CAD for RF circuits

Piet Wambacq (*), Gerd Vandersteen (*), Joel Phillips (**), Jaijeet Roychowdhury (†), Wolfgang Eberle (*), Baolin Yang (‡), David Long (†), Alper Demir (†)

(*) IMEC, Heverlee, Belgium
(**) Cadence Berkeley Laboratories, USA
(†) Bell Laboratories Murray Hill, New Jersey, USA
(‡) Cadence Design Systems, Inc., USA

e-mail: wambacq@imec.be, Gerd.Vandersteen@imec.be, jrp@cadence.com, jaijeet@celight.com

Abstract

Wireless transceivers for digital telecommunications are heterogeneous systems that combine digital hardware, software and analog circuitry. The pressure to miniaturization and lower power consumption for these transceivers imposes tight specifications on their analog RF parts. Many aspects of RF circuits cannot be simulated accurately and efficiently with a classical circuit-level SPICE approach. In this paper three important simulation problems for RF circuits are addressed:

1. *high-level simulation of analog and RF blocks for the determination of the specifications of the circuits*
2. *accurate circuit-level simulation of nonlinear circuits with time constants that differ largely,*
3. *efficient and accurate computation of phase noise in RF oscillators*

For each of these problems, solutions are proposed. These solutions illustrate that accurate and efficient simulations of RF communication circuits need a heterogeneous variety of advanced algorithms.

1. Introduction

The booming market of portable digital telecommunication transceivers requires the design of RF front-ends for these transceivers in a short time. Support from accurate and efficient simulation tools in the design process is essential, but quite challenging. One of the reasons is that a digital telecommunication transceiver unifies various domains: digital hardware, software and analog circuits. For an optimal performance of the complete transceiver, it is important that these different domains are co-designed to some extent. For example, some signal degradations caused by the analog front-end can be compensated in the digital domain. As another example, analog circuits such as power amplifiers and variable-gain amplifiers are often digitally controlled. A careful co-design requires that the different domains can be co-simulated in some way. This allows for example to simulate the bit-error-rate (BER) degradation caused by the different non-idealities in the implementation (e.g. finite wordlength effects in the digital domain, passband ripples and finite stopband suppressions of both analog and digital filters, noise and nonlinear distortion in the analog blocks). Since circuit-level simulations are not feasible to this end, a higher abstraction level is required for the different domains. Whereas this is common practice for the digital domain, it is not easy for the analog and RF blocks. First, accurate high-level models are required for the analog blocks. Next, an efficient simulation engine is required. This simulation engine needs to efficiently co-simulate at a high level RF blocks (operating in the GHz range), analog low-frequency blocks and digital blocks (both operating at baseband frequencies). To this purpose, a digital dataflow simulator for analog circuits has been developed in the simulator FAST [1]. Compared to the classical use of complex lowpass representations [2, 3, 4] for co-simulation of RF and baseband, FAST has a similar efficiency but a higher accuracy. Moreover, the co-simulation problem of the analog domain and the digital domain is greatly simplified since FAST is in essence a dataflow simulator.

The high-level simulations mentioned above can be used to determine the specifications of the individual RF circuits. During the actual design of these circuits, and when extracting parameters for high-level simulations, circuit-level simulations are used to compute transfer functions, impedances, distortion and noise levels, ...For RF circuits this is a particularly challenging simulation problem because high accuracy must be achieved on problems where the signals have vastly different dynamic ranges and timescales. Much effort in the RF circuit simulation field has gone to addressing the multiple timescale problem, but no current simulation technology is capable of achieving high accuracy solutions, reliably and efficiently, in the presence of highly nonlinear devices, realistic semiconductor device models, and malicious waveform profiles. In Section 3 we discuss our experiences in developing a new class of robust high-accuracy RF circuit simulation algorithms, in the process discussing several popular "myths" about the accuracy and numerical stability of some widely-used circuit simulation techniques.

Phase noise is a critical consideration in RF system design, since it corrupts spectral purity and generates large power content in a continuous spread of frequencies around the

1530-1591/01 $10.00 © 2001 IEEE

desired oscillator tone, thus contributing to adjacent channel interference. In digital circuits, the same phenomenon manifests itself as timing jitter. Recently, a rigorous theory has been developed for phase noise [6] that is uniformly applicable to any oscillatory system described by differential-algebraic equations. A key outcome of the theory is that a periodic vector function termed the *Perturbation Projection Vector* or *PPV*, can always be found that represents a "transfer function" between the noise perturbations to the circuit, and the phase noise manifested in the oscillator. In this paper a new technique is proposed to compute the PPV in a way that is reliable and low cost for large circuits regardless of their nature. The technique fully exploits the inherent accuracy of harmonic balance. Further, it is easy to implement in existing steady-state codes, since the same linear Jacobian solution required at each Newton step for finding the steady-state is simply invoked one extra time.

The outline of the paper is as follows. Section 2 discusses high-level simulation of analog and RF front-ends, as well as aspects of the coupling of an analog high-level simulator with a digital dataflow simulator for an analog-RF-digital co-simulation. The approach is illustrated with a high-level simulation of a superheterodyne receiver front-end (see Fig. 1).

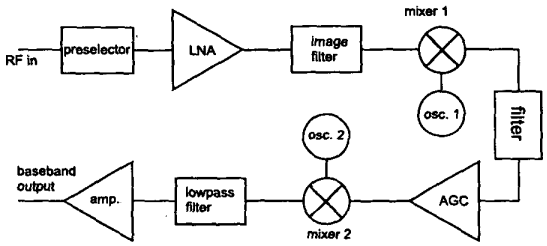

Figure 1: A classical double superheterodyne receiver front end used for high-level simulations.

Section 3 treats a new class of robust high-accuracy RF circuit simulation algorithms. Section 4 discusses efficient phase noise computations starting from a periodic steady-state solution of an oscillator, computed with a harmonic balance or shooting method.

2. High-level simulation of mixed-signal transceivers

For efficient co-simulation of the digital part of a transceiver, e.g. described in dataflow, with the analog front-ends that are modeled at a high level, it is interesting to have one simulation approach. This is realized in the FAST simulator [1], which is in essence a dataflow simulator. In essence, the digital blocks could be simulated with FAST as well. However, we opted for a co-simulation with the digital simulation environment OCAPI [5], since this environment supports different abstraction levels for the digital blocks, as well as a path down to VHDL. Aspects of the coupling between FAST and OCAPI are further discussed in Section 2.2.

The analog front-end blocks can be modeled in various ways. Top-down models are used when a block is not known at all. Usually such models have a low accuracy. An example of a top-down model is a description of an amplifier in the form of a polynomial of order three:

$$i_{out} = g_1 \cdot v_{in} + K_2 v_{in}^2 + K_3 v_{in}^3 + \ldots \qquad (1)$$

Bottom-up models, on the other hand, are extracted from circuit-level simulations or measurements. For the case of a linear transfer function, this can be a list of complex numbers as a function of frequency. For the nonlinear behavior, bottom-up models can also be extracted. For example, in [7] a bottom-up modelling approach is presented that takes into account the frequency dependence of the nonlinear circuit behavior.

Efficient co-simulation of RF blocks with analog baseband blocks is obtained in FAST by using a multicarrier signal representation, which is an extension to multiple carriers of the complex lowpass representation of signals [2] in such a way that the effect of out-of-band distortion is taken into account. Further, a high simulation efficiency is obtained by translating a high-level description into a computational graph (a kind of precompilation step) prior to simulation. The computational graph is constructed in two phases (see Section 2.1). First, the number of linear feedback loops in the architecture is reduced and the overall nonlinearity of the system is limited to order three. In the next phase, equivalent digital filters are computed to represent the s-domain transfer function.

After a construction of the computational graph, the computations are scheduled and executed. The scheduler promotes vector processing as much as possible to speed up the computations. A complete simulation example is discussed in Section 2.3.

2.1 Determining the computational graph

A FAST simulation is the execution of a computational graph prior to execution. This graph is contructed based upon the connectivity in the original architecture. However, feedback loops are best eliminated, since the simulation of a feedback loop cannot be performed with vector processing, since in fact a set of simultaneous equations needs to be solved. This slows down the computations considerably. Therefore, the feedback loops are eliminated. For linear feedback loops this is performed with a standard AC analysis. In addition to the elimination of the feedback loops, internal nodes (which are no longer of any concern) are eliminated as well. Both aspects reduce the simulation time significantly. The simulation results described below

uses such a preprocessing stage in order to eliminate linear feedback loops and the internal nodes.

Next, the s-domain transfer functions are approximated by a digital filter. A multitude of methods are available to extract a digital filter out of a transfer function specified in the frequency domain. The two most frequently used methods in filter design are

- windowing design methods that use a weighting sequence to modify the original impulse response and/or the original transfer function. The correct choice of the weighting sequence is of utmost importance since it influences both the impulse response and the frequency domain characteristic [2, 8].
- Equiripple approximation in the frequency domain using e.g. a Parks and McClellan method [8, 9]. The advantage of this approach is that it is possible provide a frequency dependent weighting of the approximation error.

The latter method is preferred since it enables the user to specify which frequency bands are of importance to its application and which frequency bands do not really matter. This implies that the resulting filter will described the frequency response accurately in the frequency band of interest, while concentrating the modeling errors in the uninteresting frequency ranges.

2.2 High-level analog-digital co-simulation

Typically, the simulation of a complex mixed-signal system is not achieved in a single specific simulation approach. While, specifically for the digital part, flexibility in data models, the order of model refinement steps, and a path towards HDL code generation are important, the analog part requires mainly accurate yet efficient simulations. The program OCAPI [5] serves the above digital requirements. Since this program does not contain the features of FAST that yield a high simulation efficiency for the analog part, and since, on the other hand, FAST does not allow model refinement as in OCAPI, the two approaches are coupled for a co-simulation of analog front-end blocks with digital modem blocks.

Since both OCAPI and FAST are based on the same dataflow scheduling along with generalized firing rules, the coupling interface has to implement queue management and data type adaptation only. To maximize simulation speed and minimize computing platform dependencies, an interface C++ class EKOCAPI with direct access to both its OCAPI and/or FAST I/O queues was preferred over a memory pipe, system pipe, or file-based solution.

Maximum independence between OCAPI and FAST partitions is achieved by slaving all OCAPI schedulers as subprocesses under a FAST master scheduler in a hierarchical way. Both OCAPI and FAST partitions are developed and tested as stand-alone applications before the system integration. This preserves the localities of the dataflow scheduling in every partition leading to a simpler system-level schedule. At instantiation time, the EKOCAPI class defines the connections between OCAPI and FAST partitions. At runtime, i.e. during the simulation, it handles I/O queue management including multi-rate adaptation at the partition boundaries.

The result is a distributed, hierarchical scheduling with lean communication at partition boundaries only, which translates into a low coupling overhead.

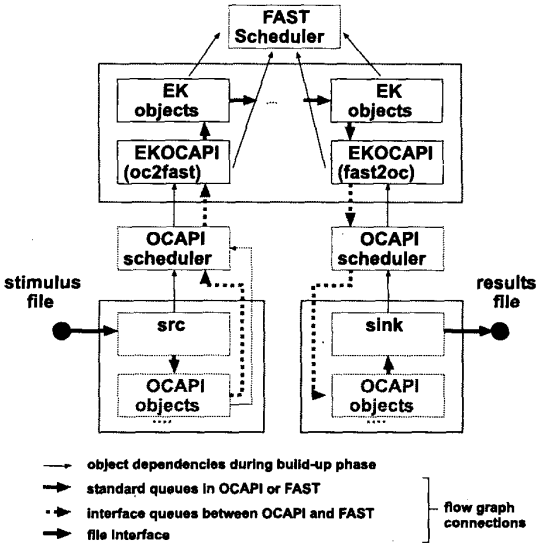

Figure 2: Coupling connects the underlying object and scheduler classes of OCAPI and FAST, respectively, by an interface object, which defines the connection during instantiation and handles the inter-scheduler communication during simulation.

An example for a forward chain topology (Fig. 2) shows how the stimulus is read from a file, preprocessed by an OCAPI partition, handed over to FAST being further processed, then passed to another OCAPI partition for post-processing and the output finally being written to a results file. This scenario is typically found in end-to-end link simulations of a transmitter and receiver over a channel. The coupling mechanism handles also other relevant topologies, including bidirectional or feedback communication between OCAPI and FAST partitions. For this case, initialization methods are foreseen in both tool classes to support cyclo-static dataflow.

2.3 High-level simulation example

A system level simulation has been setup for a classical double heterodyne receiver front end shown in figure 1.

All mixer and amplifier stages are modeled using system level specification for the gain, IIP3, input and output impedance's. The high-level description of the filters are in the form of measured S-parameters. The S-parameter specifications of the filters are used to determine the effect of the impedance mismatches between the different blocks. The equivalent digital FIR filters are extracted afterwards using an equiripple approximation in the frequency band of interest.

The purpose of the overall simulation is to determine the ability of the front-end to suppress the mirror frequency $f_{LO}+f_{IF}$ and two out-of-band spurious components: one at $f_{LO}-f_{IF}/2$ and one close to the wanted input frequency ($f_{LO}-f_{IF}$). Although four modulated tones are considered in practice (the wanted signal, the mirror frequency and the two spurious components), it has been chosen to describe the input signals using only three modulated tones. By increasing the sampling rate of modulation signals, it is possible to join the spurious signal near the RF signal with the RF signal itself. This implies that only three carrier signals (at $f_{LO}+f_{IF}$, $f_{LO}-f_{IF}/2$ and $f_{LO}-f_{IF}$) need to be considered but this at a higher sampling rate. The original RF signal was 16 times oversampled for this purpose.

The cascade of the different nonlinear distortions implies that a large number of tones at distinct frequencies are generated. Some of these tones are exclusively generated by higher-order nonlinear contributions and hence can be neglected. Only the contributions up to the third order are considered in this simulation example. This reduces the computational load significantly: the RF-part only considers $f_{LO}+f_{IF}$, $f_{LO}-f_{IF}/2$ and $f_{LO}-f_{IF}$ due to the preselector filter. The RF mixer is represented using a polynomial static nonlinearity which multiplies the LO signals with a polynomial function of the RF signal. Phase noise of the local oscillators can be included during the simulation. Furthermore, it is assumed that all spectral contributions falling outside the IF frequency band can be neglected. In addition to nonlinear distortion, it is assumed that all active blocks (LNA and mixers) produce additive noise.

After constructing the computational graph, a FAST simulation is setup. A digital symbol is first represented using a sequence of 8 samples which are then oversampled 16 times in order to represent both the wanted signal and one of spurious responses using one and the same carrier. This implies that a single digital symbol is represented by 128 time domain samples.

The overall simulation time of FAST for receiving a single digital symbol takes about 1.5ms on a PIII 500MHz. This means that the influence of additive noise, nonlinear distortion, impedance mismatches, phase noise of the oscillators, interference due to spurious components on e.g. the BER can be evaluated efficiently within a reasonable time.

3. Achieving efficient high-accuracy RF circuit simulation

The two numerical methods commonly used in RF circuit simulation are the shooting-Newton method, based on low-order finite difference discretizations such as the second-order Gear method, and the harmonic balance method, based on high-order spectral discretizations [10, 11]. The advantage of the low-order polynomial-based methods traditionally used in SPICE-class circuit simulators is that they can select time-points based on localized error estimates and as a result can easily handle sharp transitions in circuit waveforms. The harmonic balance method, on the other hand, has the advantage of attaining spectral accuracy for smooth waveforms. Neither method, however, is capable of achieving high accuracy solutions, reliably and efficiently, under all circumstances likely to occur when simulating a modern RFIC.

Because high precision computations are often necessary in simulation of sensitive RF circuits, we have recently developed a method that works by discretizing the circuit equations by dividing the simulation domain into a set of intervals whose size is adaptively chosen and using Chebyshev polynomials to represent the solution in each interval [12]. On each interval, the time-derivative operator is replaced by a Chebyshev differentiation matrix, and continuity between the interval boundaries is enforced. The order of approximation within each interval and the size of the intervals can be adjusted independently, resulting in a flexible method that can achieve high resolution on a wide variety of nonlinear circuits. It is also suitable for combining with matrix-implicit Krylov-subspace solvers in order to analyze large circuits with moderate computational cost.

Another way of viewing our discretization scheme is as an implicit Runge-Kutta (IRK) method [13]. Leveraging the theoretical framework available for analysis of IRK methods has given us an increased understanding of limitations of numerical techniques traditionally used in analog and RF circuit simulation, in particular the computational advantages associated with the superior stability properties that multistage IRK methods possess. This is important because in practice the multistep BDF formulas are not actually as numerically stable as is popularly believed. The Gear methods of orders 1 and 2 are considered to be "A-stable", that is, generating numerically stable solutions for all systems with eigenvalues λ in the closed left half-plane, regardless of the numerical step size h. In Figure 3 we show the stability region of the third-order Gear method, in the complex plane $z = \lambda h$. For a small region around the imaginary axis, the method is not stable. This region becomes considerably larger for higher order Gear methods, such that methods of order greater than three are not used in practice. We also show the stability region of a fourth-order scheme from a particular family of Chebyshev-based IRK methods. This particular family was deliberately con-

structed to be stiffly stable, but we did not design it to be A-stable, though such a construction is possible. We chose a fourth order scheme because it is one of the less stable in the family; the methods are A-stable for orders one and two and, interestingly, become much more stable at higher orders. To make a fair comparison we must adjust the regions to account for the fact that a single IRK step has several internal stages, each comparable to a single Gear steps. The method is not A-stable, with the size of the region of instability comparable to that of Gear-3. This is generally acceptable for circuit simulation.

To understand the advantages of the IRK methods, we must engage in a more general analysis. Any practical simulator uses non-uniform timesteps, and non-uniform timesteps generally degrade the stability behavior of multistep methods. In particular, the second order Gear method is *not* A-stable for nonuniform steps, and in fact it is not stable for *any* timestep if the ratios between consecutive steps exceed about 2.4. Precise statements about numerical stability are difficult to make for nonuniform timestep schemes, but we may formally define a "stability region" for a multistep formula with a fixed stepsize growth ratio to be the region where no artificial numerical growth is introduced, just as for the fixed timestep case. In Figure 3 we see that, by this metric, a second-order Gear scheme where the timestep ratios are bounded by 2.0 is considerably less stable than third order Gear. The stability properties of the third-order Gear method are even more strongly affected by varying timesteps. The practical implication is that a rapid change of timestep in a multistep code also necessarily comes with a loss of order, because higher-order schemes would be destabilized by rapid timestep variation.

There are two consequences to dropping method order. First, to achieve the same accuracy, in general the lower order methods must take smaller timesteps. Therefore an efficiency penalty is introduced. Second, even though in theory the lower order Gear methods can always achieve acceptable accuracy by using a sufficiently small timestep, beyond a certain level of precision roundoff effects start to come into play and additional accuracy cannot be obtained by decreasing the timestep in a low-order method (particularly for the first-order backward-Euler scheme). In fact, if the timesteps are made exceedingly small, the matrices associated with the boundary problems generated by an RF simulator may become ill-conditioned, slowing the convergence of the iterative Krylov-subspace solvers, and in extreme cases can even make the results *less* accurate than if a larger timestep was used. In contrast, the Chebyshev-IRK method is a one-step method. Its high order of accuracy comes from possessing multiple internal stages. Each step is independent of the ones before and after, so the method easily adapts to very rapid variation of the solution waveform without risk of numerical instability.

It is also interesting to note something that cannot be seen from the stability region plot : the worst-case growth rate of the potentially unstable mode is much less for the IRK family than for the multistep methods. In all aspects, the IRK methods with multiple implicit stages are naturally more stable than the multistep methods.

The spectral discretization methods (representation of the solution by Fourier series, as in harmonic balance) turn out to have related shortcomings. Accuracy in the harmonic balance method is usually achieved by increased the order of approximation. For a given set of collocation points, the maximal possible order of approximation is used. With smooth waveforms, this has the very desirable effect of achieving spectral convergence, meaning that as the spacing between the collocation points decreases, then asymptotically the error decreases faster than any polynomial approximation (in particular, faster than the Gear methods).

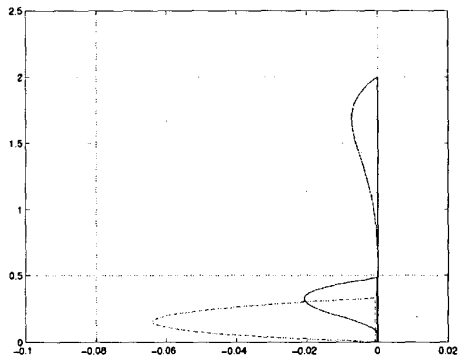

Fig. 3: Stability regions for third-order Gear (solid line), second-order Gear with timestep ratio bounded by 2.0 (dashed line), and fourth-order Chebyshev-RK method (solid area).

However, representation of the solution by Fourier series means that the basis functions, sines and cosines, must cover the entire domain. Unlike the Gear methods, which couple only local points, every point in the solution interval is coupled to every other through each basis function. The spectral convergence of the Fourier series is very desirable for smooth waveforms and nearly linear circuits, but unfortunately, a single discontinuous point can destroy the spectral convergence properties. Because of the global nature of the basis functions, spectral convergence is very difficult to recover once interrupted. Most semiconductor device models possess some degree of discontinuity in the derivatives, and so true spectral convergence is rarely observed in practice. In the worst case, near a point where a waveform (and we emphasize that the culprit waveform may be a solution variable, such as a node voltage, or a

residual quantity, such as the current inflow at a node) or its derivative is almost discontinuous, for example a breakpoint of a piecewise-linear model or source, then oscillations can occur that seriously degrade the accuracy of the method. Gear methods avoid this problem by dropping to a low-order method at the aberrant point, with costs already discussed, but remedies are more difficult for the higher order schemes. The result is that harmonic balance codes can mysteriously lose accuracy on certain types of nonlinear problems. The multi-interval Chebyshev scheme avoids these problems by locally adapting the interval size.

Many of the numerical advantages of highly implicit, multistage numerical integration schemes have been known for some time. They have not been widely used, however, because the traditional view of the circuit simulation community has been that the highly coupled nature of the linear and nonlinear systems makes the IRK methods, on a per-timestep basis, disproportionately expensive as order (i.e. accuracy) is increased. However, in RF simulation, the spread of iterative solution methods has ameliorated the computational problems associated with implementing implicit multi-stage integration schemes. With the Chebyshev collocation method, each internal stage where the solution is computed is equivalent to a single timepoint in a shooting or harmonic balance code, requiring precisely one matrix assembly. As the overall number of timepoints can be much less due to the more adaptable discretization, the issue for these methods is the construction of an effective preconditioner in order to achieve minimal computational complexity.

4. Phase noise eigenfunctions from harmonic balance or shooting

The output power spectrum of an oscillator shows a peak at the oscillation frequency and tails in both sidebands, which decrease as the frequency offset from the oscillation frequency increases. These tails are referred to as phase noise. The mechanisms that determine this phase noise have long been a point of discussion. Many analysis theories have been developed recently [14-18] as a correction to the classical paper of Leeson [19]. Depending on the analysis method, the simulation of the phase noise is also different. In [6] a rigorous theory has been developed for phase noise that is uniformly applicable to any oscillatory system described by differential-algebraic equations. A key outcome of the theory is that a periodic vector function $v_1(t)$, termed the *Perturbation Projection Vector* or *PPV*, can always be found that represents a "transfer function" between the noise perturbations to the circuit, and the phase noise manifested in the oscillator.

Existing numerical techniques for finding the PPV are based on explicit eigendecomposition of the monodromy matrix of the adjoint of the oscillator's differential equations. Full eigendecompositions are expensive for large systems; however, iterative linear methods can be used to find only a few eigenpairs with relatively little computation, to the point where the cost for finding the PPV becomes insignificant compared to, e.g., that for finding a steady-state solution of the oscillator (a prerequisite).

For high-Q oscillators, however, monodromy matrices often have many eigenvalues close to 1 that are numerically indistinguishable from the oscillatory mode. In such situations, explicit eigendecomposition methods need to find a potentially large number of candidate PPVs and choose one from amongst them using heuristics. Finding many candidate eigenpairs can raise the computation to the point where it becomes dominant. Using heuristics to find the correct PPV is also often unreliable in such cases. Furthermore, monodromy matrix calculations have so far been limited to the time domain, where numerical accuracy is inherently poor compared to carefully implemented frequency-domain techniques.

We briefly describe here[1] a new computational procedure for the PPV that does not require the monodromy matrix. Instead, the method uses only a *single* linear solution of the steady-state Jacobian matrix of the oscillator. Heuristics are not required - the linear solution directly produces the correct PPV, with an accuracy limited only by the intrinsic numerical conditioning of the steady-state equations. Furthermore, the Jacobian matrix can be either a frequency-domain one (e.g., from harmonic balance) or a time-domain one (e.g., from shooting), as appropriate for the circuit in question. Hence the new technique is a) reliable and low cost for large circuits regardless of their nature, b) able to fully exploit the inherent accuracy of harmonic balance, and c) easy to implement in existing steady-state codes, since the same linear Jacobian solution required at each Newton step for finding the steady-state is simply invoked one extra time.

The new technique is based on an elegant connection between the Fourier coefficients of the PPV and the augmented Jacobian matrix of the oscillator's steady-state equations. For harmonic balance, this connection is:

$$\bar{\mathbf{J}}^{HB^*}\,[\bar{l}^T, d]^T = [..., 0, ..., 0, ..., 1]^T. \qquad (2)$$

where $\bar{l} = \bar{\mathbf{V}}^{FD}_{v_1(t)}$ are the Fourier coefficients of the PPV $v_1(t)$. The augmented harmonic balance matrix $\bar{\mathbf{J}}^{HB^*}$ arises naturally as the Jacobian matrix of the oscillator's steady-state equations augmented by a phase condition, with the frequency of oscillation as an additional unknown. Hence, from equation (2), *the Fourier coefficients of $v_1(t)$ can be obtained from a single solution of the Hermitian of the augmented harmonic balance Jacobian of the oscillator,*

[1] Details are available in [20].

with right-hand-side equal to a unit vector with value 1 in the phase condition equation. By exploiting circulant approximations to $\tilde{J}^{HB}$ and applying iterative linear methods to solve equation (2), this computation becomes approximately linear in the system size.

To evaluate the new method, it has been compared against the established method that uses monodromy matrix eigendecomposition. The steady-state of a tank-circuit-based oscillator has been computed using harmonic balance with $m=31$ harmonics, resulting in $N=63$ distinct frequency components. The frequency of oscillation f_0 was 159154.853364298Hz. The time-domain voltage waveform at the tank capacitor and the current through the power supply are shown in Figures 4 and 5,.respectively.

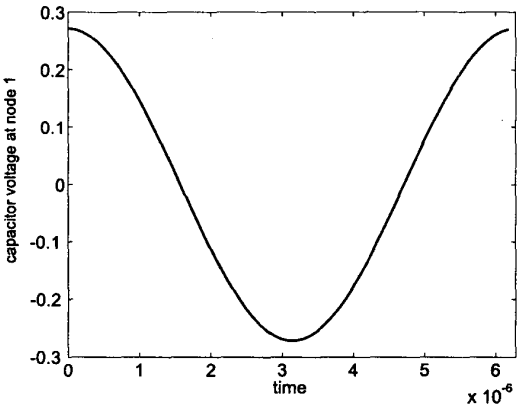

Figure 4: Oscillator steady-state: voltage over the capacitor of the tank.

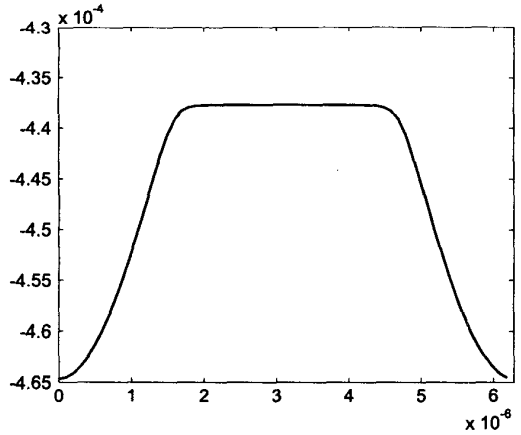

Figure 5: Oscillator steady-state: current through VDD.

The PPV $v_1(t)$ is first determined through the time-domain monodromy matrix by computing its 1-eigenpair using iterative linear methods followed by manual selection from among candidate eigenpairs. The eigenvector thus obtained was then used as an initial condition for a transient simulation of the adjoint system, using a time-step corresponding to an oversampling factor of 4 (i.e., 4N timepoints) to limit accuracy loss from linear multistep formulae for DAE solution. The result of this transient simulation, after normalization, is the conventionally computed PPV, referred to as $v_{1m}(t)$.

The new method described above simply computes the system of equation (2) directly from the oscillator's harmonic balance Jacobian, with a single iterative linear solve. No oversampling is used by the method. The PPV obtained in this manner is denoted by $v_{1d}(t)$.

Figures 6 and 7 depict the components of $v_{1d}(t)$ (solid red line) and $v_{1m}(t)$ (blue **x** marks) corresponding to the capacitor node and the power supply current, respectively.

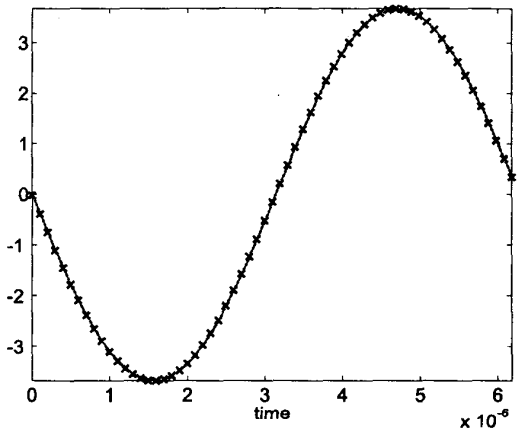

Figure 6: Capacitor node of PPVs v_{1d} and v_{1m}.

It can be seen that the PPV waveforms produced by the two methods are visually indistinguishable from each other.

A more critical assessment of the two methods can be made using the fact that the dot product of the PPV $v_1(t)$ with the tangent vector $u_1(t)$ of the steady-state orbit of the oscillator is always unity. We plot the error $\varepsilon_d(t)=\left|u_1^T v_{1d}(t)-1\right|$ versus $\varepsilon_m(t)=\left|u_1^T v_{1m}(t)-1\right|$ in Figure 8. The solid red line indicates $\varepsilon_d(t)$, the error of the new method, while the blue marks indicate $\varepsilon_m(t)$. The new method is about two orders of magnitude more accurate than monodromy matrix eigendecomposition, despite the 4x oversampling used by the latter method.

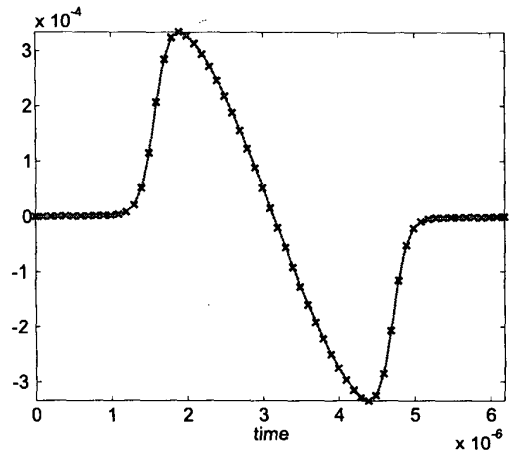

Figure 7: Power supply current component of PPVs v_{1d} and v_{1m}.

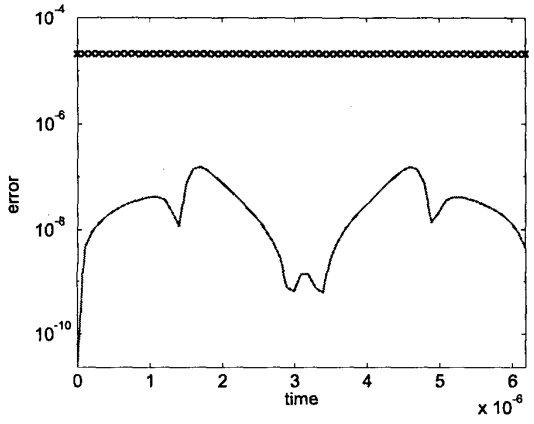

Figure 8: Errors in the PPV obtained using the monodromy and new methods.

References

[1] G. Vandersteen *et al.*, "A methodology for efficient high-level dataflow simulation of mixed-signal front-ends of digital telecom transceivers", *Proc. 37th Design Automation Conference*, 2000.

[2] Jeruchim M.C., Ph. Balaban, K.S.Shanmugan, "Simulation of Communication Systems", Plenium Press, New York and London, 1992.

[3] SPW, Cadence Design Systems, Inc., http://www.cadence.com/eda_solutions

[4] COSSAP, Synopsys, Inc., http://www.synopsys.com/products/dsp

[5] P. Schaumont et al., "A programming environment for the design of complex high speed ASICs", *Proc. DAC*, pp. 315-320, 1998.

[6] A. Demir, A. Mehrotra, and J. Roychowdhury. Phase Noise in Oscillators: A Unifying Theory and Numerical Methods for Characterization, *IEEE Trans. Circuits and Systems -I*, May 2000.

[7] P. Wambacq, Petr Dobrovolný, Stéphane Donnay, Marc Engels, Ivo Bolsens, "Compact modeling of nonlinear distortion in analog communication circuits", *Proc. DATE 2000.*

[8] Oppenheim A.V. and R.W.Schafer, "Digital Signal Processing", Englewood Cliffs, New Jersey, Prentice-Hall, 1975.

[9] Parks T.W. and J.H. McClellan, Chebyshev approximation of nonrecursive digital filters with linear phase, *IEEE Trans Circuit Theory*, CT-19, March 1972, pp.189-194.

[10] K. S. Kundert, "Introduction to RF simulation", *IEEE J. Solid-State Circuits*, September, 1999.

[11] R. Gilmore and M. Steer, Nonlinear circuit analysis using the method of harmonic balance : a review of the art. Part I: Introductory concepts. *Int. J. Microwave and Millimeter Wave Computer Aided Engineering*, 1991.

[12] B. Yang and J. Phillips, "A multi-interval Chebyshev collocation method for efficient high-accuracy RF circuit simulaiton", *Proc. DAC*, June 2000.

[13] E. Hairer and G. Wanner, "Solving ordinary differential equations II", Springer-Verlag, 1991.

[14] F. Kaertner, "Analysis of white and f^[-\alpha] noise in oscillators," *Int. J. Circuits Theory Appl.*, Vol. 18, pp. 445-519, 1990.

[15] K. A. Kouznetsov and R. G. Meyer, "Phase noise in LC oscillators," *IEEE J. Solid-State Circuits*, Vol. 35, pp. 1244-1248, Aug 2000.

[16] Q. Huang, "Phase noise to carrier ratio in LC oscillators," *IEEE Trans. Circuits and Syst.-I*, Vol 47, pp. 965-980, July 2000.

[17] C. Samori *et al.*, "Spectrum folding and phase noise in LC tuned oscillators," *IEEE Trans. Circuits and Systems – II*, Vol. 45, No. 7, pp. 781-790, July 1998.

[18] A. Hajimiri and T. Lee, "A general theory of phase noise in electrical oscillators", *IEEE J. Solid-State Circuits*, Feb. 1998.

[19] D.B. Leeson, "A simple model of feedback oscillator noise spectrum", *Proc. IEEE*, Vol. 54, pp. 329-330, February 1966.

[20] A. Demir, D. Long and J. Roychowdhury. Computing Phase Noise Eigenfunctions Directly from Steady-State Jacobian Matrices, *Proc. ICCAD*, November 2000.

Acknowledgements

The authors G. Vandersteen, P. Wambacq and W. Eberle wish to thank the European ESPRIT program (project SALOMON) for their support. The authors A. Demir, D. Long and J. Roychowdhury acknowledge Kiran Gullapalli of Motorola for providing the oscillator circuit, its steady state solution, and v1(t) computed using monodromy matrix methods.