Efficient computation of quasi-periodic circuit operating conditions via a mixed frequency/time approach

Dan Feng*, Joel Phillips*, Keith Nabors*, Ken Kundert*, Jacob White†
*Cadence Design Systems, San Jose, CA 95134
†Massachusetts Institute of Technology, Cambridge, MA 02139

Abstract

Design of communications circuits often requires computing steady-state responses to multiple periodic inputs of differing frequencies. Mixed frequency-time (MFT) approaches are orders of magnitude more efficient than transient circuit simulation, and perform better on highly nonlinear problems than traditional algorithms such as harmonic balance. We present algorithms for solving the huge nonlinear equation systems the MFT approach generates from practical circuits.

1 Introduction

One of the most difficult challenges in circuit simulation is the analysis of circuits that operate on multiple timescales. Typical examples of this class are switched-capacitor filters and circuits used in RF communications systems. Applying standard transient analysis to such circuits can imply simulating the detailed response of the circuit over hundreds of thousands of clock cycles (millions of timepoints), a generally impractical approach. Fortunately many circuits of interest are designed to operate near a time-varying, but quasi-periodic, operating point. Some of these circuits can be analyzed by assuming one of the circuit inputs produces a periodic response that can be directly calculated by steady-state methods[4], thus avoiding long transient simulation times. Any other (time-varying) circuit inputs are treated as small-signal by linearizing the circuit around the periodic operating point. Efficient algorithms[12, 7] now exist to find periodic operating points and to perform periodic time-varying small-signal analysis[13, 8].

However, many circuits cannot be analyzed with the periodic-operating-point-plus-small-signal approach. For example, predicting intermodulation distortion of a narrowband circuit, such as a mixer+filter, involves calculating the nonlinear response of the mixer circuit, driven by an LO, to two high-frequency inputs that are closely spaced in frequency. The steady-state response of such a circuit is quasi-periodic.

The situation is further complicated by the fact that many multi-timescale circuits have a response (again mixers and switched-capacitor filters are good examples) that is highly nonlinear with respect to at least one of the exciting inputs, and so steady-state approaches such as multi-frequency harmonic balance[7] do not perform well.

To circumvent these difficulties, mixed frequency-time approaches(MFT)[2, 3] have been proposed. The methods in [2, 3] exploit the fact that many circuits of engineering interest have a strongly nonlinear response to only one input, such as the clock in the case of a switched-capacitor circuit, or local oscillator in the case of a mixer, but respond only in a weakly nonlinear manner to other inputs. Recent extensions[9] of the MFT algorithms can sometimes treat circuits with strongly nonlinear responses to multiple inputs.

The original MFT algorithms suffered from several drawbacks that prevented their application to practical circuits, particularly circuits of large size. This paper makes four improvements on the original algorithms that enable the MFT method to be applied to large circuits. In section 3 we discuss a quasi-periodic boundary condition that avoids the ill-conditioning due to a poor choice of boundary condition in previous algorithms [3, 2]. In Section 4, we show how matrix-free Krylov-subspace based iterative schemes may be extended to the MFT methods to allow solution of very large scale problems. A similar approach is taken in [15]. Section 6 presents a simple continuation scheme that improves convergence of the Newton iteration. In section 5 we show that, while the unpreconditioned matrices arising from applying shooting methods are often well conditioned and thus suitable for direct application of Krylov-subspace based iterative solvers, the matrices generated by the MFT method are never well conditioned even for a well-chosen boundary condition. We present a preconditioning strategy that gives rapid convergence of the iterative solver.

2 Background on the MFT approach

Circuit behavior is usually described by a set of nonlinear differential-algebraic equations (DAEs) that can be written as

$$\frac{d}{dt}Q(v(t)) + I(v(t)) + u(t) = 0, \tag{1}$$

where $Q(v(t)) \in \Re^N$ is typically the vector of sums of capacitor charges at each node, $I(v(t)) \in \Re^N$ is the vector of sums of resistive currents at each node, $u(t) \in \Re^N$ is the vector of inputs, $v(t) \in \Re^N$ is the vector of node voltages, and N is the number of circuit nodes.

We are interested in the case in which the input signal $u(t)$ is quasiperiodic. We will say a signal is $L-$quasiperiodic if it can be written as a Fourier series with L fundamental frequencies. RF circuits are generally influenced by one periodic timing signal, often referred to as the LO or the clock, and one or more information signals. If we let f_c denote the clock frequency, and $f_1 \ldots f_S$ the S information signals, then the $(S+1)$-quasiperiodic input can be written as

DAC 99, New Orleans, Louisiana

$$u(t) = \sum_{k_1=-\infty}^{\infty} \cdots \sum_{k_S=-\infty}^{\infty} \sum_{k_c=-\infty}^{\infty} U(k_1, \cdots, k_S, k_c) \quad (2)$$
$$\times\ e^{-j2\pi k_1 f_1 t} \cdots e^{-j2\pi k_S f_S t} e^{j2\pi k_c f_c t}$$

The fundamental assumption of the MFT method is that the circuit possesses a quasiperiodic steady-state response. That is, $v(t)$ is an $S+1$ quasiperiodic signal with fundamentals $f_1, \ldots, f_S, f_c$. Furthermore, since all physical circuits have a finite bandwidth, $v(t)$ can be well approximated by taking only a finite number of terms in the Fourier series, so that

$$v(t) = \sum_{k_1=-K_1}^{K_1} \cdots \sum_{k_S=-K_S}^{K_S} \sum_{k_c=\infty}^{\infty} V(k_1, \cdots, k_S, k_c)$$
$$\times\ e^{-j2\pi k_1 f_1 t} \cdots e^{-j2\pi k_S f_S t} e^{-j2\pi k_c f_c t}. \quad (3)$$

where $V(k_1, \cdots, k_S, k_c) \in C^N$. An interesting property of the MFT method is that it is not necessary to truncate to a finite number of harmonics of f_c.

Now suppose that $v(t)$ is sampled at a discrete set of points $t'_n = t_0 + nT_c$, where $T_c = 1/f_c$ is the clock period, $t_0 \in [0, T_c)$ and n runs over the integers, to obtain a discrete signal $\bar{v}(t')$. Since

$$\bar{v}(t'_n) = \sum_{k_1=-K_1}^{K_1} \cdots \sum_{k_S=-K_S}^{K_S} \bar{V}(k_1, \cdots, k_S)$$
$$\times\ e^{-j2\pi k_1 f_1 t'_n} \cdots e^{-j2\pi k_S f_S t'_n} \quad (4)$$

where

$$\bar{V}(k_1, \cdots, k_S) = \sum_{k_c=-\infty}^{\infty} V(k_1, \cdots, k_S, k_c) e^{-j2\pi k_c f_c t_0}, \quad (5)$$

the "envelope" $\bar{v}(t')$ is S-quasiperiodic and can be represented as a Fourier series in only the "information" fundamentals. The clock fundamental has disappeared. For an example of such an envelope, see Figure 1. The continuous waveform is the waveform that has $\bar{V}$ as its Fourier coefficients, or, equivalently, obtained by Fourier interpolation of the sampled points.

In principle, since there are only $K = \prod_{s=1}^{S}(2K_s+1)$ Fourier coefficients to represent $\bar{v}$, then once the value of K distinct points $t_1, \cdots, t_K$ along the sample envelope are known, then the full envelope can be recovered. The envelope corresponding to the quasiperiodic operating point is obtained by obtaining K sample points that lie on the solution to the DAE given by (1).

We define the state transition function $\phi(v_0, t_k, t_f) = v(t_f)$: $v(t)$ satisfies equation (1) for $t \in [t_k, t_f]$ and $v(t_k) = v_0$. In particular, define the vector

$$\bar{v}_0 = [\bar{v}^T(t_1), \cdots, \bar{v}^T(t_K)]^T = [v^T(t_1), \cdots, v^T(t_K)]^T, \quad (6)$$

where superscript T denotes matrix transpose, to contain $\bar{v}$ at the K sample points $t_k = t_0 + n_k T_c, k = 1 \ldots K, n_k \in Z$. The value of the K points that follow by one cycle can be obtained from the transition function,

$$\bar{v}^T_{T_c} = [v^T(t_1 + T_c), \cdots, v^T(t_K + T_c)]^T \quad (7)$$
$$= [\phi(v(t_1), t_1, t_1 + T_c)^T, \cdots, \phi(v(t_K), t_K, t_K + T_c)^T]^T$$

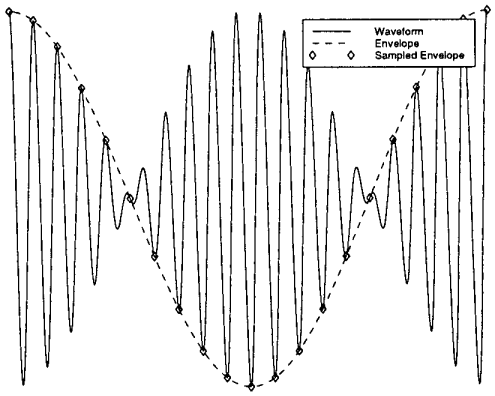

Figure 1: Sample envelope is the waveform traced out when signal is sampled with the clock period

which may be written more compactly by introducing the multi-cycle transition function that is the collection of the K transition functions from t_k to $t_k + T_c$, as

$$\bar{v}_{T_c} = \Phi_{T_c}(\bar{v}_0). \quad (8)$$

Now note that for each node n, the vector of signals on that node, at the sample time plus one clock cycle, $\bar{v}^n_T$, is a delayed version of the signals at the sample points (this will be made more clear below). Therefore there exists a linear operator D_{T_c} that maps $\bar{v}^n_0$ to $\bar{v}^n_{T_c}$

$$\bar{v}^n_{T_c} = D_{T_c} \bar{v}^n_0. \quad (9)$$

Note that D_{T_c} is a real matrix and independent of node n. Hence (9) holds for each $n = 1, \cdots, N$. It represents a boundary condition on solution to (1).

Combining (9) and (8) gives

$$(D_{T_c} \otimes I_N)\bar{v}_0 - \Phi_{T_c}(\bar{v}_0) = 0, \quad (10)$$

where $\otimes$ is the Kronecker product (see [5] for definition) and I_N is the N by N identity matrix. Equation 10 is a system of KN nonlinear equations and KN unknowns $\bar{v}_0$ that can be solved for the envelope sample points. From these sample points and the transition functions the circuit's quasiperiodic operating point (in particular, the spectrum of v) can be recovered, which is discussed in Section 7.

3 Sample point selection

To construct the matrix D_{T_c}, referred to as the delay matrix, consider the Fourier series of $\bar{v}_0$ and $\bar{v}_{T_c}$. Referring to equation (4), we have

$$\bar{v}(t'_n + T_c) = \sum_{k_1=-K_1}^{K_1} \cdots \sum_{k_S=-K_S}^{K_S} \bar{V}(k_1, \cdots, k_S)$$
$$\times\ e^{-j2\pi k_1 f_1 t'_n} \cdots e^{-j2\pi k_S f_S t'_n} \Omega_{T_c}(k_1, \cdots, k_S) \quad (11)$$

where

$$\Omega_{T_c}(k_1, \cdots, k_S) = e^{-j2\pi k_1 f_1 T_c} \cdots e^{-j2\pi k_S f_S T_c} \quad (12)$$

Thus if Γ is the matrix mapping sample points on the envelope to Fourier coefficients, then the delay matrix may be constructed as

$$D_{T_c} = \Gamma^{-1}\Omega_{T_c}\Gamma. \tag{13}$$

In particular Γ may be constructed as the Kronecker product of one-dimensional $2K_s + 1$-point Fourier-transform matrices

$$\Gamma^{(s)}_{mn} = e^{j2\pi m f_s t_n/(2K_s+1)} \tag{14}$$

as

$$\Gamma = \Gamma^{(1)} \otimes \cdots \otimes \Gamma^{(S)} \tag{15}$$

From the properties of Kronecker products, Γ^{-1} is likewise a Kronecker product of the inverses of the $\Gamma^{(s)}$. In [3], no particular care was taken in the choice of the sample points t_k, so that the $\Gamma^{(s)}$'s were ill-conditioned matrices corresponding to an "almost-periodic" Fourier transform. Fortunately, a better choice of points is possible.

Assume the K sample points can be arranged into an S-dimensional array $\tau(k_1, \cdots, k_S)$, $-K_s \le k_s \le K_s$, $1 \le s \le S$, such that for a given dimension s, there exists an integer p and

$$\tau(\cdots, k_s + 1, \cdots) - \tau(\cdots, k_s, \cdots) = \frac{T_s}{2K_s + 1} + pT_s \tag{16}$$

hold. In this case the entries of the $\Gamma^{(s)}$ matrices are

$$\Gamma^{(s)}_{mn} = e^{j2\pi mn/(2K_s+1)} \tag{17}$$

That is, they are the DFT matrices, and the matrix $\Gamma : C^{2K_1+1} \times \cdots \times C^{2K_S+1} \Longrightarrow C^{2K_1+1} \times \cdots \times C^{2K_S+1}$ represents an S-dimensional DFT. Thus Γ has a condition number of one; it is perfectly well-conditioned.

4 Matrix-implicit solution procedure

We employ Newton's method to solve

$$(D_{T_c} \otimes I_N)\bar{v}_0 - \Phi_{T_c}(\bar{v}_0) = 0, \tag{18}$$

At iteration i, the Jacobian matrix is given by

$$(D_{T_c} \otimes I_N) - \left.\frac{\partial \Phi_{T_c}}{\partial \bar{v}_0}\right|_{\bar{v}_0 = \bar{v}_0^i}, \tag{19}$$

Recall from (13) $D_{T_c} = \Gamma^{-1}\Omega_{T_c}\Gamma$, which is fixed through all Newton iterations. Let $J = \left.\frac{\partial \Phi}{\partial \bar{v}_0}\right|_{\bar{v}_0 = \bar{v}_0^i}$ be obtained from the multicycle transition function by

$$\left.\frac{\partial \Phi_{T_c}}{\partial \bar{v}_0}\right|_{\bar{v}_0^i} = \begin{bmatrix} \left.\frac{\partial \phi_1}{\partial \bar{v}_0(t_1)}\right|_{\bar{v}_0(t_1)^i} & & \\ & \ddots & \\ & & \left.\frac{\partial \phi_K}{\partial \bar{v}_0(t_K)}\right|_{\bar{v}_0(t_K)^i} \end{bmatrix}. \tag{20}$$

Note that J is block-diagonal. Defining $b = -(D_{T_c} \otimes I_N)\bar{v}_0^i - \Phi(\bar{v}_0^i)$, we perform the Newton iteration by solving the equation

$$((D_{T_c} \otimes I_N) - J)\Delta\bar{v}_0^i = b \tag{21}$$

using the iterative solver GMRES [11], and setting

$$\bar{v}_0^{i+1} = \bar{v}_0^i + \Delta\bar{v}_0^i. \tag{22}$$

Each iteration of GMRES requires a matrix-vector multiplication. For a vector $q \in \Re^{KN}$, the term $(D_{T_c} \otimes I_N)q$ is calculated by first applying a K dimensional DFT N times, then scaling each row with Ω_{T_c}, and finally applying a K dimensional inverse DFT N times.

Let q be partitioned into $q = [q_1^T, \cdots, q_K^T]^T$, $q_k \in \Re^N$, for $1 \le k \le K$. Then

$$\frac{\partial \Phi}{\partial \bar{v}} q = \begin{bmatrix} \frac{\partial \phi_1}{\partial \bar{v}_0(t_1)} q_1 \\ \vdots \\ \frac{\partial \phi_K}{\partial \bar{v}_0(t_K)} q_K \end{bmatrix}. \tag{23}$$

The calculation of each $\frac{\partial \phi_k}{\partial \bar{v}_0(t_k)} q_k$ can be carried out in the same matrix-implicit fashion as discussed in [12].

5 Preconditioning

For many problems, the GMRES method is not efficient for solving (21) without an effective preconditioner. A simple analysis reveals why. Consider the case where the state transition function of the circuit, over one clock cycle, is approximately linear, that is $\phi(x, t, t + T_c) \simeq Hx(t)$. Linear circuits are a trivial example of a case where this is true, and while nonlinear circuits will have nonlinear state-transition functions, if the method performs poorly for linear circuits, it surely will not work well for nonlinear circuits either. However, many nonlinear circuits have a state-transition function that is nearly linear[4], a fact we will exploit below to construct an effective preconditioner. The convergence of the GMRES method will depend on the location of the eigenvalues of the Jacobian matrix, $D_{T_c} - J$. If λ_H is an eigenvalue of the matrix H, then $e^{i\omega T_c} - \lambda_H$, where $\omega = 2\pi(k_1 f_1 + k_2 f_2 + \cdots k_S f_S)$ will be an eigenvalue of $D_{T_c} - J$, for every $k_1, k_2, \ldots$ in the MFT analysis. Thus unless all the secondary input frequencies are nearly commensurate with the clock frequency, the eigenvalues of $D_{T_c} - J$ will be "fanned out" by delay matrix. This will cause severe convergence problems for the GMRES solver. Roughly speaking, the GMRES algorithm in the MFT algorithm with K total harmonics will take K times as many iterations to converge than the GMRES iteration for the steady-state problem with only the clock excitation applied.[1]

The following lemmas[5] about the properties of Kronecker products are needed to perform the formal analysis.

Lemma 5.1 *If* $A_1, A_2, \cdots, A_p \in \mathcal{F}^{m\times m}$, $B_1, B_2, \cdots, B_p \in \mathcal{F}^{n\times n}$ *then*

$$(A_1 A_2 \cdots A_p) \otimes (B_1 B_2 \cdots B_p) = (A_1 \otimes B_1)(A_2 \otimes B_2) \cdots (A_p \otimes B_p).$$

Lemma 5.2 *If* $A \in \mathcal{F}^{m\times m}$, $B \in \mathcal{F}^{n\times n}$ *then*

(a) $(A \otimes I_n)(I_m \otimes B) = (I_m \otimes B)(A \otimes I_n)$.

(b) $(A \otimes B)^{-1} = A^{-1} \otimes B^{-1}$.

Theorem 5.3 *If the system in (1) is linear time-invariant then if* λ_H *is an eigenvalue of* $\partial\Phi/\partial\bar{v}_0$, *then* $e^{j2\pi\omega_k T_c} - \lambda_H$ *is an eigenvalue of the MFT Jacobian matrix.*

[1] This follows because the eigenvalues of H are typically inside the unit circle of the complex plane. The delay matrix replicates the eigenvalue structure K times, each shift being a complex number of order unity, and generally causing the convex hull of the eigenvalues of $D_{T_c} - J$ to enclose the origin.

Proof. For linear time-invariant circuits, the diagonal blocks of $\frac{\partial \Phi}{\partial \bar{v}_0}$ are the same, i.e., $\frac{\partial \phi_1}{\partial \bar{v}_0(t_1)} = \cdots = \frac{\partial \phi_K}{\partial \bar{v}_0(t_K)} = H$. The Jacobian matrix is equal to

$$
\begin{aligned}
&(\Gamma^{-1}\Omega_{T_c}\Gamma) \otimes I_N - (I_K \otimes H) && (24)\\
&= (\Gamma^{-1} \otimes I_N)(\Omega_{T_c} \otimes I_N)(\Gamma \otimes I_N) - (I_K \otimes H) && (25)\\
&= (\Gamma^{-1} \otimes I_N)\{(\Omega_{T_c} \otimes I_N) - && (26)\\
&\quad (\Gamma^{-1} \otimes I_N)^{-1}(I_K \otimes H)(\Gamma \otimes I_N)^{-1}\}(\Gamma \otimes I_N)\\
&= (\Gamma^{-1} \otimes I_N)\{(\Omega_{T_c} \otimes I_N) && (27)\\
&\quad -(\Gamma \otimes I_N)(I_K \otimes H)(\Gamma \otimes I_N)^{-1}\}(\Gamma \otimes I_N)\\
&= (\Gamma^{-1} \otimes I_N)\{(\Omega_{T_c} \otimes I_N) && (28)\\
&\quad -(I_K \otimes H)(\Gamma \otimes I_N)(\Gamma \otimes I_N)^{-1}\}(\Gamma \otimes I_N)\\
&= (\Gamma^{-1} \otimes I_N)\{(\Omega_{T_c} \otimes I_N) && (29)\\
&\quad -(I_K \otimes H)\}(\Gamma \otimes I_N).
\end{aligned}
$$

Equations (24) to (25) is because of $I_N = I_N I_N I_N$ and Lemma 5.1. Equations (26) to (27) is due to Lemma 5.2 (b), and equations (27) to (28) due to Lemma 5.2 (a). Since $(\Gamma^{-1} \otimes I_N)$ is unitary and its inverse is $(\Gamma \otimes I_N)^{-1}$, the right hand side of equation (29) has the same spectrum as $(\Omega_{T_c} \otimes I_N) - (I_K \otimes H)$. It is easy to verify that $(\Omega_{T_c} \otimes I_N) - (I_K \otimes H)$ is block diagonal, hence its eigenvalues are the union of eigenvalues of all the blocks, $e^{j2\pi\omega_k T_c} I_N - H$, for $k = 1, \cdots, K$. □

The preceding analysis suggests a good way of preconditioning for solving the Newton equation (21). Solving (21) is equivalent to solving

$$\{(\Omega_{T_c} \otimes I_N) - ((\Gamma \otimes I_N)J(\Gamma^{-1} \otimes I_N))\}y = (\Gamma \otimes I_N)b, \quad (30)$$

where $y = \Gamma\Delta\bar{v}^i$. A good choice of preconditioner is $P = (\Omega_{T_c} \otimes I_N) - (I_K \otimes H)$, where H can be chosen as the Jacobian matrix from the steady-state analysis in the initial guess stage discussed in Section 6, or any of the diagonal blocks, $\frac{\partial \phi}{\partial v(t_i)}$, for $i = 1, \cdots, K$, of $\frac{\partial \Phi}{\partial \bar{v}_0}$. In particular, if the single-cycle state-transition function is linear and time invariant, then the Newton equation can be solved in a single GMRES iteration. Note that the preconditioner presented here will be effective if the Jacobian of the state-transition function is nearly constant over multiple cycles. The circuit behavior *inside* each clock cycle is hidden from the preconditioner. This is not the case in, for example, the time- or frequency-averaged prenditioners typically used in modern harmonic balance[7, 6] codes. For this reason the preconditioner presented here may perform well under much weaker assumptions about the circuit behavior, in particular at higher power levels.

For each GMRES iteration, a system $Pu = v$ has to be solved. Since P is block diagonal, we need to solve a sequence of K systems $(e^{j2\pi\omega_k T_c} I_N - H)u_k = v_k$, for $k = 1, \cdots, K$, where $u^T = [u_1^T, \cdots, u_K^T]^T$ and $v^T = [v_1^T, \cdots, v_K^T]^T$. The preconditioner can be applied very efficiently by incorporating a Krylov subspace reuse algorithm[13], as the linear systems to be solved are the same as arise in the small-signal analysis for periodic time-varying systems. The basic idea of the algorithm is that the Krylov subspaces associated with the matrices $e^{i\omega T_c} - H$ are very similar for different ω_k. Essentially, the Krylov-subspace re-use algorithm allows the preconditioner for the matrix $D_{T_c} \otimes I_N - J$ to be applied with only slightly more cost than an iterative solve with the matrix H.

Figure 2 shows the effectiveness of this preconditioner in compressing the eigenvalues for an example RF receiver circuit.

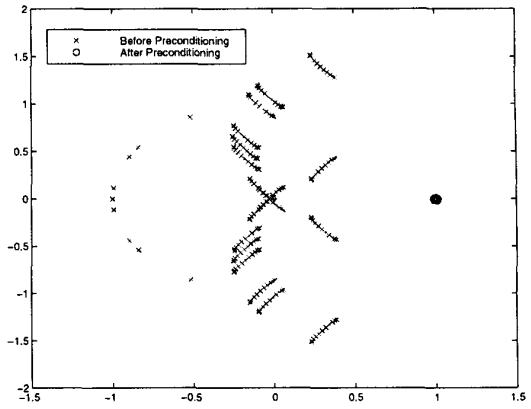

Figure 2: Eigenvalue distribution before and after preconditioning

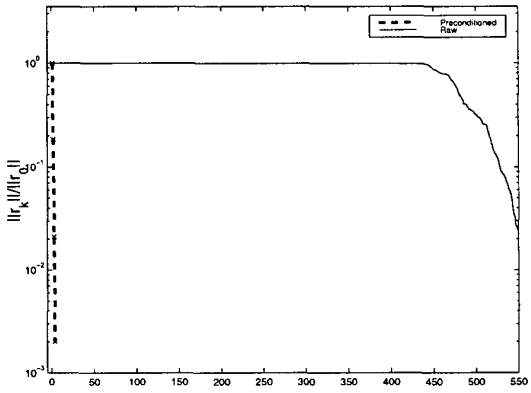

Figure 3: Convergence of GMRES with (dashed line) and without (solid line) preconditioner for one MFT Newton iteration.

The eigenvalues are *very* tightly clustered around unity, indicating excellent performance of the preconditioner and very rapid GMRES convergence.

Figure 3 shows the effectiveness of the preconditioner in reducing the number of GMRES iterations needed to solve each MFT Newton update equation, for the same RF circuit mentioned above. Only three iterations were needed to reduce the residual by a factor of 10^{-2}, whereas without the preconditioner, over 400 iterations are needed to achieve any reduction in the residual at all.[2]

6 Improving Newton convergence

Rapid convergence of Newton's method can only be assured with a good initial guess. To achieve a good initial guess, we first calculate the periodic steady state response of the circuit with the clock signal applied, while suppressing other non-DC signals. Using

[2]Since the MFT circuit equations are not solved exactly, we have found that, on average, there is a performance advantage in the MFT method to using approximate solves of the Newton update equation, and therefore GMRES is converged to a relatively loose tolerance.

the steady state solution as an operating point, a small-signal[13] analysis is performed by treating non-clock fundamentals as small signals. As a result of the small signal analysis, amplitudes at $f_s + k_s f_c$, for $-K_s \leq k_s \leq K_s$, $1 \leq s \leq S$, are generated. These amplitudes are transformed into time domain initial conditions via inverse Fourier transform. At higher input power levels, using a Newton continuation [1] method, with the amplitude of the non-clock signals as the continuation parameter, is generally effective in securing convergence.

7 Spectrum calculation

After the solution is converged, the values $\bar{v} = [v(t_1)^T, v(t_2)^T, \cdots, v(t_K)^T]^T$ and the integration solution in $[t_i, t_i + T_c]$, $i = 1, \cdots, K$ are available. From these pieces of information, the spectrum $v(t)$ can be obtained. Let

$$v(t) = \sum_{k_c=-K_c}^{K_c} \sum_{k_1=-K_1}^{K_1} \cdots \sum_{k_S=-K_S}^{K_S} V(k_1, \cdots, k_S, k_c) \times e^{-j2\pi k_1 f_1 t} \cdots e^{-j2\pi k_S f_S t} e^{-j2\pi k_c f_c t}. \quad (31)$$

Define $\bar{v}(\tau) = [v(t_1 + \tau)^T, v(t_2 + \tau)^T, \cdots, v(t_K + \tau)^T]^T$. Then

$$\bar{v}(\tau) = \{(\Gamma^{-1}\Omega(\tau)) \otimes I_N\} \times \begin{bmatrix} \vdots \\ \sum_{k_c=-K_c}^{K_c} V(k_1, \cdots, k_S, k_c) e^{-j2\pi k_c f_c \tau} \\ \vdots \end{bmatrix}. \quad (32)$$

Then for each KN-vector $V(\cdot, k_c)$, where $-K_c \leq k_c \leq K_c$, which is collection of all N-vectors $V(k_1, \cdots, k_S, k_c)$, where $-K_1 \leq k_1 \leq K_1, \cdots, -K_S \leq k_S \leq K_S$ (the actual order is determined by the Fourier transform),

$$V(\cdot, k_c) = \frac{1}{T_c} \int_0^{T_c} \{(\Omega(\tau)^{-1}\Gamma) \otimes I_N\} \bar{v}(\tau) e^{j2\pi k_c f_c \tau} d\tau \quad (33)$$

Forming $\{(\Omega(\tau)^{-1}\Gamma) \otimes I_N\}\bar{v}(\tau)$ requires the values for $v(t_1 + \tau), \cdots, v(t_K + \tau)$, or synchronized time steps between cycles. The total cost is one KN-vector integration and M Fourier transforms, where M is the number of synchronized time points.

The synchronized time step requirement may not be easily met in practice. One alternative is to use interpolation schemes. However they potentially lose accuracy. Another alternative is to trade integrations for Fourier transforms. Specifically, it is easy to verify that

$$\begin{aligned} & V(k_1, \cdots, k_S, k_c) \quad (34) \\ = & \frac{1}{T_c} \int_0^{T_c} E_p^T \{(\Omega(\tau)^{-1}\Gamma) \otimes I_N\} \bar{v}(\tau) e^{j2\pi k_c f_c \tau} d\tau \\ = & E_p^T (\Gamma \otimes I_N)(\frac{1}{T_c} \int_0^{T_c} \bar{v}(\tau) e^{j2\pi(\sum_{i=1}^{S} k_i f_i)\tau} e^{j2\pi k_c f_c \tau} d\tau), \end{aligned}$$

where E_p is a $KN \times N$ block matrix whose pth $N \times N$ block is I_N and other blocks zero, and p is determined by $(k_1, \cdots, k_S)$ from the Fourier transform. Calculating (34) does not require synchronized time points. The total cost of calculating $V(\cdot, k_c)$ is K KN-vector integrations plus one final Fourier transform. However, it might be more expensive since integrations normally cost more than Fourier transforms.

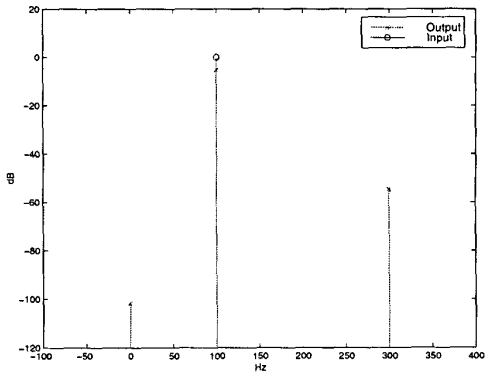

Figure 4: Harmonic distortion of a switched-capacitor filter

8 Test results

8.1 Switched-capacitor Filter

The first example is a low-pass switched-capacitor filter of 4kHz bandwidth and having 238 nodes, resulting in 337 equations. To analyze this circuit, the MFT analysis was performed with an 8-phase 100kHz clock and a $1V$ sinusoidal input at 100Hz.

The 1000 to 1 clock to signal ratio makes this circuit difficult for traditional circuit simulators to analyze. In the MFT method, three harmonics were used to model the input signal. The eight-phase clock resulted in the need to use about 1250 timepoints in each transient integration. This brings the total number of variables solved by the analysis to slightly less than three million ($337 \times (2 \times 3 + 1) \times 1250 = 2,948,750$). The simulation took a little less than 20 minutes CPU time to finish, on a Sun UltraSparc1 workstation with 128 Megabyte memory and a 167MHz CPU clock. Figure 4 shows the output spectrum of the filter.

8.2 High-performance receiver

The second example is the high-performance image rejection receiver also discussed in [14]. It consists of a low-noise amplifier, a splitting network, two double-balanced mixers, and two broad-band Hilbert transform ouput filters combined with a summing network that is used to supress the undesired side-band. A limiter in the LO path is used for controlling the amplitude of the LO. It is a rather large RF circuit that contains 167 bipolar transistors and uses 378 nodes. This circuit generated 987 equations in the simulator.

To determine the intermodulation distortion characteristics, the circuit was driven by a 780MHz LO and two 50mV closely placed RF inputs, at 840MHz and 840MHz+10KHz, respectively. Three harmonics were used to model each of the RF signals. 200 time points were used in each transient clock-cycle integration, considered to be conservative in terms of accuracy for this circuit. As a consequence, nearly ten million unknowns ($987 \times (2 \times 3 + 1)^2 \times 200 = 9,672,600$) were generated. It took 55 CPU minutes to finish on a Sun UltraSparc10 workstation with 128 Megabytes of physical memory and a 300MHz CPU clock. Figure 5 shows 3rd and 5th order distortion products.

To understand the efficiency of the MFT method, consider that traditional transient analysis would need at least 80,000 cycles of the LO to compute the distortion, a simulation time of over two days. Additionally, the results would be very inaccurate, because

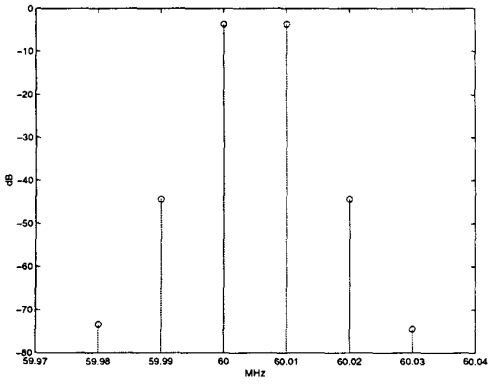

Figure 5: Intermodulation distortion of a high-performance receiver

of the large amount of numerical error accumulated by integrating over so many cycles. In contrast, the MFT method is able to resolve very small signal levels, such as the 5th order distortion products show in Figure 5.

Solving the MFT equations by direct factorization methods is also impractical, as the storage needed for the factored rank-50,000 ($987 \times (2 \times 3 + 1)^2 = 48,363$) MFT Jacobian of Equation 19 is several gigabytes. Forming the Jacobian matrix by direct methods would also require computation time proportional to the cost of 50,000 transient integrations, again a number on the order of days.

9 Conclusion

In this paper we have demonstrated that the MFT method is an efficient approach to analyzing multi-frequency nonlinear effects such as intermodulation distortion. Making the MFT method computationally efficient on problems of engineering interest required careful construction of the delay matrix, matrix-implicit Krylov subspace iterative linear solvers, and a preconditioner tailored to the MFT method and the circuits it typically analyzes. As a result, nonlinear systems comprising tens of millions of unknowns can be solved in less than an hour with computational resources commonly available to engineering designers.

One salient advantage of the MFT method as described here is that the dominant part of the computation is in computing the functions ϕ and the product of the Jacobian of ϕ with some vector. Both computations are essentially the solution of an initial value problem. Each application of the operator $D_{T_c} - J$, or calculation of the Newton residual, involves solving K such initial value problems, that is, integrating K sets of DAEs forward in time over one clock period. Each of the K problems, however, is essentially decoupled. Parallel implementations of the MFT will therefore enjoy very efficient processor utilization. This decoupling also assists the implementation of out-of-core solvers. In fact we have observed it is possible to implement the MFT algorithm as an out-of-core algorithm with over 80% average CPU utilization.

Several possible extensions of this work are possible. Of particular interest is the computation of poly-cyclostationary[10] noise statistics that can be performed by linearizing around the the quasiperiodic steady-state.

References

[1] A. ALLGOWER AND K. GEORG, *Numerical Continuation Methods*, Springer-Verlag, New York, 1990.

[2] L. O. CHUA AND A. USHIDA, *Algorithms for computing almost periodic steady-state response of nonlinear systems to multiple input frequencies*, IEEE Trans. Circuits and Systems, 28 (1981), pp. 953–971.

[3] 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, 24 (1989), pp. 443–451.

[4] K. S. KUNDERT, J. K. WHITE, AND A. SANGIOVANNI-VINCENTELLI, *Steady-State Methods for Simulating Analog And Microwave Circuits*, Kluwer Academic Publishers, Boston, 1990.

[5] P. LANCASTER AND M. TISMENETSKY, *The Theory of Matrices*, Academic Press, second ed., 1985.

[6] D. LONG, R. MELVILLE, K. ASHBY, AND B. HORTON, *Full chip harmonic balance*, in Proceedings of the Custom Integrated Circuits Conference, May 1997.

[7] R. MELVILLE, P. FELDMANN, AND J. ROYCHOWDHURY, *Efficient multi-tone distortion analysis of analog integrated circuits*, in Proceedings of the Custom Integrated Circuits Conference, May 1995.

[8] M. OKUMURA, T. SUGAWARA, AND H. TANIMOTO, *An efficient small signal frequency analysis method for nonlinear circuits with two frequency excitations*, IEEE transactions of computer-aided design of integrated circuits and systems, 9 (1990), pp. 225–235.

[9] J. ROYCHOWDHURY, *Efficient methods for simulating highly nonlinear multirate circuits*, in Proceedings of the 34^{th} Design Automation Conference, Anaheim, CA, June 1997, pp. 269–274.

[10] J. ROYCHOWDHURY, D. LONG, AND P. FELDMANN, *Cyclostationary noise analysis of large RF circuits with multitone excitations*, IEEE J. Sol. St. Circuits, 33 (1998), pp. 324–336.

[11] Y. SAAD AND M. H. SCHULTZ, *GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems*, SIAM J. Sci. Stat. Comput., 7 (1986), pp. 856–869.

[12] R. TELICHEVESKY, K. S. KUNDERT, AND J. K. WHITE, *Efficient steady-state analysis based on matrix-free Krylov-subspace methods*, in Proceedings of the 1995 Design Automation Conference, June 1995.

[13] ——, *Efficient AC and noise analysis of two-tone RF circuits*, in Proceedings of the 1996 Design Automation Conference, June 1996.

[14] R. TELICHEVESKY, J. WHITE, AND K. KUNDERT, *Receiver characterization using periodic small-signal analysis*, in Proceedings of the Custom Integrated Circuits Conference, May 1996.

[15] Y. THODESEN, *Two stage method for efficient simulation of parametric circuits*, PhD thesis, Department of telecommunications, the Norwegian institute of technology, 1996.