation," *IEEE Trans. Power Electron.*, vol. 6, no. 2, pp. 303–307, Apr. 1991.

[9] L. M Silveira, J. White, and S. Leeb, "A modified envelope-following approach to clocked analog circuit simulation," in *Proc. ICCAD*, Nov. 1991, pp. 20–23.

[10] C. Deml and P. Turkes, "Fast simulation technique for power electronic circuits with widely different time constants," in *32nd Conf. Rec. IAS Annu. Meeting*, Oct. 1997, vol. 2, pp. 1204–1210.

[11] A. Brambilla and P. Maffezzoni, "Envelope-following method to compute steady-state solutions of electrical circuits," *IEEE Trans. Circuits Syst. I, Fundam. Theory Appl.*, vol. 50, no. 3, pp. 407–417, Mar. 2003.
[12] 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.*, 1996, pp. 1365–1368.
[13] D. Sharrit, "New method of analysis of communication systems," presented at the MTTS WMFA: Nonlinear CAD Workshop, Jun. 1996.
[14] P. Feldmann and J. Roychowdhury, "Computation of circuit waveform envelopes using an efficient, matrix-decomposed harmonic balance algorithm," in *Proc. ICCAD*, 1996, pp. 295–300.
[15] 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, no. 2, pp. 103–112, Apr. 1996.
[16] 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.
[17] C. W. Gear and K. Gallivan, "Automatic methods for highly oscillatory ordinary differential equations," in *Lecture Notes in Mathematics No. 912: Numerical Analysis*. Berlin, Germany: Springer-Verlag, 1981, pp. 115–124.
[18] O. Narayan and J. Roychowdhury, "Analyzing oscillators using multitime PDEs," *IEEE Trans. Circuits Syst. I, Fundam. Theory Appl.*, vol. 50, no. 7, pp. 894–903, Jul. 2003.
[19] J. Roychowdhury, "Making Fourier-envelope simulation robust," in *Proc. ICCAD*, 2002, pp. 240–245.
[20] C. W. Gear, *Numerical Initial Value Problem in Ordinary Differential Equations*. Englewood Cliff, NJ: Prentice-Hall, 1971.
[21] R. A. DeCarlo, *Linear Systems*. Englewood Cliffs, NJ: Prentice-Hall, 1989.
[22] M. Farkas, *Periodic Motions*. New York: Springer-Verlag, 1994.
[23] 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.
[24] S. Skelboe, "Computation of the periodic steady-state response of nonlinear networks by extrapolation methods," *IEEE Trans. Circuits Syst.*, vol. CAS-27, no. 3, pp. 161–175, Mar. 1980.
[25] L. O. Chua and P.-M. Lin, *Computer-Aided Analysis of Electronic Circuits: Algorithms and Computational Techniques*. Englewood Cliffs, NJ: Prentice-Hall, 1975.
[26] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes—The Art of Scientific Computing*. Cambridge, U.K.: Cambridge Univ. Press, 1989.
[27] T. Mei and J. Roychowdhury, "A robust envelope following method applicable to both non-autonomous and oscillatory circuits," in *Proc. 43rd DAC*, New York, 2006, pp. 1029–1034.
[28] K. S. Kundert, J. K. White, and A. Sangiovanni-Vincentelli, *Steady-State Methods for Simulating Analog and Microwave Circuits*. Norwell, MA: Kluwer, 1990.
[29] M. Gourary, S. Ulyanov, M. Zharov, S. Rusakov, K. K. Gullapalli, and B. J. Mulvaney, "Simulation of high-Q oscillators," in *Proc. ICCAD*, 1998, pp. 162–169.
[30] M. Gourary, S. Ulyanov, M. Zharov, S. Rusakov, K. K. Gullapalli, and B. J. Mulvaney, "A robust and efficient oscillator analysis technique using harmonic balance," *Comput. Methods Appl. Mech. Eng.*, vol. 181, no. 4, pp. 451–466, Jan. 2000.

Ting Mei (S'04) received the B.S. and M.Sc. degrees in electrical engineering from Huazhong University of Science and Technology, Wuhan, China, in 1998 and 2001, respectively, and the Ph.D. degree in electrical engineering from the University of Minnesota, Minneapolis, in 2006.

She is currently with Sandia National Laboratories, Livermore, CA. Her research interests include circuit- and system-level analysis and simulation of analog, RF, and mixed-signal systems.

Jaijeet Roychowdhury (S'85–M'87–SM'06) received the B.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.

From 1993 to 1995, he was with the Computer-Aided Design (CAD) Laboratory, AT&T Bell Laboratories, Allentown, PA. From 1995 to 2000, he was with the Communication Sciences Research Division, Bell Laboratories, Murray Hill, NJ. From 2000 to 2001, he was with CeLight Inc. (an optical networking startup), Silver Spring, MD. Since 2001, he has been with the Electrical and Computer Engineering Department and the Digital Technology Center, University of Minnesota, Minneapolis. His professional interests include design, analysis, and simulation of electronic, electrooptical, and mixed-domain systems, particularly for high-speed and high-frequency communication circuits. Over the years, he has authored or coauthored five best or distinguished papers at Asia and South Pacific Design Automation Conference (ASP-DAC), DAC, and International Conference on Computer Aided Design (ICCAD). He is the holder of ten patents.

Prof. Roychowdhury was cited for Extraordinary Achievement by Bell Laboratories in 1996. He was an IEEE Circuits and Systems (IEEE CAS) Society Distinguished Lecturer in 2003–2005 and served as Program Chair of IEEE's Technical Committee on Computer-Aided Network DEsign (CANDE) and Behavioral Modeling and Simulation Workshop (BMAS) workshops in 2005. Currently, he serves on the Technical Program Committees of DAC, Design, Automation and Test in Europe (DATE), ASP-DAC, The International Symposium on Quality Electronic Design (ISQED), and BMAS, on the Executive Committee of ICCAD, on the Nominations and Appointments Committee of the IEEE Council on Electronic Design Automation (CEDA), and as the Treasurer of CANDE.