A Robust Periodic Arnoldi Shooting Algorithm for Efficient Analysis of Large-scale RF/MM ICs *

Xue-Xin Liu†, Hao Yu‡, and Sheldon X.-D. Tan†
† Department of Electrical Engineering, University of California, Riverside, CA 92521
‡ School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore
xliu@ee.ucr.edu, haoyu@ntu.edu.sg, stan@ee.ucr.edu

ABSTRACT

The verification of large radio-frequency/millimeter-wave (RF/MM) integrated circuits (ICs) has regained attention for high-performance designs beyond 90nm and 60GHz. The traditional time-domain verification by standard Krylov-subspace based shooting method might not be able to deal with newly increased verification complexity. The numerical algorithms with small computational cost yet superior convergence are highly desired to extend designers' creativity to probe those extremely challenging designs of RF/MM ICs. This paper presents a new shooting algorithm for periodic RF/MM-IC systems. Utilizing a periodic structure *of the state matrix, a periodic Arnoldi shooting algorithm is developed to exploit the structured Krylov-subspace. This leads to an improved efficiency and convergence. Results from several industrial examples show that the proposed periodic Arnoldi shooting method, called PAS, is 1000 times faster than the direct-LU and the explicit GMRES methods. Moreover, when compared to the existing industrial standard, a matrix-free GMRES with non-structured Krylov-subspace, the new PAS method reduces iteration number and runtime by 3 times with the same accuracy.*

Categories and Subject Descriptors

B.7.2 [**Integrated Circuits**]: Design Aids—simulation

General Terms

Algorithms, Design, Performance

Keywords

Periodic Steady-State Analysis, Shooting Newton Algorithm, Krylov Subspace

1. INTRODUCTION

For radio-frequency (RF) and millimeter-wave (MM) integrated circuits (ICs), there is a second scaling direction along the operating frequency in addition to the device size [1–3]. The use of 7GHz unlicensed spectrum around 60GHz provisions a bandwidth for many high data-rate applications like multimedia processing, sensor beamforming and car-radar relaying. A 2–4Gb/s bandwidth wireless link can open many applications such as reporting a patient medical information to a clinician, joining a high-definition (HD) TV and a HD-DVD and uploading a movie to iPhone, or preventing car collision and reporting traffic condition. However, the design for RF/MM-IC front-end circuits at this frequency region is very challenging [4]. At the scale of 60GHz, all active and passive devices are closely coupled and the resulting post-layout circuit model has a drastically increased complexity. Moreover, to follow the high-frequency carrier, a traditional transient analysis requires to use small time-steps and hence results in a long simulation time. A fast RF/MM-IC simulation is thereby needed to deal with the design complexity and reduce the design cycle.

Since most RF/MM-IC designs are with periodic inputs, a periodic steady state (PSS) analysis [5–10] is generally employed to calculate an operating point for subsequent small-signal analysis and predict distortion, noise and etc. This is similar to a DC analysis. There are three different means to find a PSS solution: time-domain finite difference method (FDM), frequency-domain harmonic balance (HB) and time-domain shooting method [7, 10]. Due to the need to form a large block matrix, FDM is not commonly used to compute steady-state solutions. On the other hand, HB algorithm is usually employed for mildly nonlinear circuits with multi-tone inputs, and the shooting Newton method is better suited for more drastically nonlinear circuits.

This paper focuses on the time-domain shooting-Newton method. Compared to the traditional transient analysis solving an initial-value problem (IVP), a shooting method is to solve a two-point boundary-value problem (BVP) at one period on top of the transient analysis [11]. Different from harmonic balance, shooting method has a better convergence for strongly nonlinear circuits due to the underlying transient analysis with adaptive time-steps. However, direct solving of the shooting-Newton equations is expensive since it forms a dense and shifted shooting-Jacobian. Though an iterative GMRES with the use of a standard Krylov-subspace [12] can alleviate the cost, the explicit formulation of the shifted shooting-Jacobian could still be expensive. An implicit 'matrix free' approach [8] avoids the formulation of the shifted shooting-Jacobian and hence improves the speed of the GMRES method for the shooting.

However, the aforementioned GMRES methods are all based on the standard Krylov-subspace methods, which ig-

*This work is funded in part by NSF grant under No. CCF-0448534, in part by NSF grant under No. OISE-0929699.

DAC'10, June 13-18, 2010, Anaheim, California, USA

nore the periodic structure of the underlying Krylov-subspace. In this paper, we show that the two-point boundary condition problem for the PSS solution leads to a periodic structured Krylov-subspace. If one captures this structure when identifying the Krylov-subspace, the convergence and efficiency can be improved. Accordingly, we propose a robust yet efficient periodic Arnoldi shooting method, named *PAS*, which can build a structured Krylov-subspace for the shooting-Newton iteration. The numerical results show that our proposed PAS method is 1000 times faster than the direct-LU method and explicit GMRES. More importantly, under the similar error bound, PAS reduces the iteration number and runtime by 3 times when compared to the state-of-the-art matrix-free GMRES using the regular non-structured Krylov-subspace.

In the remainder part of the paper, the background is reviewed including the periodic steady state (PSS) analysis, shooting-Newton method and the existing explicit and implicit Krylov-subspace based GMRES methods. A periodic Arnoldi shooting (PAS) algorithm is then introduced to identify the periodic structured Krylov-subspace. Finally, experimental results are presented by a few industrial design examples, and the paper is concluded with a summary.

2. PERIODIC STEADY STATE

A nonlinear RF/MM-IC circuit can be described in general by a differential-algebra-equation (DAE) shown below,

$$f(x(t),t) = \frac{d}{dt}q(x(t)) + j(x(t)) + u(t) = 0. \tag{1}$$

where $x \in R^n$ is the state variable vector including nodal voltage and branch current, $j(x(t))$ is for a nonlinear node current, $q(x(t))$ is for a nonlinear node charge or flux, and $u(t)$ is for an external periodic source.

DEFINITION 1. *A state vector $x : R \to R^n$ is called a periodic steady-state (PSS) of (1) if:*

- *x is a solution of (1) and*
- *x is periodic, i.e., there is a $T > 0$ such that for all $t_0 \in R$, $x(t_0) = x(t_0 + T)$, called two-point boundary constraint.*

Thereby, obtaining the PSS solution is equivalent to finding the initial condition $x(t_0)$ for the circuit-associated DAE (1) such that the solution $x(t_0 + T)$ at the end of one period T agrees with the initial condition $x(t_0)$.

3. SHOOTING-NEWTON METHOD

DEFINITION 2. *A state-transition function $\phi_T(x_0, t_0)$ is the solution of (1) at $t_0 + T$, starting from a guessed initial state $x(t_0)$ at t_0, or*

$$x(t_0 + T) = \phi_T(x(t_0), t_0), \tag{2}$$

and one can have the following two-point boundary constraint

$$\phi_T(x(0), 0) = x(0) \tag{3}$$

for one period with $t_0 = 0$.

A shooting-sensitivity, or called shooting-Jacobian, can be further defined by

$$J_{\phi_T}(x(0), 0) = \frac{dx(T)}{dx(0)} = \frac{d\phi_T(x(0), 0)}{dx(0)}. \tag{4}$$

As discussed below, J_{ϕ_T} determines how to update the initial state $x(0)$ in order to reduce its difference from the final state $x(T)$.

Since (3) is in the nonlinear algebraic form, Newton iteration is deployed to solve for $x(0)$ or denoted as x_0 (and $x(T)$ as x_T). Such a method is so-called shooting-Newton algorithm [5, 7, 8, 10] with the following Newton iteration

$$x_0^k = x_0^{k-1} + \left[I - J_{\phi_T}(x_0^{k-1}, 0)\right]^{-1} \left[\phi_T(x_0^{k-1}, 0) - x_0^{k-1}\right] \tag{5}$$

where I is the identity matrix and k is the iteration number.

Therefore, one first obtains the transient response of the circuit over one period and then J_{ϕ_T} gives how much correction is needed to the solution x_0 from the update of the final state x_T with respect to the initial state x_0.

Specifically, J_{ϕ_T} is obtained by the following manner. One first integrates the DAE (1) in one period, for example, by the Backward-Euler method

$$\frac{1}{h_j}\left[q(x_j) - q(x_{j-1})\right] + j(x_j) + u_j = 0, \tag{6}$$

where h_j is the j^{th} time-step: $h_j = t_j - t_{j-1}$, with $t_0 = 0$, $t_p = T$ for total p time-steps.

Note that (6) becomes an algebraic equation and can now be solved by a so-called inner transient-Newton iteration, distinguished from the outside shooting-Newton. As such, the linearized (6) at l^{th} transient-Newton iteration becomes

$$\left[G(x_j^{l-1}) + \frac{C(x_j^{l-1})}{h_j}\right](x_j^l - x_j^{l-1}) = - \frac{1}{h_j}\left(q(x_j^{l-1}) - q(x_{j-1})\right) - j(x_j^{l-1}) - u_j \tag{7}$$

with $G(x_j^{l-1}) = dj(x_j^{l-1})/dx$ and $C(x_j^{l-1}) = dq(x_j^{l-1})/dx$.

Moreover, when differentiating (6) with respect to x_0, the following relation can be obtained

$$\left[G(x_j) + \frac{C(x_j)}{h_j}\right]\frac{dx_j}{dx_0} = \frac{C(x_{j-1})}{h_j}\frac{dx_{j-1}}{dx_0}. \tag{8}$$

When this is applied recursively following the chain-rule for all time steps in one period, one can eventually obtain the shooting-Jacobian by

$$J_{\phi_T} = \prod_{j=1}^{p}\left[G(x_j) + \frac{C(x_j)}{h_j}\right]^{-1}\frac{C(x_{j-1})}{h_j}. \tag{9}$$

Note that as the transient analysis is first applied, the matrices $G(x_j) + \frac{C(x_j)}{h_j}$, $j = 1, \ldots, p$, are already available and are stored as LU-factored sparse matrices. Therefore the computational cost is mainly from the n^2 backward and forward evaluations of forming the dense J_{ϕ_T}, and the n^3 flops of the direct LU-factorization of (5).

4. NON-STRUCTURED KRYLOV SUBSPACE METHOD

To reduce the computational cost when solving the shooting-Newton, the traditional non-structured Krylov-subspace method with a GMRES iteration is illustrated in Algorithm 1.

We further define

$$A = I - J_{\phi_T}, \quad b = \phi_T(x_0^{k-1}, 0) - x_0^{k-1}$$

ALGORITHM 1: A NON-STRUCTURED KRYLOV-SUBSPACE BASED GMRES

Inputs: b and pre-factored $\left(G(x_j)+\frac{C(x_j)}{h_j}\right)$ $(j = 1,\ldots,p)$

1. Guess an initial solution: δx_0^0;
2. Initialize the first Krylov base: $v^0 = b - A\delta x_0^0$;
3. Iterate with index i till: $||r^i|| < tol$;
4. Compute the new Krylov base: $v^i = Av^{i-1}$;
5. Orthogonalized: $v^i = v^i - \sum_{m=0}^{i-1}\beta_{im}v^j$ (β_{im} is orth-coeff);
6. Find $\delta x_0^i = \delta x_0^{i-1} + \alpha_i v^i$ to minimize $||r^i||$ (α_i is lms-coeff);
7. End iteration of i;
Outputs: δx_0^i

A MATRIX-FREE COMPUTATION OF THE NEW KRYLOV BASE

4.1 Replace $v^i = Av^{i-1}$ by $v^i = J_\phi v^{i-1}$;
4.2 Initialize $w = v^{i-1}$;
4.3 Iterate with index j for p time-steps:
4.4 Solve $\left(G(x_j)+\frac{C(x_j)}{h_j}\right)v^i = \frac{C(x_j)}{h_j}w$;
4.5 Set $w = v^i$;
4.6 End iteration of j;
4.7 Update $v^i = v^{i-1} - v^i$ to consider the shifted I;

The update

$$\delta x_0 = x_0^k - x_0^{k-1}$$

is the solution of

$$A \times \delta x_0 = b$$

based on (5). As J_{ϕ_T} is dense and so is $A = I - J_{\phi_T}$, direct LU decomposition of A has a complexity of $O(n^3)$. Instead, an iterative GMRES method with a regular non-structured Krylov-subspace can be employed as shown in Algorithm 1. Note that the orthogonalized bases

$$span\{v^0, v^1, ..., v^i ..., \}$$

compose of the traditional non-structured Krylov-subspace.

Clearly, explicitly forming A involves n^2 evaluations and n^2 multiplies for Av^{i-1} for each GMRES iteration. To further reduce the computational cost, one way is to avoid the explicit formulation of matrix of A. The work in [8, 10] introduced a 'matrix free' approach to directly calculating the matrix-vector-product Av^{i-1} without forming A. As shown in the subroutine, the 'matrix free' calculates the product of $J_{\phi_T}v^{i-1}$ instead of Ap^{i-1} (4.1) by iteratively reusing the pre-factored chain-rule values (4.4). The shift by I is finally corrected in v^i (4.7).

The GMRES method with the standard Krylov-subspace based iteration converges fast only when the eigenvalues of J_{ϕ_T} has a clustered distribution (close to 1). In other words, the circuit time-constants need to be much smaller than the period T. Unfortunately this is not always the case for RF/MM-IC designs since the time-period T become very small at this frequency scale. For example, Fig. 1 shows that when increasing the carrier frequency, the eigenvalue distribution of J_{ϕ_T} becomes more widely distributed, which makes the convergence of the GMRES slower. As such, one may need to find a more robust solution when searching PSS solution. Considering the underlying structure of Krylov-subspace is definitely one of promising solutions.

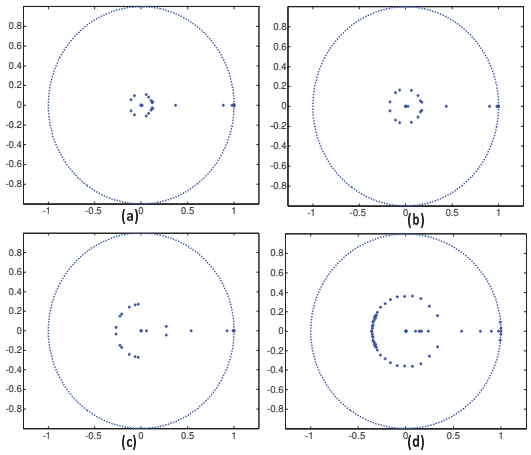

Figure 1: The eigenvalue distribution of the periodic cmos-frequency-multiplier under different input periods: (a)500MHz; (b) 600MHz; (c) 800MHz; and (d)1GHz.

5. PERIODIC ARNOLDI SHOOTING — PAS

The new algorithm is based on the observation that the Krylov-subspace of a periodic system usually contains a periodic (cyclic-block) structure. We show that if one can identify the underlying Krylov-subspace with the consideration (or preservation) of the periodic structure, the convergence and efficiency of the GMRES can be further improved. The proposed periodic Arnoldi method, *PAS*, can identify and explore a periodic structured Krylov-subspace. In the following, we explain in detail the new algorithm.

5.1 Conventional Arnoldi Method in GMRES

Before surfing into the periodic Arnoldi method, we first briefly review the application of the traditional Arnoldi method [13] for the GMRES iteration.

An Arnoldi process begins with a starting base

$$v^1 = r^0/\alpha \quad (\alpha = ||r^0||).$$

After a sequential of iterations with the orthonormalization, one m^{th}-order subspace V^m can be defined as

$$V^m = [v^1, v^2, \ldots, v^m],$$

which satisfies an Arnoldi decomposition

$$AV^m = V^{m+1}H^m.$$

Here, H^m is a Hessenberg matrix with an upper-block-triangular structure. As a result, the projection of A by the subspace V^m results in a structured matrix operation with a smaller computational cost.

By further mapping δx_0 ($\in R^n$) into a subspace spanned by V^m ($\in R^{n\times m}$, $m < n$)

$$\delta x_0 = V^m z,$$

one can obtain the residue

$$r^m = r^0 - AV^m z = V^{m+1}\left(\alpha u^1 - H^m z\right) \qquad (10)$$

where u^1 is the unit vector in the spanned subspace that satisfies $v^1 = V^{m+1}u^1$.

Thereby, if one can generate an m^{th}-order Hessenberg matrix H^m such that

$$||H^m z - \alpha u^1|| < tol/||V^m||. \quad (11)$$

As such, the GMRES converges under one pre-specified error tolerance tol.

5.2 Periodic Structured Krylov Subspace

Note that the RF/MM-IC system with periodic inputs is in fact a periodic system. Hence the associated Krylov-subspace would also exhibit a periodic structure. As such, one may exploit the structure and structure-preservation in Krylov-subspace based GMRES for the periodic system.

Let's first identify the periodic structure of the underlying Krylov-subspace. For $j = 1, \ldots, p$, defining

$$G_j = G(x_j),\ C_j = C(x_j),\ A_j = \left[G_j + \frac{C_j}{h_j}\right]^{-1} \frac{C_{j-1}}{h_j}, \quad (12)$$

the shooting-Jacobian J_{ϕ_T} in (9) becomes

$$J_{\phi_T} = A_p \times A_{p-1} \times \cdots \times A_1. \quad (13)$$

In the following, we show that the multiplied product J_{ϕ_T} has an identical invariant subspace as the following cyclic-block matrix

$$\mathcal{J} = \begin{bmatrix} 0 & & & A_p \\ A_1 & \ddots & & \\ & \ddots & & \\ & & A_{p-1} & 0 \end{bmatrix}, \quad (14)$$

which has a periodic Krylov-subspace determined through a periodic Arnoldi method [14].

Definition 3. *The periodic Krylov-subspace $\mathcal{V}^m$ of $\mathcal{J}$ is defined through the following periodic Arnoldi decomposition*

$$\mathcal{J}\mathcal{V}^m = \mathcal{V}^{m+1}\mathcal{H}^m, \quad (15)$$

where

$$\begin{aligned} \mathcal{V}^m &= V_1^m \oplus V_2^m \cdots \oplus V_p^m \\ \mathcal{V}^{m+1} &= V_1^{m+1} \oplus V_2^m \cdots \oplus V_p^m \end{aligned} \quad (16)$$

and

$$\mathcal{H}^m = \begin{bmatrix} 0 & & & H_p^m \\ H_1^m & \ddots & & \\ & \ddots & & \\ & & H_{p-1}^m & 0 \end{bmatrix}. \quad (17)$$

Operation $\oplus$ denotes the direct sum operation of two matrices: given $A \in R^{m\times n}$ and $B \in R^{p\times q}$, $A \oplus B$ is a $(m+p)\times(n+q)$ matrix with A and B in the diagonal. Clearly, the periodic Krylov-subspace of $\mathcal{J}$ characterized by $\mathcal{V}^m$ and $\mathcal{H}^m$, is composed of block matrices V_j^m and H_j^m, $j = 1, \ldots, p$. The following lemma further reveals that the V_j^m and H_j^m can be used to identify the periodic Krylov-subspace for J_{ϕ_T}.

Lemma 1. *The periodic system with a cyclic-block matrix $\mathcal{J}$, characterized by ($\mathcal{V}^m$ and $\mathcal{H}^m$) has an identical Krylov-subspace with the periodic system with a multiplied-product matrix J_{ϕ_T}, characterized by V_j^m and H_j^m, $j = 1, \ldots, p$.*

Proof: Starting from the periodic Arnoldi decomposition (15), one can replace the $\mathcal{J}$, $\mathcal{V}^m$ and $\mathcal{H}^m$ in (15) by block matrices, using the relations in (14), (16) and (17). Therefore, one can obtain the following periodic Arnoldi iteration at a block-matrix level

$$\begin{cases} J_{\phi_T} V_1^m = V_1^{m+1}\mathbf{H}_1^m & \text{if } j = 1 \\ J_{\phi_T}^j V_j^m = V_j^{m+1}\mathbf{H}_j^m & \text{if } j > 1 \end{cases},$$

where

$$\mathbf{H}_j^m = H_{j+p-1}^m \times H_{j+p-2}^m \times \cdots \times H_j^m.$$

When the order of multiplication is fixed for J_{ϕ_T}, i.e. (13), the case of relation for $j = 1$ is applied for the Arnoldi iteration. When the order of multiplication is permuted j times for J_{ϕ_T}, the case of relation for $j > 1$ is applied for the Arnoldi iteration. With both, a series of periodic Krylov-subspaces $V_1^m, \ldots, V_p^m$ are obtained, which further composes of the periodic Krylov-subspace $\mathcal{V}^m$. Q.E.D.

Though only the case of $j = 1$ is needed in the GMRES iteration, a procedure is required to find all block matrices V_j^m and H_j^m, $j = 1, \cdots, p$. This is resolved by the periodic Arnoldi method described in Algorithm 2 below, which compute all the V_i^m and H_i^m.

5.3 PAS Algorithm

Algorithm 2: A matrix-free periodic Arnoldi method

Inputs: Pre-factored $\left(G(x_j) + \frac{C(x_j)}{h_j}\right)$ $(j = 1, \ldots, p)$

1. Initialize V_1^1 by v^1 and $V_0^i, (i = 2, \ldots, p)$ by 0;
2. Iterate index i for subspace orders to m;
4. Iterate index j for periodic blocks to p;
5. Set $h_j^i = V_{j+1}^{i-1} A_j v_i^j$, $w = A_j v_i^j - V_{j+1}^{i-1} h_j^i$;
$g_j^i = ||w||_2$, $H_j^i = \begin{bmatrix} H_j^{i-1} & h_j^i \\ 0 & g_j^i \end{bmatrix}$;
$v_{j+1}^i = w/g_j^i$, $V_{j+1}^i = [V_{j+1}^{i-1}, v_{j+1}^i]$;
6. End iteration of j;
7. End iteration of i;

Outputs: Block matrices V_j^m and H_j^m $(j = 1, \ldots, p)$

We call the overall procedure in Algorithm 2 as periodic Arnoldi shooting (PAS) method. In the PAS based GMRES, the matrix-free based GMRES in Algorithm 1 can be employed except for two parts: (1) when generating the Krylov bases v^m for $J_{\phi_T} \times \delta x_0$; and (2) when computing residue. In Algorithm 2, a periodic Arnoldi method is employed to generate the periodic Krylov-subspace, i.e., the block matrices V_j^m and H_j^m, $j = 1, \ldots, p$, as suggested by Lemma 1. Here, the subscript j denotes the index of the periodic blocks ($j = 1, \ldots, p$), and the superscript i denotes the index of the order of the Krylov-subspace ($i = 1, \ldots, m$). Similar to the concept of 'matrix free' in Algorithm 1, there is no need to form each A_j defined in (12), but the product of $A_j \times v_j^i$ is directly calculated by solving $\left(G(x_j) + \frac{C(x_j)}{h_j}\right)^{-1} \frac{C(x_{j-1})}{h_j} v_j^i$, where pre-factored sparse LUs of $\left(G(x_j) + \frac{C(x_j)}{h_j}\right)$ are reused.

In addition, same as in Algorithm 1, after the correction to consider the shifted I, Algorithm 2 produces the following Hessenberg matrix

$$\mathbf{H}_1^m = H_p^m \times H_{p-1}^m \times \cdots \times H_1^m$$

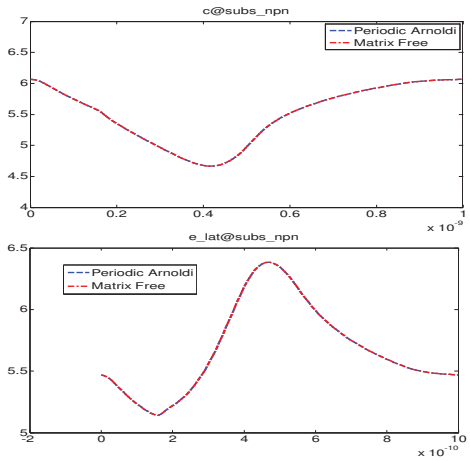

Figure 2: The PSS waveform accuracy comparison at two nodes of a BJT-mixer. The red-line is the non-structured MF-GMRES, and the blue-line is the structured PAS-GMRES.

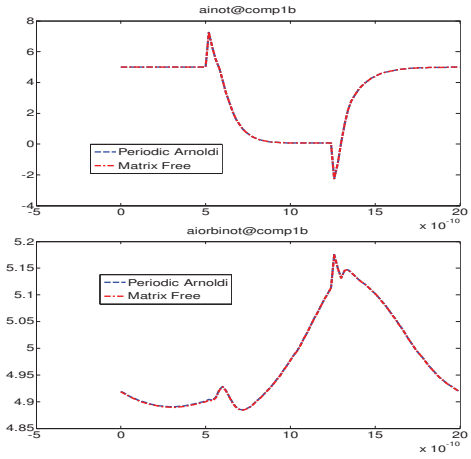

Figure 3: The PSS waveform accuracy comparison at two nodes of a cmos-switch-cap. The red-line is the non-structured MF-GMRES, and the blue-line is the structured PAS-GMRES.

which is embedded into the GMRES iteration by (10) to calculate the residue till the convergence tolerance is satisfied.

However, different from Algorithm 1, the periodic structure of the generated Krylov-subspace is preserved in Algorithm 2 because each orthnormalized base v_j^i is constructed separately for each A_j. In contrast, the base of the Krylov-subspace in Algorithm 1 is generated for the overall $J_{\phi_T} = A_1 \times \cdots \times A_p$ in Algorithm 1. As a result, the periodic structure of the underlying Krylov-subspace is destroyed in Algorithm 1.

The complexity and convergence of a periodic Arnoldi method is similar to the standard Arnoldi method. At one Arnoldi iteration, there would be p matrix-vector multiplications for A_j's. Due to the use of 'matrix free' formulation, this cost is reduced to p right-hand-solve reusing the sparse LUs. Moreover, $\mathbf{H}_1^m$ can be formed efficiently as each H_j^m is a triangular-block-structured matrix. More importantly, since our periodic Arnoldi method identifies the Krylov-subspace with the consideration of the periodic structure, experiments show significant speedup and better convergence in comparison with the non-structured Krylov-subspace based method.

6. EXPERIMENTAL RESULTS

The proposed periodic Arnoldi shooting, called PAS-GMRES in the following, is implemented in Matlab. The direct-LU, the explicit GMRES with the non-structured Krylov-subspace, and the matrix-free GMRES with the non-structured Krylov-subspace, are implemented for the comparison. The non-structured Krylov-subspace is identified through the least-minimum-square constraint using a template from [15]. The matrix-free GMRES is implemented exactly following the procedure described in [8, 10], and is called MF-GMRES in the following. The GMRES iteration tolerance is set as 10^{-6} for voltage nodes. All experimental results are measured on an Intel core-duo server with 2.4GHZ CPU and 4GB memory. We compare the accuracy and runtime with a scalability study using 5 industrial analog/RF examples, including a CMOS-dc-converter, a CMOS-frequency-multiplier, a BJT-mixer, a CMOS-low-noise-amplifier (LNA), and a CMOS-switch-cap. We increase the complexity by adding extracted parasitics.

Table 1 summarizes the simulation results for all examples. Column-1 shows the type of circuits, Column-2 shows the numbers of equations, Column-3 shows the runtime of direct-LU, Column-4 shows the runtime of explicit GMRES, Column-5 shows the iteration number and runtime of the non-structured MF-GMRES method, and Column-6 shows the iteration number and runtime of the structured PAS-GMRES method.

Clearly, we can observe and conclude from the table as follows. The explicit formulation of matrix in GMRES leads to the similar runtime to a direct-LU. On the other hand, the direct calculation of matrix-vector product with reusing the pre-factored sparse LUs lead to a significant speedup in MF-GMRES and PAS-GMRES. The accurately identification of the periodic-structured Krylov-subspace in PAS-GMRES further improves the convergence and reduces the runtime. Let's explain the data in the table with details. The direct-LU and the explicit GMRES can only complete the simulation up to 100 states in a reasonable runtime (30 mins), but MF-GMRES and PAS-GMRES can complete all examples in less than 5 seconds. For one cmos-frequency-multiplier, MF-GMRES is 997 times faster (1.52s vs 1516s) and PAS-GMRES is 1804 times faster (0.84s vs 1516s). When compared to the non-structured MF-GMRES, our structured PAS-GMRES on average shows 2 times speedup for runtime, and 3 times less iterations. This is because of the identification and utilization of the structured-Krylov-subspace as explained in Algorithm 2.

We show details of two examples with a further waveform accuracy comparison. The first example is a BJT-mixer with 620 states and a carrier input frequency of 1GHz. Fig. 2 shows two periodic-steady-states at two nodes, generated by the MF-GMRES and PAS-GMRES at the same tolerance, respectively. Clearly, The two converged waveforms are identical to each other. For this example, the PAS-GMRES method converges in 3 iterations using 0.37s and the MF-GMRES method converges in 10 iterations using 0.80s. The next example is a CMOS switch-cap with 654

Table 1: Comparison of shooting update time using different methods

1	2	3	4	5		6	
circuit	eqn #	Direct-LU	Explicit-GMRES	MF-GMRES		PAS-GMRES	
		time(s)	time(s)	iteration #	time(s)	iteration #	time(s)
CMOS-freq-multiplier	100	1516	1409	9	1.52	4	0.84
CMOS-LNA	447	305	65	8	5.33	3	2.44
CMOS-DC-converter	481	NA	NA	10	0.83	3	0.36
BJT-mixer	620	NA	NA	10	0.80	3	0.37
CMOS-switch-cap	654	NA	NA	11	1.71	3	0.73

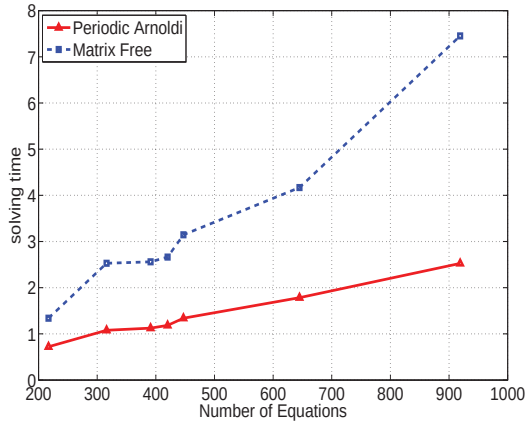

Figure 4: The runtime-scalability comparison for a CMOS-LNA with increased sizes of parasitic. The blue-line is the scale of the matrix-free within the non-structured GMRES, and the red-line is the scale of the periodic Arnoldi within the structured GMRES.

states and a carrier input frequency of 1GHz. Fig. 3 shows two agreed periodic-steady-states at two nodes. For this example, the PAS-GMRES method converges in 3 iterations using 0.73s and the MF-GMRES method converges in 11 iterations using 1.71s. Both of the direct-LU and explicit GMRES can not complete simulations in a reasonable time.

In Fig. 4, we further study the runtime scalability by increasing the size of extracted parasitics. The example used is a CMOS low-noise-amplifier with a carrier input frequency of 1GHz. The circuit complexity is increased from 200 to 900 nodes. The runtime is measured for both of MF-GMRES and PAS-GMRES under the same error tolerance. For a case of 900 nodes, our PAS-GMRES method has a smaller runtime by 3 times than the MF-GMRES method. Clearly, the result shows that the runtime of MF-GMRES grows with a sharper slope than the one of the PAS-GMRES method. Therefore, even bigger speedup is expected for our PAS-GMRES than the MF-GMRES when the circuit size becomes bigger.

7. CONCLUSIONS

The aggressive frequency-scaling in the design of RF/MM-IC requires a robust time-domain shooting algorithm that can deal with the large-scale complexity. In this paper, we have presented a novel periodic Arnoldi shooting (PAS) method to identify and explore the periodic-structure of the underlying Krylov-subspace, which improves the convergence rate of the iterative solver in shooting method. This has resulted in robust yet efficient shooting-Newton iterations, usually only requiring to use the first few of Krylov-subspace bases. Experimental results on a set of industrial RF circuits show that the new PAS method can deliver 1000 times speedup over the direct-LU or explicit GMRES methods. More importantly, the new PAS method reduces the iteration number and runtime by 3 times when compared to the state-of-the-art matrix-free GMRES with the same accuracy. Future work will show the parallel extension.

8. REFERENCES

[1] C. H. Doan *et al.*, "Design considerations for 60 GHz CMOS radios," *IEEE Commun. Mag.*, pp. 132–140, 2004.

[2] A. Hajimiri, "Holistic design in mm-wave silicon ICs," *IEICE Trans. on Electronics*, pp. 817–828, 2008.

[3] B. Razavi, "Design of millimeter-wave CMOS radios: A tutorial," *IEEE Trans. on CAS-I*, pp. 4–16, 2009.

[4] H. Chang and K. Kundert, "Verification of complex analog and RF IC designs," *IEEE Proc.*, pp. 622–639, 2007.

[5] T. J. Aprille and T. N. Trick, "Steady-state analysis of nonlinear circuits with periodic inputs," *IEEE Proc.*, pp. 108–114, 1972.

[6] S. Skelboe, "Computation of the periodic steady-state response of nonlinear networks by extrapolation methods," *IEEE Trans. on Circuits and Systems*, pp. 161–175, 1980.

[7] K. S. Kundert, J. K. White, and A. Sangiovanni-Vincentelli, *Steady-State Methods for Simulating Analog and Microwave Circuits*. Kluwer Academic Publishers, 1990.

[8] R. Telichevesky, K. S. Kundert, and J. K. White, "Efficient steady-state analysis based on matrix-free krylov-subspace methods," in *Proc. ACM/IEEE DAC*, 1995.

[9] K. Mayaram, D. C. Lee, *et al.*, "Computer-aided circuit analysis tools for RFIC simulation: algorithms, features, and limitations," *IEEE Trans. on Circuits and Systems-II*, pp. 274–286, 2000.

[10] O. Nastov, R. Telichevesky, K. Kundert, and J. White, "Fundamentals of fast simulation algorithms for RF circuits," *IEEE Proc.*, pp. 600–621, 2007.

[11] U. M. Ascher and L. R. Petzold, *Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations*. SIAM Publisher, 1998.

[12] Y. Saad and M. H. Schultz, "GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems," *SIAM J. on Sci and Sta. Comp.*, pp. 856–869, 1986.

[13] G. W. Stewart, *Matrix Algorithms, VOL II*. SIAM Publisher, 2001.

[14] O. G. Ernst, "Equivalent iterative methods for p-cyclic matrices," *Numerical Algorithms*, pp. 161–180, 2000.

[15] R. Barrett, M. Berry, *et al.*, *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods, 2nd Edition*. Philadelphia, PA: SIAM, 1994.