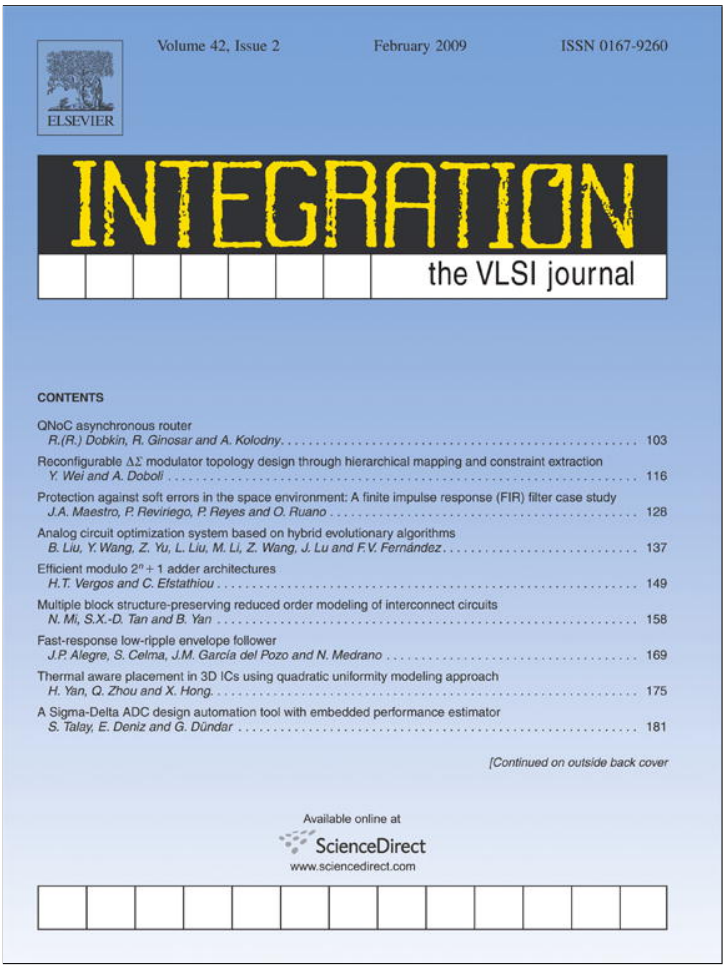

INTEGRATION, the VLSI journal 42 (2009) 193–202

Contents lists available at ScienceDirect

INTEGRATION, the VLSI journal

journal homepage: www.elsevier.com/locate/vlsi

Hierarchical Krylov subspace based reduction of large interconnects ☆

Duo Li [a], Sheldon X.-D. Tan [a,*], Lifeng Wu [b]

[a] Department of Electrical Engineering, University of California, Riverside, CA 92521, USA
[b] Cadence Design Systems Inc., San Jose, CA 95134, USA

ARTICLE INFO

Article history:
Received 6 January 2008
Received in revised form
26 June 2008
Accepted 26 June 2008

Keywords:
Model order reduction
Krylov subspace
Interconnect

ABSTRACT

In this paper, we propose a new model order reduction approach for large interconnect circuits using hierarchical decomposition and the Krylov subspace projection-based model order reduction methods. The new approach, called *hiePrimor*, first partitions a large interconnect circuit into a number of smaller subcircuits and then performs the projection-based model order reduction on each of subcircuits in isolation and on the top-level circuit thereafter. The new approach is very amenable for exploiting the multi-core based parallel computing platforms to significantly speed up the reduction process. Theoretically we show that hiePrimor can deliver the same accuracy as the flat reduction method given the same reduction order and it can also preserve the passivity of the reduced models as well. We also show that partitioning has large impacts on the performance of hierarchical reduction and the minimum-span objective should be required to attain the best performance for hierarchical reduction. The proposed method is suitable for reducing large global interconnects like coupled bus, transmission lines, large clock nets in the post-layout stage. Experimental results demonstrate that hiePrimor can be significantly faster and more scalable than the flat projection methods like PRIMA and be order of magnitude faster than PRIMA with parallel computing without loss of accuracy. Interconnect circuits with up to 4 million nodes can be analyzed in a few minutes even in Matlab by the new method.

1. Introduction

Compact modeling of passive RLC interconnect networks has been a research-intensive area in the past decade owing to increasing delays and signal integrity effects and increasing design complexity in today's nanometer VLSI designs. Reducing the parasitic interconnect circuits by approximate compact models can significantly speed up the simulation and verification process in nanometer VLSI designs. As the technology moves to 45 nm, the massive extracted post-layout circuits will make the reduction imperative before any meaningful simulations and verifications. Hence the reduction algorithm must be able to scale to attack very large circuit sizes in the current and future technologies.

Reduction algorithms based on subspace projection have been proved to be very effective in the past [2–10]. Those methods typically project the original circuit into the dimensioned-reduced Krylov subspace to reduce the model order. Krylov subspace methods can lead to a localized moment matching link between the original model and the reduced one. It was introduced to the interconnect reduction by the Pade via Lanczos (PVL) [2] method, as it can mitigate the numerical problems in the explicit moment matching methods like asymptotic waveform evaluation (AWE) algorithm [11]. Thereafter, some similar approaches such as Arnoldi transformation method [3] was also proposed. Later, the congruence transformation method [4] and PRIMA [5] were further proposed, which produce passive models. Recently a general structure-preserving reduction method [8] was proposed to keep thematrix structure and at the same time improve the model accuracy by exploiting the symmetry property of circuit matrices. To mitigate the low efficiency problem of reduction of interconnect circuits with large terminals, the combined terminal and model order reduction techniques by singular value decomposition been have proposed [7] and the method was improved in [9]. Another efficient approach for reducing port-intensive circuits is by means of *s*-domain hierarchical reduction, in which the target transfer function are computed and truncated in a hierarchical way [6].

At the same time, many other approaches have also been proposed, such as balanced truncation based reduction methods

☆ Some preliminary results of this paper were presented at *IEEE Proceedings of the Asia South Pacific Design Automation Conference* (ASP-DAC'08) [1], Seoul, South Korea, January 2008. This work is supported in part by NSF Award under Grant no. CCF-0448534, UC Micro Program #06-252 and #07-105 via Cadence Design Systems Inc.

* Corresponding author.
E-mail address: stan@ee.ucr.edu (S.X.-D. Tan).

0167-9260/$ - see front matter © 2008 Elsevier B.V. All rights reserved.
doi:10.1016/j.vlsi.2008.06.004

[12–14], local node reduction methods [15,16] and general node reduction method [17,18]. But Krylov subspace based-reduction method remains a viable approach for many practical interconnect reduction problems owning to its high efficiency. Existing projection-based reduction methods, however, lack a general way to exploit the parallel computing capabilities, which become more popular with emerging multi-core computing architectures.

But in this paper, we investigate the parallelism within the reduction operations in one expansion point for one large interconnect circuit. Grimme has explored the parallel computation for multi-point Krylov based reduction where each Krylov subspace from each expansion point can be computed in parallel [19]. Hierarchical reduction of interconnects have also been studied from different perspectives in the past. In HiPRIME algorithm [20], hierarchical reduction has been extended in the extended Krylov subspace method (EKS) to compute the responses of on-chip power grid networks. The HiPRIME method reduces both system and input signals at the same time in a hierarchical way, but it does not produce a reduced model for general use. In the RecMOR method [21], Feldmann and Liu applied the combined terminal and model order reduction on the subcircuits based on the observation that partitioning may lead to many circuits with many new terminals, which will affect the efficiency for projection-based reduction methods. However, terminal reduction in general still remains a difficult problem and may not be effective for many practical problems [12,22].

In this paper, we propose a new hierarchical Krylov subspace based reduction method. The new method combines the partitioning strategy and the Krylov subspace method to speed up the reduction process. The proposed method is more suitable for reducing many large global interconnects like coupled bus, transmission lines and large clock nets where the number of ports are generally not significant. The new method is a very general hierarchical model order reduction technique and it works for general parasitic interconnect circuits modeled as RLC circuits.

The new method, called *hiePrimor*, first partitions a large RLC circuit into two or more levels and then perform the projection-based reduction on subcircuits in a bottom-up way. Our contributions are as follows: (1) theoretically we show that if kth order block moment order is preserved in all the reduction processes for all subcircuits and top-level circuit, first k block moments will be preserved in the final reduced models; (2) we prove that the new hierarchical reduction method also preserves the passivity of the reduced models for interconnects at all the hierarchical levels; (3) we show that the proposed method not only can exploit parallel computing to speed up the reduction process, but also can significantly improve the analysis capacity by partitioning strategy; (4) we study the impacts of partitioning on the reduction efficiency and show that partitioning is critical for the hierarchical reduction process and minimum-span (min-span) or minimum-cut (min-cut) objective should be attained for best reduction performance. We apply the existing hMETIS partitioning tools [23] to perform the min-cut partitioning.

The proposed method, for the first time, exploits the partitioning-based reduction strategy, which enable the parallel computing and more scalability for handle very large parasitic interconnect circuits. Experimental results show that the proposed method can lead to significant speed up over the flat projection-based method like PRIMA and order of magnitudes speed up over PRIMA if parallel computing is used. Interconnect circuits with millions of nodes can be analyzed by hiePrimor in a desktop PC using Matlab in a few minutes.

The rest of the paper is organized as follows. We review the Krylov subspace based model order reduction in Section 2. Then we present the main idea of the new method using an illustrative example in Section 3. We show the moment matching property for the new method in Section 4. In Section 5 we prove that the new method preserves the passivity of the reduced models. We discuss the partitioning impacts and schemes in Section 6. Section 7 presents the computational cost of the proposed method. We present the experimental results in Section 8, and conclude the paper in Section 9.

2. Review of subspace projection-based MOR methods

In this section, we review the Krylov subspace projection-based methods, which are also used for the new hierarchical projection MOR method.

Without loss of generality, a linear m-port RLC circuit can be expressed as

$$\begin{aligned} C\dot{\mathbf{x}}_n &= -G\mathbf{x}_n + B\mathbf{u}_m \\ \mathbf{i}_m &= L^{\mathrm{T}}\mathbf{x}_n \end{aligned} \tag{1}$$

where $\mathbf{x}_n$ is the vector of state variables and n is the number of state variables, m is the number of independence sources specified as ports. C, G are storage element and conductance matrices, respectively. B and L are position matrices for input the output ports, respectively.

Define $A=-G^{-1}C$, $A\in\Re^{n\times n}$ and $R=G^{-1}B$, $R=[r_0,r_1,\ldots,r_m]$, $R\in\Re^{n\times m}$. The transfer function matrix after Laplace transformation is $H(s)=L^{\mathrm{T}}(G+sC)^{-1}B=L^{\mathrm{T}}(I_n-sA)^{-1}R$ where I_n is the $n\times n$ identity matrix. The block moments of $H(s)$ are defined as the coefficients of Taylor expansion of $H(s)$ around $s=0$:

$$H(s)=M_0+M_1s+M_2s^2+\ldots \tag{2}$$

where $M_i\in\Re^{m\times m}$ and can be computed as $M_i=L^{\mathrm{T}}A^iR$. In the sequel, we use m_i to denote the terminal count for subcircuit i.

The idea of model order reduction is to find a compact system of a much smaller size than the original system. The Krylov subspace based method accomplishes this by projecting the original system on a special subspace which spans the same space as the block moments of the original system. Specifically, the block Krylov subspace is defined as

$$\begin{aligned} Kr(A,R,q) = colsp[&R, AR, A^2R,\ldots,A^{k-1}R, \\ &A^kr_0, A^kr_1,\ldots,A^kr_l] \end{aligned} \tag{3}$$

$$k=\lfloor q/m\rfloor,\quad l=q-km \tag{4}$$

For simplicity of expression, we assume $q=m\times k$ in the following and k is the order of block moments used in the Krylov subspace. i.e. k order block moments will be matched if Krylov subspace $Kr(A,R,mk)$ is used. Then, projection MOR method tries to find orthogonal matrix $X\in\Re^{n\times q}$ such that $colsp(X)=Kr(A,R,q)$. With

$$\begin{aligned} \tilde{C}&=X^{\mathrm{T}}CX,\quad \tilde{G}=X^{\mathrm{T}}GX \\ \tilde{B}&=X^{\mathrm{T}}B,\quad \tilde{L}=X^{\mathrm{T}}L \end{aligned}$$

the reduced system of size q is found as

$$\begin{aligned} \tilde{C}\dot{\tilde{\mathbf{x}}}_n &= -\tilde{G}\tilde{\mathbf{x}}_n + \tilde{B}\mathbf{u}_m \\ \mathbf{i}_m &= \tilde{L}^{\mathrm{T}}\tilde{\mathbf{x}}_n \end{aligned} \tag{5}$$

The reduced transfer function become $\tilde{Y}(s)=\tilde{L}^{\mathrm{T}}(\tilde{G}+s\tilde{C})^{-1}\tilde{B}$. An important result for projection-based MOR methods is that the

reduced system approximates the original systems in terms of moment matching: if $Kr(A,R,q) \subseteq span(X)$, then the reduced transfer function $\tilde{Y}(s)$ and the original transfer function $H(s)$ matches the first k block moments where $k = q/m$. Also when $L = B$, the reduction process preserves passivity.

3. Hierarchical projection MOR method: hiePrimor

3.1. A walkthrough example

We introduce our method by using an illustrative RC example circuit shown in Fig. 1. This circuit has been partitioned into three parts, the two subcircuits I and II and the top part, which connects the two subcircuits. The two subcircuits are connected via the top-level circuit only.

As a result, we have the partitioned MNA equations as shown in (6), where we partition the matrix into three parts and the input sources into two parts as input sources only appear in partition I and the top-level partition.

$$\left[\begin{array}{cccc|cc|cc} G_1 & -G_1 & -1 & & 0 & 0 & 0 & 0 \\ -G_1 & G_1+G_2 & 0 & & 0 & 0 & -G_2 & 0 \\ 1 & 0 & 0 & & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & & (G_3+G_4) & -G_4 & -G_3 & 0 \\ 0 & 0 & 0 & & -G_4 & G_4 & 0 & 0 \\ \hline 0 & -G_2 & 0 & & -G_3 & 0 & (G_2+G_3) & -1 \\ 0 & 0 & 0 & & 0 & 0 & 1 & 0 \end{array}\right] \left[\begin{array}{c} v_1 \\ v_2 \\ i_{u_1} \\ \hline v_3 \\ v_4 \\ \hline v_5 \\ i_{u_2} \end{array}\right] +$$

$$\left[\begin{array}{ccc|cc|cc} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & C_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & C_3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & C_4 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & C_2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{array}\right] \left[\begin{array}{c} \dot{v}_1 \\ \dot{v}_2 \\ \dot{i}_{u1} \\ \dot{v}_3 \\ \dot{v}_4 \\ \dot{v}_5 \\ \dot{i}_{u2} \end{array}\right] = \left[\begin{array}{cc} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ \hline 0 & 0 \\ 0 & 0 \\ \hline 0 & 0 \\ 0 & 1 \end{array}\right] \left[\begin{array}{c} u_1 \\ \hline u_2 \end{array}\right]$$

$$\left[\begin{array}{c} i_{u_1} \\ \hline i_{u_2} \end{array}\right] = \left[\begin{array}{ccc|cc|cc} 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{array}\right] x_n$$

$$x_n^T = [v_1\ v_2\ i_{u_1}\ v_3\ v_4\ v_5\ i_{u_2}]$$

$$x_n^{\mathrm{T}} = [v_1 v_2 i_{u_1} v_3 v_4 v_5 i_{u_2}] \tag{6}$$

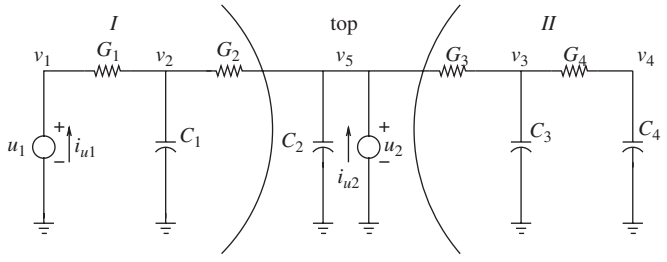

Fig. 1. A partitioned RC circuit.

In general, we can write a w-way partitioned RLC circuit into the following general form:

$$\left[\begin{array}{cccc} G_1 & 0 & \dots & G_{1t}^{\mathrm{T}} \\ 0 & G_2 & \dots & G_{2t}^{\mathrm{T}} \\ \vdots & \vdots & \dots & \vdots \\ G_{1t} & G_{2t} & \dots & G_{tt} \end{array}\right] \left[\begin{array}{c} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \vdots \\ \mathbf{x}_t \end{array}\right] + \left[\begin{array}{cccc} C_1 & 0 & \dots & 0 \\ 0 & C_2 & \dots & 0 \\ \vdots & \vdots & \dots & \vdots \\ 0 & 0 & \dots & C_{tt} \end{array}\right] \left[\begin{array}{c} \dot{\mathbf{x}}_1 \\ \dot{\mathbf{x}}_2 \\ \vdots \\ \dot{\mathbf{x}}_t \end{array}\right]$$

$$= \left[\begin{array}{cccc} B_1 & 0 & \dots & 0 \\ 0 & B_2 & \dots & 0 \\ \vdots & \vdots & \dots & \vdots \\ 0 & 0 & \dots & B_{tt} \end{array}\right] \left[\begin{array}{c} \mathbf{u}_1 \\ \mathbf{u}_2 \\ \vdots \\ \mathbf{u}_t \end{array}\right] \tag{7}$$

where the $\mathbf{x}_i$ is the internal variable vector for partition i and $\mathbf{u}_i$ is the external input vector for partition i. $\mathbf{x}_t$ and $\mathbf{u}_t$ are the variables and external input vectors of the top-level circuit. If there are no external inputs for partition i, then the corresponding columns in the position matrix can be removed as shown in (6).

3.2. The hiePrimor algorithm

For a general RLC circuit, we can rewrite (7) as

$$G\mathbf{x} + C\dot{\mathbf{x}} = B\mathbf{u} \tag{8}$$

The idea of hierarchical projection-based reduction is to first perform the reduction using projection MOR method for each subcircuit assuming that the subcircuits are disconnected from the rest of the circuit. After the subcircuits are reduced, we perform the reduction on their parent circuits of the subcircuits until we reach to the top-level circuit. The benefit of doing this is that we can reduce the computation complexity by performing the reduction on the subcircuits and intermediate circuits and parallelism can be exploited to speed up the reduction process as subcircuits in one hierarchical level can be reduced independently.

To illustrate this idea, we still use the example in Fig. 1. To reduce the subcircuit I, we have the following subcircuit matrix:

$$\left[\begin{array}{ccc} G_1 & -G_1 & -1 \\ -G_1 & G_1+G_2 & 0 \\ 1 & 0 & 0 \end{array}\right] \left[\begin{array}{c} v_1 \\ v_2 \\ i_{u_1} \end{array}\right] + \left[\begin{array}{ccc} 0 & 0 & 0 \\ 0 & C_1 & 0 \\ 0 & 0 & 0 \end{array}\right] \left[\begin{array}{c} \dot{v}_1 \\ \dot{v}_2 \\ \dot{i}_{u_1} \end{array}\right]$$

$$= \left[\begin{array}{cc} 0 & 0 \\ 0 & 1 \\ 1 & 0 \end{array}\right] \left[\begin{array}{c} u_1 \\ i_2 \end{array}\right] \tag{9}$$

where i_2 is the current source attached to node 2, which becomes a terminal node now. The added current source is just for reduction propose. Note that the position matrix $B_1 = [0\ 0\ 1]^{\mathrm{T}}$ for this subcircuit has been changed to

$$B_1' = \left[\begin{array}{cc} 0 & 0 \\ 0 & 1 \\ 1 & 0 \end{array}\right] \tag{10}$$

This modification reflects the fact that the subcircuit I now has two terminal nodes: node 1 and node 2. Notice that all the internal nodes, which are inside a subcircuit and are connected to boundary node at the upper level via a device branch, become the terminal nodes of the subcircuits for the reduction propose (as the case of node 2). If a subcircuit does not have any external input (such as the subcircuit II), all the nodes incident on the boundary nodes will become the terminal nodes for the reduction of the subcircuit. As projection-based MOR method becomes less effective for increasing terminal counts, we should try to

minimize the terminal counts of subcircuit. Therefore, the hierarchical reduction requires the min-span[1] partitioning of the circuit. In this way, we can achieve better reduction performance. We will discuss the partitioning issue in Section 6.

After the projection matrix V_1 is computed using (9), where V_1 spans the kth order block Krylov subspace, i.e. $V_1 \subseteq Kr(G_1^{-1}B_1, G_1^{-1}C_1, km_1)$, where m_1 is the terminal count of subcircuit 1, we can perform the reduction. But now we need to look at the subcircuit in the context of the whole circuit. From (7), for the subcircuit 1, we have

$$G_1\mathbf{x}_1 + C_1\dot{\mathbf{x}}_1 + G_{1t}^{\mathrm{T}}\mathbf{x}_t = B_1\mathbf{u}_1 \tag{11}$$

After the reduction, we have

$$\tilde{G}_1\mathbf{z}_1 + \tilde{C}_1\dot{\mathbf{z}}_1 + \tilde{G}_{1t}^{\mathrm{T}}\mathbf{x}_t = \tilde{B}_1\mathbf{u}_1 \tag{12}$$

where $\mathbf{x} = V_1\mathbf{z}$, $\tilde{G}_1 = V_1^{\mathrm{T}}G_1V_1$, $\tilde{C}_1 = V_1^{\mathrm{T}}C_1V_1$, $\tilde{B}_1 = V_1^{\mathrm{T}}B_1$, $\tilde{G}_{1t} = V_1G_{1t}$. We remark that we use original B_1 here instead of B_1'. Since the $colsp(B_1) \subseteq colsp(B_1')$, we can use subspace defined in V_1 to perform the reduction.

We repeat the reduction process on all the subcircuits as mentioned above until we end up with the following order reduced system at the top level:

$$\begin{bmatrix} \tilde{G}_1 & 0 & \dots & \tilde{G}_{1t}{}^{\mathrm{T}} \\ 0 & \tilde{G}_2 & \dots & \tilde{G}_{2t}{}^{\mathrm{T}} \\ \vdots & \vdots & \dots & \vdots \\ \tilde{G}_{1t} & \tilde{G}_{2t} & \dots & G_{tt} \end{bmatrix} \begin{bmatrix} \mathbf{z}_1 \\ \mathbf{z}_2 \\ \vdots \\ \mathbf{x}_t \end{bmatrix} + \begin{bmatrix} \tilde{C}_1 & 0 & \dots & 0 \\ 0 & \tilde{C}_2 & \dots & 0 \\ \vdots & \vdots & \dots & \vdots \\ 0 & 0 & \dots & C_{tt} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{z}}_1 \\ \dot{\mathbf{z}}_2 \\ \vdots \\ \dot{\mathbf{x}}_t \end{bmatrix} = \begin{bmatrix} \tilde{B}_1 & 0 & \dots & 0 \\ 0 & \tilde{B}_2 & \dots & 0 \\ \vdots & \vdots & \dots & \vdots \\ 0 & 0 & \dots & B_{tt} \end{bmatrix} \begin{bmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \\ \vdots \\ \mathbf{u}_t \end{bmatrix} \tag{13}$$

In this paper, we only present the results for two-level reduction as shown in (7). But the proposed method can be trivially extended to more hierarchical levels. We can also rewrite (13) as

$$G_r\tilde{\mathbf{x}}_r + C_r\dot{\tilde{\mathbf{x}}}_r = B_r\mathbf{u}_r \tag{14}$$

With (13), we can continue the reduction by performing the reduction at the top-level circuit using the projection-based reduction method again. Finally, we have the reduced model

$$\tilde{G}\tilde{\mathbf{x}} + \tilde{C}\dot{\tilde{\mathbf{x}}} = \tilde{B}\mathbf{u} \tag{15}$$

where $\tilde{G} = V_t^{\mathrm{T}}G_rV_t$, $\tilde{C} = V_t^{\mathrm{T}}C_rV_t$, $\tilde{B} = V_t^{\mathrm{T}}B_r$. G_r, C_r and B_r are the circuit matrices in (14) and $V_t \subseteq Kr(G_r^{-1}B_r, G_R^{-1}C_r, q_t)$, where $q_t = km_t$ and m_t is the terminal count at the top-level circuit.

3.3. The algorithm flow for hiePrimor

In this subsection, we summarize the algorithm flow of the hiePrimor method shown in Fig. 3.

Algorithm 1. Hierarchical Krylov subspace projection-based model order reduction method (hiePrimor)

Input: Circuit matrices G, C, B, reduced order q, partition number w
Output: Reduced matrices $\hat{G}$, $\hat{C}$, $\hat{B}$
1. Partition original large circuit into w small subcircuits using hMETIS.
2. Form original circuit matrices as in (7).
3. For each subcircuit i, find sub-level projection matrix V_i using Krylov subspace method.
4. Reduce subcircuit matrices
$\tilde{G}_i = V_i^{\mathrm{T}}G_iV_i$, $\tilde{C}_i = V_i^{\mathrm{T}}C_iV_i$, $\tilde{B}_i = V_i^{\mathrm{T}}B_i$, $\tilde{G}_{it} = V_iG_{it}$.
5. Form top-level circuit matrices as in (13).
6. Compute top-level projection matrix V_t using Krylov subspace method.
7. $\hat{G} = V_t^{\mathrm{T}}\tilde{G}V_t$,$\hat{C} = V_t^{\mathrm{T}}\tilde{C}V_t$, $\hat{B} = V_t^{\mathrm{T}}\tilde{B}$.
8. End.

4. Moment matching connection

In this section, we analyze the moment matching property of the proposed method. We show that if the kth order block moment is preserved/matched in the reductions for all the subcircuits and for the top-level circuit as well, the final reduced model preserves the first k block moments of the original system.

Assume that we have an interconnected circuit system with the transfer function $H(s)$, which consists of n subcircuits that connects together. Assume that we denote subcircuit i as (G_i, C_i, B_i) and we perform the projection-based model order reduction on the subcircuit i only

$$(\tilde{G}_i, \tilde{C}_i, \tilde{B}_i) = (V_i^{\mathrm{T}}G_iV_i, V_i^{\mathrm{T}}C_iV_i, V_i^{\mathrm{T}}B_i) \tag{16}$$

and keep all the other system unchanged. We generate the projection matrix V_i such that

$$V_i \subseteq Kr(A_i, R_i, q_i) \tag{17}$$

where $A_i = -G_i^{-1}C_i$, $R_i = G_i^{-1}B_i$ and $q_i = km_i$. Then we have the following result:

Lemma 1. *The resulting interconnected circuit system transfer $\tilde{H}_1(s)$, which consists of the order reduced subcircuit $(\tilde{G}_i, \tilde{C}_i, \tilde{B}_i)$ with rest of subcircuits unchanged, matches the first k block moments of $H(s)$.*

The detailed proof of this lemma can be found in [24]. Here we give an intuitive example to explain the lemma. For instance, we have two connected subsystems A and B with two transfer function $H_A(s)$ and $H_B(s)$, where the outputs of A drive the inputs of B. So the whole system transfer function is $H(s) = H_A(s)H_B(s)$. If we replace $H_A(s)$ with $\hat{H}_A(s)$, which is accurate to qth order of $H_A(s)$. It can be easily seen that $\hat{H}(s) = \hat{H}_A(s)H_B(s)$ will be accurate to the qth order of $H(s)$ if we write both $\hat{H}_A(s)$ and $H_B(s)$ in the moment (Taylor's series) form.

For the interconnected circuit system $H(s)$, all of its subcircuits are reduced by the projection-based MOR method such that

$$(\tilde{G}_i, \tilde{C}_i, \tilde{B}_i) = (V_i^{\mathrm{T}}G_iV_i, V_i^{\mathrm{T}}C_iV_i, V_i^{\mathrm{T}}B_i), \quad i = 1, \dots, w \tag{18}$$

such that $V_i \subseteq Kr(A_i, R_i, q_i)$, $q_i = km_i$ for all the subcircuits. Based on Lemma 1, we can easily obtain the following result:

Corollary 1. *The resulting interconnected circuit system transfer $\tilde{H}_2(s)$, which consists of the order reduced subcircuit, $(\tilde{G}_i, \tilde{C}_i, \tilde{B}_i), i = 1, \dots, w$, for all subcircuits, matches the first q block moments of $H(s)$.*

The proof of Corollary 1 can be obtained when we apply Lemma 1 w times to the interconnected circuit system $H(s)$ such that we reduce one subcircuit at a time.

Now we are ready to present the main result regarding the proposed hierarchical model order reduction method, hiePrimor.

Theorem 1. *Given a partitioned RLC circuit defined in (7) with transfer function H(s), if we perform the projection based reduction on all the subcircuits and then on the top-level circuit such that kth*

[1] *span* of cut net is the number of internal nodes that a cut net connects from all the partitions.

order block moment is preserved in all the reduction processes, the transfer function $\tilde{H}(s)$ of the reduced system in (15) will match the first k block moments of $H(s)$.

The proof of the theory is obvious in light of Corollary 1 and the fact that top-level reduction on (13) also preserves the kth order block moment. Theorem 1 also indicates that for the hierarchical reduction process, we should always use the same block moment order for all the reduction processes. For the same reduction order k, different subcircuit may have different reduced model sizes as the size of the reduced model is km_i, where m_i is the terminal count of the subcircuit i.

In summary, the proposed hierarchical projection-based reduction method, hiePrimor, will have the same accuracy as the flat-projection based method if both methods use the same block moment order.

5. Analysis of passivity preservation

In this section, we show that the proposed hiePrimor method preserves the passivity of the reduced models.

It is obvious that the passivity is preserved during the reduction of the leaf level subcircuits as they are reduced by normal projection-based model order reduction [25]. The only thing left is to show that reduction on the intermediate or top level circuits indicated by (13) still preserves the passivity.

Theorem 2. *hiePrimor preserves the passivity of the order reduced models at intermediate and top-level circuits.*

Proof. For a passive circuit, its transfer function must be positive real. A network circuit with a matrix transfer function $H(s)$ is said to be positive real iff

(1) $H(s)$ is analytic for $\mathrm{Re}(s)>0$
(2) $H^*(s)=H(s^*)$ for $\mathrm{Re}(s)>0$
(3) $H(s)+H(s^*)^T\geqslant 0$ for $\mathrm{Re}(s)>0$

where $\geqslant$ means nonnegative definite (or positive semidefinite). Condition (1) and (2) are typically always satisfied for a RLC circuit as it does not have unstable poles and the system has real response. And condition (3) needs to be proved.

Let us denote the transfer function of the final reduced model in (15) as $H(s)=\tilde{B}^{\mathrm{T}}(\tilde{G}+s\tilde{C})^{-1}\tilde{B}$. Condition (3) is equivalent to the requirement that $\tilde{G}+s\tilde{C}$ is positive real

$$W(s)=\tilde{G}+s\tilde{C}\geqslant 0 \tag{19}$$

Let $s=\sigma+j\omega$ and $\sigma>0$, then $W(s)+W(s^*)^T$ becomes

$$K_h(s)=\tilde{G}+\tilde{G}^{\mathrm{T}}+2\sigma\tilde{C} \tag{20}$$

We now need to prove that $K_h(s)$ is positive real.

As we know $\tilde{G}=V_t^{\mathrm{T}}G_rV_t$ and $\tilde{C}=V_t^{\mathrm{T}}C_rV_t$. If we define $V_r=diag(V_1,V_2,\ldots,V_w,I)$, where V_i is the projection matrix for subcircuit i. We can rewrite (14) as

$$\begin{aligned} &V_r^{\mathrm{T}}GV_r\mathbf{x}_r+V_r^{\mathrm{T}}CV_r\dot{\mathbf{x}}_r=V_r^{\mathrm{T}}B\mathbf{u}_r\\ &V_t^{\mathrm{T}}V_r^{\mathrm{T}}GV_rV_t\tilde{\mathbf{x}}+V_t^{\mathrm{T}}V_r^{\mathrm{T}}CV_rV_t\dot{\tilde{\mathbf{x}}}=V_t^{\mathrm{T}}V_r^{\mathrm{T}}B\mathbf{u}\end{aligned} \tag{21}$$

As a result, $K_h(s)$ becomes

$$K_h(s)=V_t^{\mathrm{T}}V_r^{\mathrm{T}}(G+G^{\mathrm{T}}+2\sigma C)V_rV_t \tag{22}$$

where G and C are the circuit matrices written in the partitioned form as defined in (7). Therefore we need to prove that $G+G^{\mathrm{T}}+2\sigma C$ is positive real as $K_h(s)$ is obtained by congruence transformation $W^{\mathrm{T}}(G+G^{\mathrm{T}}+2\sigma C)W$, where $W=V_tV_r$.

We notice that the partitioned G matrix is no longer in the so-called passive form using MNA formation from a RLC circuit as shown in (6). The passive form should be in the following form [5]:

$$G_p=\begin{bmatrix} N & E^{\mathrm{T}}\\ -E & 0\end{bmatrix} \tag{23}$$

where $G_p\geqslant 0$ is nonnegative definite as $N\geqslant 0$ is nonnegative definite. Notice that each subcircuit conductance matrix G_i is in the passive form as we need to perform passive reduction using projection methods for each subcircuit. In this case, one can easily prove that the passive form G_p can be obtained by permuting the same set of rows and columns simultaneously. Such a permutation can be represented by the permutation matrix P:

$$G=P^{\mathrm{T}}G_pP \tag{24}$$

This is true for C matrix. Finally, we have

$$K_h(s)=V_t^{\mathrm{T}}V_r^{\mathrm{T}}P^{\mathrm{T}}(G_p+G_p^{\mathrm{T}}+2\sigma C_p)PV_rV_t \tag{25}$$

Since $G_p+G_p^{\mathrm{T}}$ and C_p are nonnegative definite for RLC circuits, $K_h(s)$ is positive real. □

6. Circuit partitioning

Partitioning plays an important role in the performance of the proposed reduction method. The reason is that the nodes that are inside a subcircuit and are incident on the boundary nodes at the top level will become the terminal nodes for subcircuits. The sizes of the reduced models grow linearly with the terminal number of the original circuits in the projection-based reduction framework as the size of the reduced model is km_i, where k is the block moment order and m_i is terminal count for subcircuit i. To have smaller sizes of the reduced matrices (thus smaller nonzero elements in the matrices), which will be stamped into the higher-level circuit matrix for further reduction, we need to reduce terminal count of subcircuits as much as possible. This calls for the min-span or min-cut partitioning to achieve this.

Also the size of subcircuits cannot be too small compared with the number of terminals to have meaningful reduction on subcircuits. As a result, the proposed method is more suitable for very large RLC networks like bus, coupled transmission lines and clock nets with loosely coupled subcircuits.

After partitioning, the subcircuit terminals generated by partitioning will be driven by current sources in general, which requires the subcircuit to have DC path for all the nodes. If this is not the case, we have to introduce voltage sources at the terminal for the reduction purpose (to make the subcircuit G_i non-singular). This will add more interface terminals to the original subcircuits. As a result, we should minimize the capacitive cut, which can lead to non-DC path nodes. But the proposed method does not have any restrictions on types of boundary nodes.

To meet the partitioning requirement, we apply hMETIS partition tool suite [23], which employs the hierarchical partitioning strategy and is the best min-cut partitioning tools available. Specifically, we abstract a circuit netlist into a hypergraph, where components (such as resistors, capacitors, inductors, etc.) are considered as vertices in abstracted hypergraph, and nodes in circuit netlist are considered as hyperedges. Then we use hMETIS partitioning suites of the hypergraph partitioning to partition the original large circuit into several small subcircuits.

hMETIS can balance the sizes of each partition automatically without any change to its cost function. In the experiments, we set two-level partitioning: one top-level circuit and many second-level subcircuits. hMETIS tool suite is very efficient for partitioning very large networks. With hMETIS, the hiePrimor is able to

reduce very large interconnect circuits with millions of nodes in a PC using Matlab.

For very densely coupled circuits, the proposed method can still be applied. First, for the capacitively coupled circuits, it is well known that the coupling is more localized, which means the coupling can be further reduced without loss of much accuracy. For inductive coupling, which has long-range effects owning to partial inductance formation [26], many methods have been proposed to reduce the coupling using various window-based truncation techniques [27–29] before the hierarchical reduction.

7. Computational complexity analysis

The major steps in the hiePrimor consists of the partitioning, the reductions of the each subcircuit and the reduction of the top-level circuit.

For the partitioning step, the time complexity is about $O(n)$, n is the size of the circuit. Assume that after the balanced partitioning, the size of each subcircuit is about l.

For the reduction using the Krylov subspace method, we need to solve the subcircuit. Typically solving a $l \times l$ linear matrix takes $O(l^{\alpha})$ (typically, $1 \leqslant \alpha \leqslant 1.2$ for sparse circuits), and matrix factorizations take $O(l^{\beta})$ (typically, $1.1 \leqslant \beta \leqslant 1.5$ for sparse circuits). For each subcircuit, assuming that we need to compute k block moments, the computing cost then is

$$O(kl^{\alpha} + l^{\beta} + lk^2) \tag{26}$$

where the last term is the cost of QR operation in the reduction process [30].

For the top-level circuit, whose size is p, the reduction cost (with kth moment matching) is

$$O(kp^{\alpha} + p^{\beta} + pk^2). \tag{27}$$

So the total computational cost is

$$O(kp^{\alpha} + p^{\beta} + pk^2 + w(kl^{\alpha} + l^{\beta} + lk^2)) \tag{28}$$

where w is the number of partitions. If the parallel computing is performed, the best time complexity can be

$$O(kp^{\alpha} + p^{\beta} + pk^2 + kl^{\alpha} + l^{\beta} + lk^2) \tag{29}$$

if all the subcircuit reduction can be performed in parallel.

8. Experimental results

In this section, we report the experiment results of hiePrimor on some interconnected circuits. We compare it with PRIMA [5] with and without parallel computing settings. We implement the hiePrimor method and PRIMA using Matlab 7.0 and Python. Sparse matrix structures are used in Matlab. Python is used for a parser converting Spice format netlist into Matlab format.

Our test circuits are created based on a bus circuit structure, where each circuit has capacitively coupled bus lines with different lengths and each of them is modeled as a RC ladder-like circuit. To partition the testing circuits in Spice format, we transform the netlists into the ones that hMETIS can read and then partition the circuits into several small spice-formatted sub-circuits of equal size with the min-cut objective.

We first show that hiePrimor and PRIMA give almost the same accuracy for the given block moment order k (our claim in Section 4). We set the reduction order q as $q = n \times k$, where n is the number of ports. Fig. 2 (Ckt1, 25 K) and Fig. 4 (Ckt7, 1 M) show the frequency responses of $Y(1,1)$ and $Y(1,2)$ from the reduced models by hiePrimor and PRIMA. Figs. 3 and 5 show the differences between hiePrimor and PRIMA. In all the test circuits, the accuracy of PRIMA and the hiePrimor are the almost the same numerically, although their results may be a little bit different from the exact one (Fig. 5).

If we increase the value of k, we can obtain the more accurate models. Figs. 6 and 7 show the comparison results for Ckt1 and Ckt7 for $k = 8$. We can see that the results are much better than the previous cases when $k = 4$. Notice that the results from hiePrimor and PRIMA are still almost the same again and their sizes after reduction are the same too.

Next, we compare hiePrimor with PRIMA in a single CPU setting in terms of reduction times. Table 1 shows the circuit statistics and comparison results of PRIMA and hiePrimor. #Node is the number of nodes, #Sub is the number of subcircuits, #Ports is number of ports (terminals) of the circuit and '-' means out of memory or could not end in a reasonable time. Note that Ckt1–Ckt8 are run on an Intel Xeon 3.0 GHz dual CPU workstation with 2 GB memory; Ckt9 and Ckt10 are run on an workstation with an Intel Xeon Quad-Core CPU (3.0 GHz and 16 GB memory).

We set the reduction (block moment) order to 4 ($k = 4$) in all the test circuits so that each circuit has the same reduced order (size) after reduction. It may be not accurate enough for $k = 4$ in all the circuits. But given that fact that hiePrimor gives almost the same accuracy as PRIMA, $k = 4$ is sufficient for us to compare the

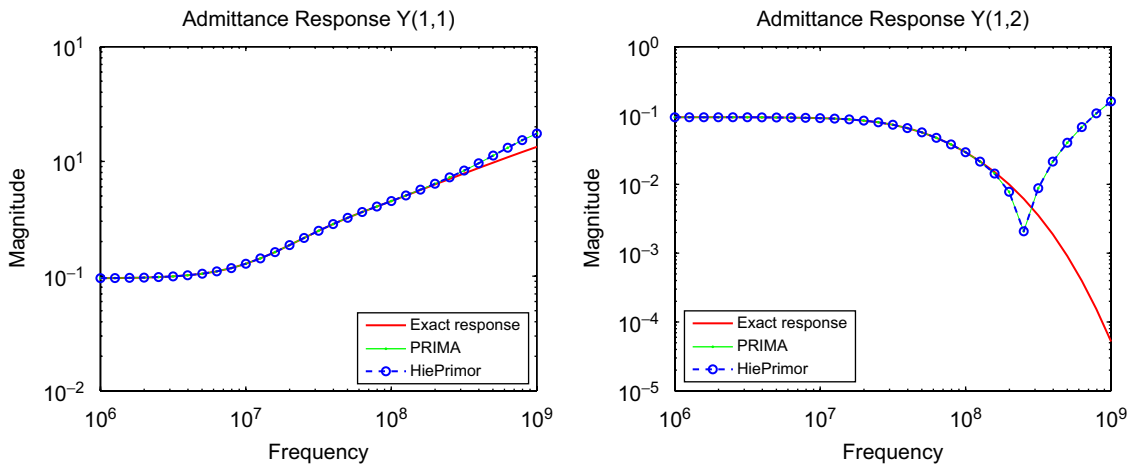

Fig. 2. Accuracy comparison of PRIMA and hiePrimor in Ckt1 when $k = 4$.

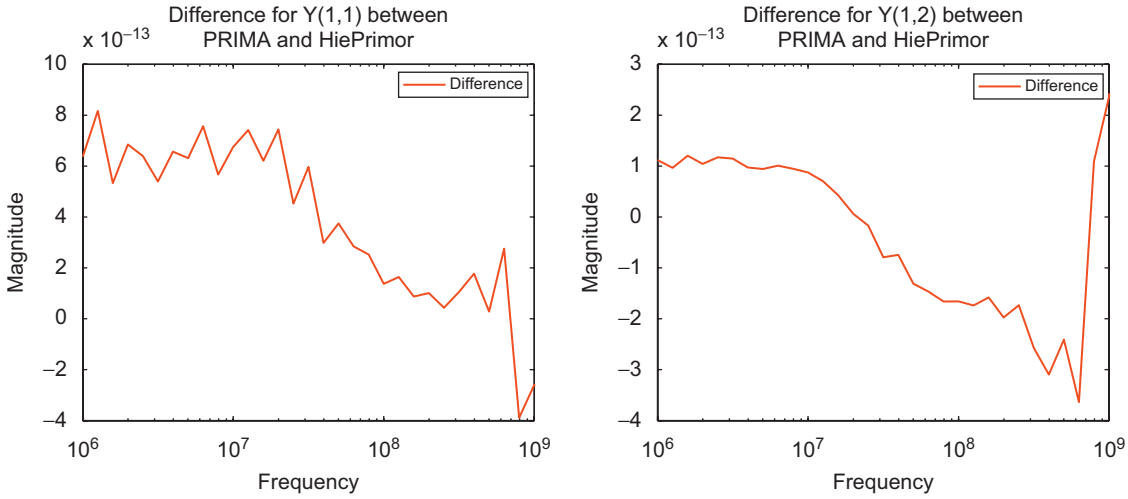

Fig. 3. Difference between PRIMA and hiePrimor in Ckt1 when $k = 4$.

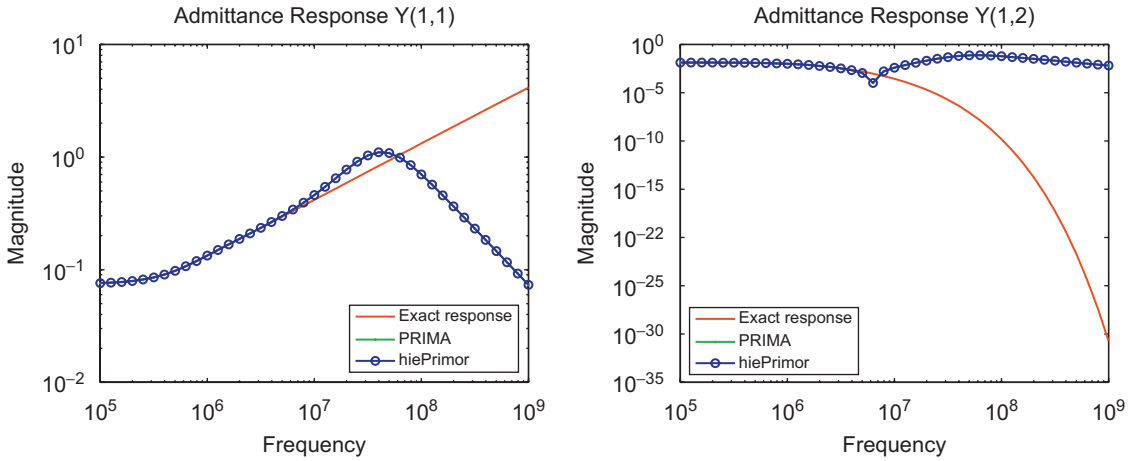

Fig. 4. Accuracy comparison of PRIMA and hiePrimor in Ckt7 when $k = 4$.

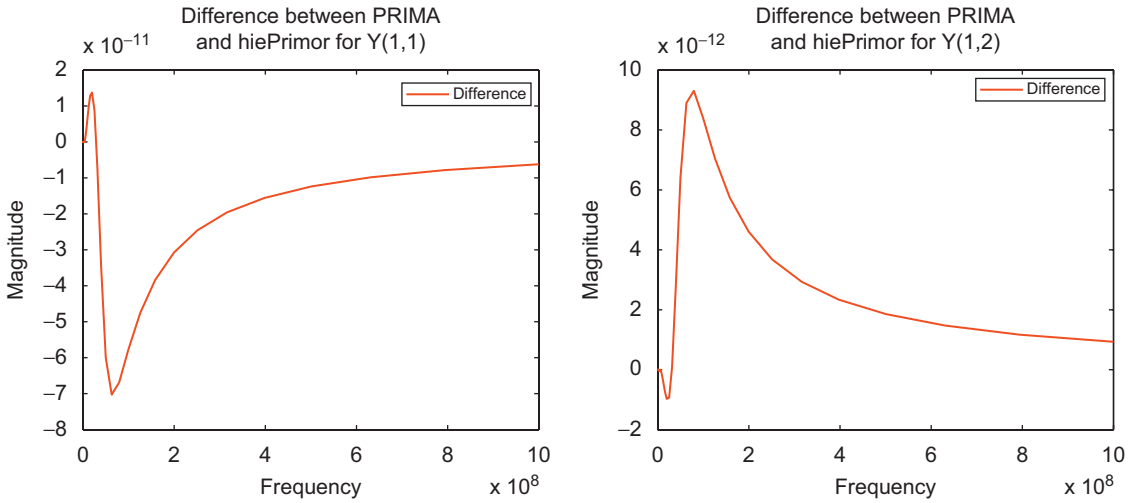

Fig. 5. Difference between PRIMA and hiePrimor in Ckt7 when $k = 4$.

reduction CPU times for them. The last column is the speed up of hiePrimor over PRIMA. We can see that hiePrimor can roughly run five times faster than PRIMA for large scale circuits. It also shows that the hiePrimor has a better performance when the size of the circuits grow larger. Note that such a speed up is gained without any accuracy loss.

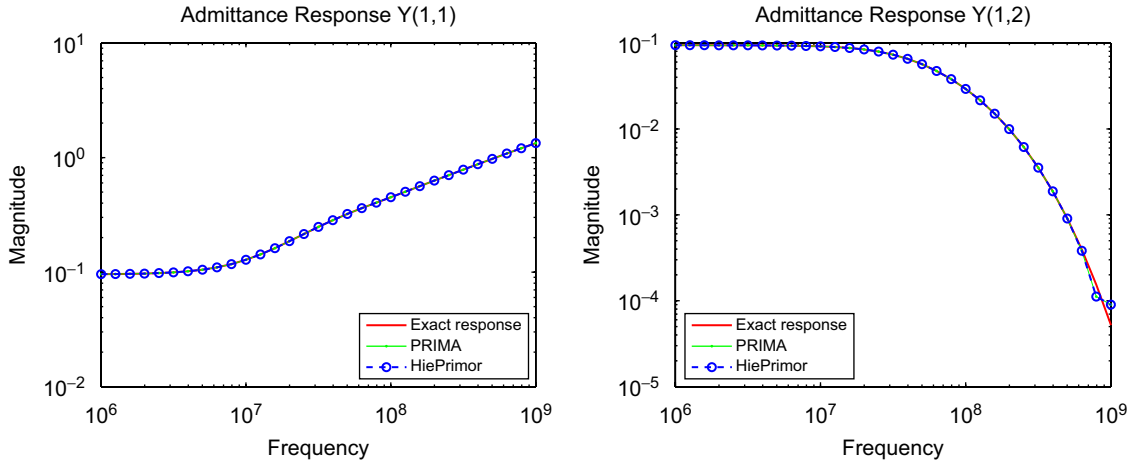

Fig. 6. Accuracy comparison of PRIMA and hiePrimor in Ckt1 when $k = 8$.

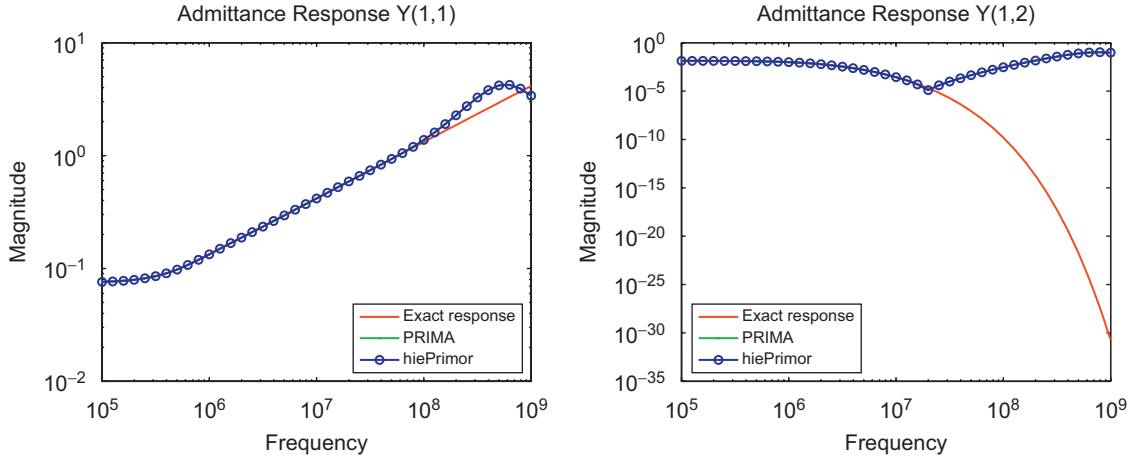

Fig. 7. Accuracy comparison of PRIMA and hiePrimor in Ckt7 when $k = 8$.

Table 1
Reduction time comparison of PRIMA and hiePrimor ($k = 4$, $q = n \times k$)

Test Ckts	#Nodes	w = #Parts	#Ports	PRIMA (s)	hiePrimor (s)	Speed up
Ckt1	25 K	2	8	5	4	1.25
Ckt2	50 K	4	16	16	9	1.78
Ckt3	100 K	8	16	32	13	2.46
Ckt4	200 K	8	16	69	27	2.56
Ckt5	500 K	16	24	248	60	4.13
Ckt6	800 K	16	24	401	99	4.05
Ckt7	1 M	16	32	863	154	5.60
Ckt8	1.5 M	16	20	–	176	–
Ckt9	2 M	32	32	–	136	–
Ckt10	4 M	32	64	–	305	–

Typically, the more partition number we have, the more speed up is attained. But we also need to consider the cost of combining all the lower level subcircuits into higher level. Also as we get more partitions, the ratio of the terminal node count and the internal node count may get smaller, which may hurt the reduction efficiency as the subcircuits may not be effectively reduced. So the number of partitions need to be properly selected practically based on the actual situation. Table 2 shows the relationship between the partition number and the reduction time.

Table 2
Reduction time for different numbers of partitions ($k = 4$, $q = n \times k$)

Test Ckts	w = #Parts = 2	#Parts = 4	#Parts = 8	#Parts = 16
Ckt5	116	100	71	60
Ckt6	374	251	128	99
Ckt7	383	298	204	154
Ckt8	675	394	257	176
Ckt9	363	257	200	164
Ckt10	–	886	582	405

Table 3
Reduction time for different numbers of ports (Ckt7, $k = 4$, $q = n \times k$)

#Ports	PRIMA	hiePrimor	Speed up
8	189	56	3.38
16	339	96	3.53
32	863	154	5.60

Another observation is that hiePrimor becomes more efficient than PRIMA when the number of ports increases. We use different number of ports for the same circuit (Ckt7) using both hiePrimor and PRIMA with the same reduction order. With larger number of ports, hiePimor become faster than PRIMA. Table 3 shows the reduction time comparison of PRIMA and hiePrimor for the same

Table 4
Reduction time comparison of PRIMA and hiePrimor with parallel computing settings ($k = 4$, $q = n \times k$)

Test Ckts	Max sub (s)	Top (s)	Sum (s)	Speed up
Ckt1	2	0	2	2.50
Ckt2	3	1	4	4.00
Ckt3	3	1	4	8.00
Ckt4	5	1	6	11.50
Ckt5	6	1	7	35.43
Ckt6	10	1	11	36.46
Ckt7	17	3	20	43.15
Ckt8	14	1	15	–
Ckt9	8	1	9	–
Ckt10	19	2	21	–

large circuit (Ckt7) with different number of ports. One reason is that ports are dispersed into subcircuits after partitioning and model order at the top level is already much smaller than the original. While for PRIMA, its time complexity is highly related to the number of ports given the same number of block moments k.

Further, we compare the two methods in the artificial parallel computing settings. It is relatively easy to parallelize our method because each subcircuit can be reduced independently. In parallel computing setting, the running time of hiePrimor is only the sum of the maximum subcircuit level reduction time among all the subcircuits and the top-level reduction time (for two-level reduction). The results in Table 4 show that we can have one order of magnitude or more speed up if parallel computing is applied. With more levels, it is reasonable to expect more speed up as more parallelism can be exploited.

9. Conclusion

In this paper, we have proposed a new hierarchical reduction method for interconnect circuits. The new method, called, hiePrimor, applies divide-and-conquer strategy to reduce the reduction complexity and speed up the reduction process. It applies Krylov subspace projection-based reduction on a partitioned circuit where subcircuits are reduced in a bottom-up way until top-level circuit is reduced. Compared to the existing flat projection-based reduction approaches, hiePrimor can give the same accuracy given the same reduction order. It also preserves the passivity of the reduced models as well. We also present a partitioning method and show that partitioning is important and min-span objective is required to achieve the best performance for hierarchical reduction. Experimental results demonstrate that hiePrimor can be significantly faster and scalable than PRIMA given a good partitioning and be much faster if parallel computing is used without loss of accuracy.

References

[1] D. Li, S.X.-D. Tan, B. McGaughy, Hierarchical Krylov subspace reduced order modeling of large rlc circuits, in: Proceedings of the Asia South Pacific Design Automation Conference (ASPDAC), 2008, pp. 170–175.
[2] P. Feldmann, R.W. Freund, Efficient linear circuit analysis by Pade approximation via the Lanczos process, IEEE Trans. Comput. Aided Des. Integrated Circuits Syst. 14 (5) (1995) 639–649.
[3] M. Silveira, M. Kamon, I. Elfadel, J. White, A coordinate-transformed Arnoldi algorithm for generating guaranteed stable reduced-order models of RLC circuits, in: Proceedings of the International Conference on Computer Aided Design (ICCAD), 1996, pp. 288–294.
[4] K.J. Kerns, A.T. Yang, Stable and efficient reduction of large, multiport rc network by pole analysis via congruence transformations, IEEE Trans. Comput. Aided Des. Integrated Circuits Syst. 16 (7) (1998) 734–744.
[5] A. Odabasioglu, M. Celik, L. Pileggi, PRIMA: passive reduced-order interconnect macromodeling algorithm, IEEE Trans. Comput. Aided Des. Integrated Circuits Syst. (1998) 645–654.
[6] S.X.-D. Tan, A general hierarchical circuit modeling and simulation algorithm, IEEE Trans. Comput. Aided Des. Integrated Circuits Syst. (2005) 418–434.
[7] P. Feldmann, Model order reduction techniques for linear systems with large numbers of terminals, in: Proceedings of the European Design and Test Conference (DATE), 2004, pp. 944–947.
[8] P. Liu, S.X.D. Tan, B. Yan, B. McGaughy, An efficient terminal and model order reduction algorithm, integration, VLSI J. 41 (2) (2008) 210–218.
[9] N. Mi, B. Yan, S.X.-D. Tan, Multiple block structure-preserving reduced order modeling of interconnect circuits, integration, VLSI J., doi:http://dx.doi.org/10.1016/j.vlsi.2008.04.006, to appear.
[10] S.X.-D. Tan, L. He, Advanced Model Order Reduction Techniques in VLSI Design, Cambridge University Press, Cambridge, 2007.
[11] L.T. Pillage, R.A. Rohrer, Asymptotic waveform evaluation for timing analysis, IEEE Trans. Comput. Aided Des. Integrated Circuits Syst. (1990) 352–366.
[12] J.R. Phillips, L. Daniel, L.M. Silveira, Guaranteed passive balanced transformation for model order reduction, IEEE Trans. Comput. Aided Des. Integrated Circuits Syst. 22 (8) (2003) 1027–1041.
[13] N. Wang, V. Balakrishnan, Fast balanced stochastic truncation via a quadratic extension of the alternating direction implicit iteration, in: Proceedings of the International Conference on Computer Aided Design (ICCAD), 2005, pp. 801–805.
[14] B. Yan, S.X.-D. Tan, P. Liu, B. McGaughy, SBPOR: second-order balanced truncation for passive model order reduction of RLC circuits, in: Proceedings of the Design Automation Conference (DAC), 2007, pp. 158–161.
[15] B.N. Sheehan, TICER: realizable reduction of extracted RC circuits, in: Proceedings of the International Conference on Computer Aided Design (ICCAD), 1999, pp. 200–203.
[16] B.N. Sheehan, Branch merge reduction of RLCM networks, in: Proceedings of the International Conference on Computer Aided Design (ICCAD), 2003, pp. 658–664.
[17] Z. Qin, C. Cheng, Realizable parasitic reduction using generalized Y-Δ transformation, in: Proceedings of the Design Automation Conference (DAC), 2003, pp. 220–225.
[18] S.X.-D. Tan, A general s-domain hierarchical network reduction algorithm, in: Proceedings of the International Conference on Computer Aided Design (ICCAD), 2003, pp. 650–657.
[19] E.J. Grimme, Krylov projection methods for model reduction, Ph.D. Thesis, University of Illinois, Urbana-Champaign, 1997.
[20] Y. Lee, Y. Cao, T. Chen, J. Wang, C. Chen, HiPRIME: hierarchical and passivity preserved interconnect macromodeling engine for RLC power delivery, IEEE Trans. Comput. Aided Des. Integrated Circuits Syst. 24 (6) (2005) 797–806.
[21] P. Feldmann, F. Liu, Sparse and efficient reduced order modeling of linear subcircuits with large number of terminals, in: Proceedings of the International Conference on Computer Aided Design (ICCAD), 2004, pp. 88–92.
[22] P. Liu, H. Li, L. Jin, W. Wu, S.X.-D. Tan, J. yang, Fast thermal simulation for runtime temperature tracking and management, IEEE Trans. Comput. Aided Des. Integrated Circuits Syst. 25 (12) (2006) 2882–2893.
[23] ⟨http://www.cs.umn.edu/metis⟩.
[24] A. Vandendorpe, P.V. Dooren, On model reduction of interconnected systems, in: Proceedings of the International Symposium on Mathematical Theory Network and Systems, 2004.
[25] B. Salimbahrami, B. Lohmann, Krylov subspace methods in linear model order reduction: introduction and invariance properties, Technical Report, Institute of Automation, University of Bremen, August 2002.
[26] A.E. Ruehli, Equivalent circuits models for three dimensional multiconductor systems, IEEE Trans. Microwave Theory Tech. (1974) 216–220.
[27] B. Krauter, L. Pileggi, Generating sparse partial inductance matrices with guaranteed stability, in: Proceedings of the International Conference on Computer Aided Design (ICCAD), 1995, pp. 45–52.
[28] A. Devgan, H. Ji, W. Dai, How to efficiently capture on-chip inductance effects: introducing a new circuit element K, in: Proceedings of the International Conference on Computer Aided Design (ICCAD), 2000, pp. 150–155.
[29] G. Zhong, C. Koh, K. Roy, On-chip interconnect modeling by wire duplication, in: Proceedings of the International Conference on Computer Aided Design (ICCAD), 2002, pp. 341–346.
[30] G.H. Golub, C.F.V. Loan, Matrix Computations, third ed., Johns Hopkins University Press, Baltimore, MD, 1989.

Duo Li (S'06) received his B.S. and M.S. degrees in Computer Science from Northeastern University, Shenyang and Tsinghua University, Beijing, in 2003 and 2006, respectively. Now he is a Ph.D. candidate in Electrical Engineering at the University of California, Riverside. His current research interests include model order reduction, fast simulation for power grid networks, thermal modeling and thermal simulation.

Sheldon X.-D. Tan (S'96-M'99-SM'06) received his B.S. and M.S. degrees in Electrical Engineering from Fudan University, Shanghai, China in 1992 and 1995, respectively and the Ph.D. degree in Electrical and Computer Engineering from the University of Iowa, Iowa City, in 1999.

He is an Associate Professor in the Department of Electrical Engineering, University of California, Riverside. He was a Faculty Member in the Electrical Engineering Department of Fudan University from 1995 to 1996. His research interests include several aspects of design automation for VLSI integrated circuits—modeling and simulation of analog/RF/mixed-signal VLSI circuits, high performance power and clock distribution network simulation and design, signal integrity, power modeling, thermal modeling, thermal optimization in VLSI physical and architecture levels and embedded system designs based on FPGA platforms.

Dr. Tan is the recipient of NSF CAREER Award in 2004. Dr. Tan received a Best Paper Award from 2007 International Conference on Computer Design (ICCD'07), Best Paper Award Nomination from 2005 IEEE/ACM Design Automation Conference, Best Paper Award from 1999 IEEE/ACM Design Automation Conference and the Best Poster Award from 1999 Spring Meeting of the NSF Center for Design of Analog and Digital Integrated Circuits (CDADIC). He also co-authored book "Symbolic Analysis and Reduction of VLSI Circuits" published by Springer/Kluwer 2005 and Advanced Model Order Reduction Techniques for VLSI Designs, published by Cambridge University Press 2007. He is an Associate Editor for Journal of VLSI Design and served as a Technical Program Committee Member for ASPDAC, BMAS, ASPDAC, ISQED, ICCAD.

Lifeng Wu received the B.S., M.S., and Ph.D. degrees from Tsinghua University, Beijing, China, in 1984, 1986 and 1990, respectively, all in Electrical Engineering. He was an Assistant Professor at the Institute of Microelectronics, Tsinghua University after graduation in 1990. Since 1993, he has been a Research Associate at University of Washington and University of California, Berkeley. He joined BTA Technology in 1995 and Celestry Design Technologies in 2000. He is currently with Cadence Design Systems. His research interests include analog and mixed-signal circuit simulation, power grid IR drop and electromigration analysis, MOSFET device modeling, hot carrier and NBTI reliability modeling and simulation.