

# A Factorization-Based Framework for Passivity-Preserving Model Reduction of RLC Systems\*

Q. Su

School of Electrical and  
Computer Engineering  
Purdue University  
West Lafayette, IN

qsu@ecn.purdue.edu

V. Balakrishnan

School of Electrical and  
Computer Engineering  
Purdue University  
West Lafayette, IN

ragu@ecn.purdue.edu

C.-K. Koh

School of Electrical and  
Computer Engineering  
Purdue University  
West Lafayette, IN

chengkok@ecn.purdue.edu

## ABSTRACT

We present a framework for passivity-preserving model reduction for RLC systems that includes, as a special case, the well-known PRIMA model reduction algorithm. This framework provides a new interpretation for PRIMA, and offers a qualitative explanation as to why PRIMA performs remarkably well in practice. In addition, the framework enables the derivation of new error bounds for PRIMA-like methods. We also show how the framework offers a systematic approach to computing reduced-order models that better approximate the original system than PRIMA, while still preserving passivity.

## Categories and Subject Descriptors

G.1.3 [NUMERICAL ANALYSIS]: Numerical Linear Algebra—*linear systems*; F.2.1 [ANALYSIS OF ALGORITHMS AND PROBLEM COMPLEXITY]: Numerical Algorithms and Problems—*computations on matrices*

## General Terms

Algorithms

## Keywords

Model Reduction, Large Scale Systems, RLC interconnect, Passivity Preserving, Factorization

## 1. INTRODUCTION

As VLSI technology advances, integrated circuits are designed with ever-decreasing sizes, and ever-increasing speeds of operation, with a consequence that RLC interconnect effects have an increasing impact on many critical design criteria [5]. Therefore, accurate

\*This work was supported in part by NSF under contract number CCR-9984553, SRC under contract number 99-TJ-689, and Purdue Research Foundation.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

DAC 2002, June 10-14, 2002, New Orleans, Louisiana, USA.  
Copyright 2002 ACM 1-58113-297-2/01/0006 ...\$5.00.

modeling of RLC interconnect has become very important. A typical interconnect model usually involves thousands or even millions of tightly coupled RLC components, whose direct simulation can stretch the limit of computing resources. A standard practice that addresses such issues is that of model reduction, i.e., finding an approximate model of the original system with far fewer variables. It is highly desirable that this approximate model inherit many of the properties of the original system such as stability and passivity.

Model reduction of linear time-invariant systems is a well-studied topic. Some well-known techniques are moment-matching and projection methods. While these methods are well understood in theory, the challenges in the context of model reduction of VLSI interconnect have arisen from the large-scale nature of these models. Some work towards addressing these challenges are the asymptotic waveform evaluation (AWE) technique due to Huang, Pillage, and Rohrer [14, 20], and Padé approximations through moment matching [6]. For state-space models, Krylov subspace computation techniques such as the Arnoldi and the Lanczos iterations have proved to be very attractive; see for example Padé approximation via Lanczos iterations in [7, 10], and via Arnoldi iterations in [21]. Krylov methods have also been used to efficiently compute a basis for the principal controllable and observable subspaces; see [9]. The reduced-order model then can be obtained by projecting the state-space on these subspaces [12, 13]. Another model reduction technique whose implementation for VLSI interconnect has been recently studied is the balance-and-truncate method (see for example, [18, 11]). The underlying numerical problem with this techniques is the solution of two large-scale Lyapunov equations. Efficient implementation of the balance-and-truncate method is the subject of [17, 2].

The drawback with some (but not all) of the methods described above is that the reduced-order models obtained through them do not inherit the important property of *passivity* of the original model. It has been recognized that when orthogonal projections are employed prior to model reduction, passivity is indeed inherited by the reduced-order model, and methods based on this observations have found favor with the VLSI community. Perhaps the best known method in this class is PRIMA [19], which is based on orthogonal projection of the state vector onto a lower dimensional subspace. It has been recognized that in practice, PRIMA offers excellent time and frequency domain approximation properties. However, there exist no error bounds for PRIMA in the spirit of the system-theoretic error bounds such as those for the balance-and-truncate method.

The main contributions of the this paper are a framework for passivity-preserving model reduction of RLC systems that includes

PRIMA as a special case. The underlying idea is that in order to preserve passivity, the order reduction must be performed not on the original model, but on a ‘‘factor’’, which is analogous to a square root of the real part of the transfer function. This framework offers a number of advantages: It provides a new interpretation for PRIMA, and offers a qualitative explanation as to why PRIMA performs remarkably well in practice. In addition, the framework enables the derivation of new error bounds for PRIMA-like methods. Perhaps most important, the framework offers a systematic approach to computing reduced-order models that better approximate the original system than PRIMA, while still preserving passivity. This improvement comes at the expense of additional computational cost.

## 2. A FACTORIZATION-BASED MODEL REDUCTION FRAMEWORK FOR PASSIVE SYSTEMS

Given an RLC circuit comprising only passive linear elements, we can extract the time-domain equations as follows:

$$\begin{aligned} E\dot{x} &= Ax + Bu, \\ y &= Cx + Du, \end{aligned} \quad (1)$$

where  $x(t) \in \mathbb{R}^N$ ,  $u(t) \in \mathbb{R}$  and  $y(t) \in \mathbb{R}$ . Hence  $E \in \mathbb{R}^{N \times N}$ ,  $A \in \mathbb{R}^{N \times N}$ ,  $B \in \mathbb{R}^{N \times 1}$ ,  $C \in \mathbb{R}^{1 \times N}$  and  $D \in \mathbb{R}$ . We will consider only single-input single-output systems in this paper; the extension of the results presented herein to multi-input multi-output systems is straightforward. We also assume in the sequel that  $E$  is invertible; this assumption is made purely for ease of exposition, in [23] it was shown that a general RLC system with noninvertible  $E$  in (1) can be transformed to an equivalent state space realization where this assumption holds. Since  $E$  is assumed to be invertible, without loss of generality, we assume  $E = I$ .

As (1) describe a stable linear time-invariant system, all the eigenvalues of  $A$  have negative real parts. Moreover, as system (1) is passive, we must have

$$H(j\omega) + H^*(j\omega) \geq 0 \text{ for all } \omega,$$

where  $H(s) = C(sI - A)^{-1}B$  is the transfer function. It turns out that from system theory [1, 3], stability and passivity are equivalent to the existence of a symmetric, positive-definite matrix  $P$  such that

$$M \triangleq \begin{bmatrix} A^T P + PA & PB - C^T \\ B^T P - C & -(D + D^T) \end{bmatrix} \leq 0. \quad (2)$$

The matrix  $P$  can be interpreted as representing the energy stored in the system: It is readily verified that as a consequence of (2), we have

$$\begin{aligned} & \frac{d}{dt}(x(t)^T Px(t)) \\ &= x(t)^T (A^T P + PA)x(t) + 2x(t)^T PBu(t) \\ &= \begin{bmatrix} x(t) \\ u(t) \end{bmatrix}^T M \begin{bmatrix} x(t) \\ u(t) \end{bmatrix} + 2u(t)y(t) \\ &\leq 2u(t)y(t), \end{aligned}$$

so that for every  $T \geq 0$ , we have

$$x(T)^T Px(T) \leq x(0)^T Px(0) + 2 \int_0^T u(t)y(t) dt. \quad (3)$$

For VLSI systems,  $u(t)y(t)$  has the interpretation of the instantaneous power input to the system, so that (3) means that the energy stored in the system never exceeds the initial energy plus the energy input to the system, i.e., it is passive.

Another consequence of (2) is that we have a factorization of  $H(s) + H(-s)$ . Let

$$M = - \begin{bmatrix} L^T \\ D_v^T \end{bmatrix} \begin{bmatrix} L & D_v \end{bmatrix},$$

where  $[L \ D_v]$  has  $r$  linearly independent rows ( $r \leq N$ ). Then, it is readily verified that

$$H(s) + H(-s) = V(-s)^T V(s),$$

where  $V(s) = L(sI - A)^{-1}B + D_v$ . We will refer to  $V(s)$  as a generalized factor of  $H(s)$ . Note that although  $H(s)$  is a scalar transfer function, the generalized factor  $V(s)$  is a column vector with  $r$  entries.

The generalized factor  $V$  suggests a model reduction scheme that preserves passivity and stability for the system (1).

*Factorization-based model-reduction algorithm (FMR)*

*Step 1.* Find any  $P$  satisfying (2).

*Step 2.* Form  $V(s)$ , and find a stable reduced-order approximant  $V_r$ .

*Step 3.* Form  $H_r$  so that it is stable and satisfies  $H_r(s) + H_r(-s) = V_r(-s)^T V_r(s)$ .

Note that  $H_r(s)$  is guaranteed to be stable and passive. In Step 2 above, the reduced-order model can be obtained by any model reduction technique. However, the balanced-truncation method appears to be the natural one for a number of reasons:

1. Although  $V(s)$  is a single-input multi-output transfer function, its model reduction via balanced-truncation is particularly simple. Let  $W_c$  be the controllability Gramian of the realization  $(A, B, L, D_v)$  of  $V$ , given by

$$AW_c + W_c A^T + BB^T = 0.$$

The observability Gramian of the realization  $(A, B, L, D_v)$  of  $V$  is simply the matrix  $P$ . Then, with  $W_c = XX^T$  and  $P = YY^T$  being square root decompositions, let the singular value decomposition (SVD) of  $X^T Y$  satisfy

$$X^T Y = U \Sigma V^T \approx U_k \Sigma_k V_k^T,$$

where the diagonal entries of  $\Sigma_k$  contain the significant Hankel singular values of  $V(s)$ . Then, it can be shown that a realization of  $H_r$  derived from a balanced truncation of  $V$  to  $V_r$  is

$$(T_l A T_r, T_l B, C T_r, D)$$

where

$$T_l = \Sigma_k^{-\frac{1}{2}} V_k^T Y_k^T, \quad T_r = X_k U_k \Sigma_k^{-\frac{1}{2}}.$$

We make the important observation that the matrix  $L$  and  $D_v$  need not be explicitly computed in order to obtain  $H_r$ . All we need are square root decompositions of  $W_c$  and  $P$ .

2. When the balanced-truncation method is used, we have a bound on the approximation error [25, 4]:

$$\leq \frac{1}{2}(4N+1)2\sum_{k=1}^N \sigma_i(4\Sigma_1^k \sigma_i + 2\Sigma_{k+1}^N \sigma_i), \quad (4)$$

where  $\sigma_i$  are the Hankel singular values of  $V(s)$ .

### 3. PRIMA IN THE CONTEXT OF FMR

In order to implement FMR, we first need to find one generalized factor. This corresponds to finding some  $P$  satisfying (2). While this is a convex feasibility problem [3], solving it for a large-scale scenario as that encountered in VLSI CAD models is a daunting task. However, in the case when the system matrices in (1), are obtained via a modified nodal analysis (MNA), such a  $P$  is readily available. In such cases, the state space equations are as follows [19]:

$$\begin{aligned}\hat{E}\dot{x} &= \hat{A}\dot{x} + \hat{B}u, \\ y &= \hat{C}\dot{x},\end{aligned}$$

where the state-space matrices have the following structure.

$$\begin{aligned}\hat{E} &= \begin{bmatrix} F & 0 \\ 0 & L \end{bmatrix} \geq 0, \quad \hat{B} = \hat{C}^T, \\ \hat{A} &= \begin{bmatrix} G & N \\ -N^T & 0 \end{bmatrix}, \text{ with } \hat{A} + \hat{A}^T \leq 0,\end{aligned}\quad (5)$$

As shown in [23], we can always find an equivalent state space realization  $(E, A, B, C, D)$ , where  $E$  is invertible. Further, without loss of generality, we may take  $E = I$ , and

$$\begin{bmatrix} A^T + A & B - C^T \\ B^T - C & -(D + D^T) \end{bmatrix} \leq 0. \quad (6)$$

*The most important observation with the inequality (6) is that  $P = I$  satisfies inequality (2), that is, from the form of the MNA equations, we have “**for free**” a generalized factor.*

Once again, with

$$\begin{bmatrix} A^T + A & B - C^T \\ B^T - C & -(D + D^T) \end{bmatrix} = - \begin{bmatrix} L^T \\ D_v^T \end{bmatrix} [L \ D_v],$$

we have

$$H(s) + H(-s) = V(-s)^T V(s),$$

where  $V(s)$  has a state-space realization  $(A, B, L, D_v)$ . Note that the observability Gramian of this realization is simply the identity (this follows from  $A + A^T + L^T L = 0$ ), so that the balanced truncation method corresponds to simply projecting the state vector on the principal eigenspace of the controllability Gramian, an orthogonal projection. As the principal eigenspace of the controllability Gramian coincides with the principal subspace of the matrix

$$C = [A^{-1}B \ A^{-2}B \ \dots], \quad (7)$$

this method is, in principle, the same as PRIMA. In other words:

*We can interpret PRIMA as an instance of our FMR algorithm, when the generalized factor corresponding to  $P = I$  is used.*

Indeed, we can demonstrate further connections. Suppose the Smith method with shift  $p < 0$  is used to calculate the Gramian  $W_c$  approximately [22, 2]:

$$W_c = \sum_{i=0}^k A_p^i B_p B_p^T (A_p^i)^T,$$

where  $A_p = (pI + A)^{-1}(pI - A)$ ,  $B_p = \sqrt{-2p}(pI + A)^{-1}B$ . The first  $k$  eigenvectors of  $W_c$  in  $V$  span the same column space as  $\text{Krylov}(A_p, B_p, k)$ . Thus, the PRIMA method applied around the expansion point  $p$  can be interpreted as a special case of the FMR

algorithm, with the generalized factor corresponding to  $P = I$ , and with the modified Smith method used to approximately compute  $W_c$ .

While these comments are true in principle, computational issues will lead to minor numerical differences. One issue is how an approximation to the principal eigenspace of the controllability Gramian of the realization  $(A, B, L, D_v)$  is computed: Practical implementations of PRIMA obtain this approximation by computing a basis for the first few columns of the matrix  $C$  in (7), while the FMR algorithm with  $P = I$  corresponds to computing the principal eigenspace by directly computing an approximation to the Gramian.

In addition to a new interpretation for PRIMA, the above development offers an error bound, in principle, for PRIMA:

$$\|H - H_r\|_{L_\infty} \leq \frac{1}{2}(4N+1)2\sum_{k=1}^N \sigma_i(4\sigma_1^k \sigma_i + 2\sum_{j=1}^N \sigma_j), \quad (8)$$

where  $\sigma_i$  are the square roots of the eigenvalues of the controllability Gramian  $W_c$ .

### 4. AN IMPROVED IMPLEMENTATION OF FMR

Motivated by the observation that the error bound in (4) depends crucially on the matrix  $P$  that satisfies (2) (specifically, they depend on the square roots of the eigenvalues of  $W_c P$ , where  $W_c$  is the controllability Gramian of  $(A, B)$ ), with “smaller” values of  $P$  being better, we present a systematic procedure for obtaining  $P < I$ , thereby improving on the performance of PRIMA. We assume that the system is modeled by equations (5). Without loss of generality, we have the same assumption as in last section:  $E = I$  and the inequality (6) holds.

We first explore the set of  $P$  that satisfy (2) for such systems. Applying the Schur complement lemma [3], (2) is equivalent to

$$A^T P + PA + (PB - C^T)(D + D^T)^{-1}(B^T P - C) \leq 0, \quad (9)$$

if  $D$  is nonzero. In the case where  $D = 0$ , (2) is equivalent to

$$A^T P + PA \leq 0, \quad PB = C^T, \quad (10)$$

which can be transformed to an equivalent Riccati inequality of the form

$$\tilde{A}Q + Q\tilde{A}^T + QRQ + S \leq 0 \quad (11)$$

with some lower order  $(N-1)$  matrices  $\tilde{A}$ ,  $R$  and  $S$ . For simplicity of the presentation, we assume  $D \neq 0$ . The procedure for obtaining a smaller solution  $P$  for (10), or equivalently a smaller solution  $Q$  for (11), for the case  $D = 0$  is the same as that for the case  $D \neq 0$ .

Note that  $P = I$  is one feasible solution to (9); this is the “natural” solution obtained from the form of the MNA equations, and the one corresponding to PRIMA, when it is interpreted in the context of our FMR algorithm.

We next turn to the problem of finding the “smallest”  $P$  satisfying (9). It turns out that there is indeed such a minimal element [24, 15]: There exists  $P_{\min} > 0$  such that any  $P$  satisfying (9) also satisfies  $P \geq P_{\min}$ . This matrix  $P_{\min}$  satisfies (9) with equality, i.e.,

$$\begin{aligned}A^T P_{\min} + P_{\min} A \\ + (P_{\min} B - C^T)(D + D^T)^{-1}(B^T P_{\min} - C) = 0,\end{aligned}\quad (12)$$

It also turns out that the quantity  $A^T P_{\min} + P_{\min} A$  is rank one, so that the corresponding generalized factor  $V_{\min}(s)$  is scalar. This is referred to as the *spectral factor* in the system theory literature [8]. Thus, computing  $P_{\min}$  exactly requires the solution of the Riccati equation (12). The corresponding computation is prohibitive even

for moderate-sized problems. Indeed, for moderate-sized problems, the preferred solution method for solving equation (9) is iterative, with each iteration requiring the exact solution of a Lyapunov equation. Therefore, computing  $P_{\min}$  directly is impractical for VLSI systems.

In summary, the case  $P = I$  corresponds to PRIMA, and the optimal  $P_{\min}$ , while it exists, is prohibitively expensive to compute. We therefore propose a scheme that finds a “sub-optimal”  $P$ , one that satisfies (9), and also has the property that  $P_{\min} \leq P \leq I$ , thereby guaranteeing a better error bound than with PRIMA.

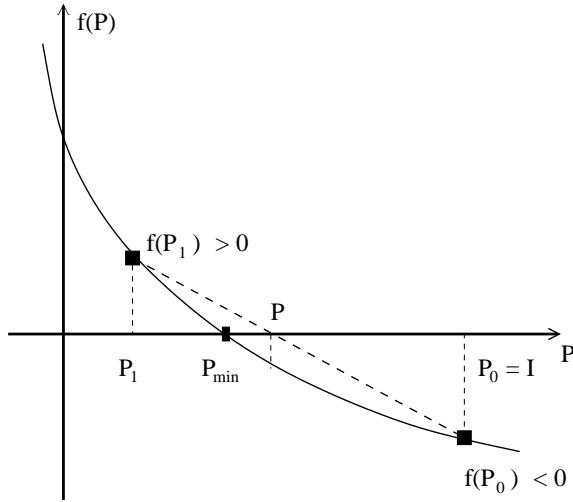
With  $P$  denoting the set of  $P \geq 0$ , we define a function  $f : P \rightarrow \mathbb{R}$  by

$$f(P) = \lambda_{\max}(A^T P + PA + (PB - C^T)(D + D^T)^{-1}(B^T P - C)),$$

where  $\lambda_{\max}(\cdot)$  denotes the maximum eigenvalue of a symmetric matrix. Thus,  $P > 0$  is a solution of (9) if and only if  $f(P) \leq 0$ . It is easily established that  $f$  is a convex function, that is, for all  $P_1, P_2 \in P$  and for all  $\mu \in [0, 1]$ ,

$$f(\mu P_1 + (1 - \mu) P_2) \leq \mu f(P_1) + (1 - \mu) f(P_2). \quad (13)$$

Note that from the definition of  $f$ , we have that  $f(I) \leq 0$ , and that  $f(P_{\min}) = 0$ . Suppose that we can find some  $0 \leq P_1 < I$  with  $f(P_1) > 0$ . As shown in Figure [reffig:alfig](#), by convexity of the function  $f$ , and the fact that  $f(I) \leq 0$ , we can find some  $P = \mu P_1 + (1 - \mu)I$  between  $P_1$  and  $I$ , such that  $f(P) \leq \mu f(P_1) + (1 - \mu)f(I) = 0$ . Such  $P$  is guaranteed to be a solution to (9), because  $f(P) \leq 0$ ; and yield a better error bound than PRIMA, because  $P$  is between  $P_1$  and  $I$ , hence  $P \leq I$ .



**Figure 1: A systematic procedure for obtaining  $P$  with  $0 < P \leq I$  and  $f(P) \leq 0$ .**

It now remains to find  $P_1$  such that  $0 \leq P_1 < I$  with  $f(P_1) > 0$ . One possible choice is  $P_1 = 0$ ; this follows from (9). There are some other choices for  $P_1$ , for example, take  $P_1$  as the solution of the Lyapunov equation

$$(A - B(D + D^T)^{-1}C)^T P_1 + P_1(A - B(D + D^T)^{-1}C) + C^T(D + D^T)^{-1}C = 0.$$

It is easily shown that  $0 < P_1 < I$  and  $f(P_1) > 0$ , as desired.

We can run several iterations of the above procedure to obtain better (i.e., “smaller”) and better  $P$ . Indeed, this iterative procedure

can be adapted to compute  $P_{\min}$ . However, each iteration requires the solution of one Lyapunov equation. As we will demonstrate through numerical examples, it suffices to run one iteration to obtain significant improvement over PRIMA.

## Computational cost over PRIMA

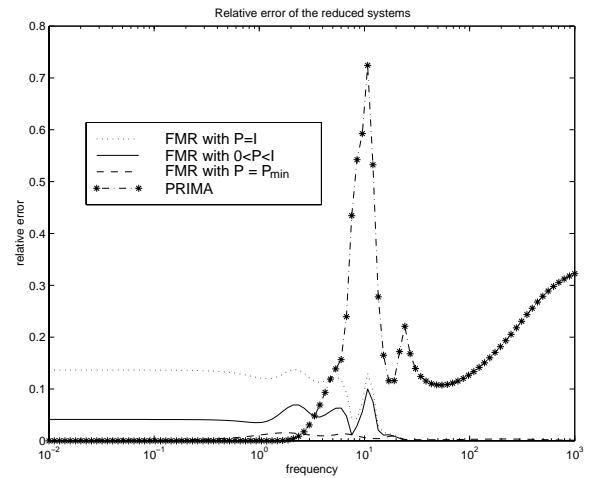
We now briefly discuss the amount of additional computation required by the improved FMR scheme over PRIMA. As PRIMA in principle corresponds to using  $P = I$ , the additional computational costs comes mainly from the evaluation of  $f(P)$  which involves some matrix multiplications during the search of a smaller  $P$ . This is approximately  $4N^3$ .

## 5. NUMERICAL RESULTS

We present two numerical examples that are representative of the performance of the factorization-based passivity-preserving model reduction framework presented in this paper.

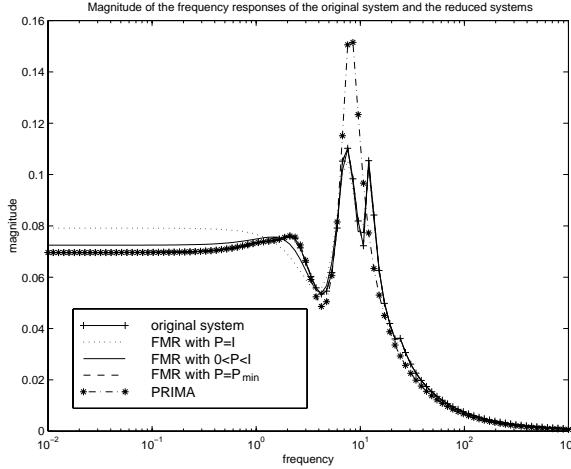
### Example 1

We consider a randomly generated passive system with state space realization  $(A, B, C)$ , where  $A + A^T < 0$ , and  $B = C^T$ . There are 50 state components. We apply the passivity preserving model-reduction algorithm (FMR) to this system to obtain reduced systems of order 13 using three different symmetric, positive-definite matrices: (i)  $P = I$ , (ii)  $0 < P_{\min} < P < I$ , and (iii)  $P = P_{\min}$ . We compute  $P$  in (ii) using one step of the technique outlined in Section 4.  $P_{\min}$  is obtained by solving the Riccati equation (12). With these three matrices, we form three different generalized factors for the original system, and apply the approximate balanced truncation method on them to obtain three different reduced systems. For purposes of comparison, we also apply the PRIMA model-reduction algorithm from [19] on the original system. Figures 2, 3 and 4 show the relative errors and the frequency responses of the four reduced systems. From these plots, it is evident that the reduced order system from FMR with  $P = P_{\min}$  gives us the most accurate approximation. The reduced order system from FMR with  $0 < P_{\min} < P < I$  approximates the original system better the one with  $P = I$ . All the reduced systems from FMR have lower peak errors than the reduced system from PRIMA.

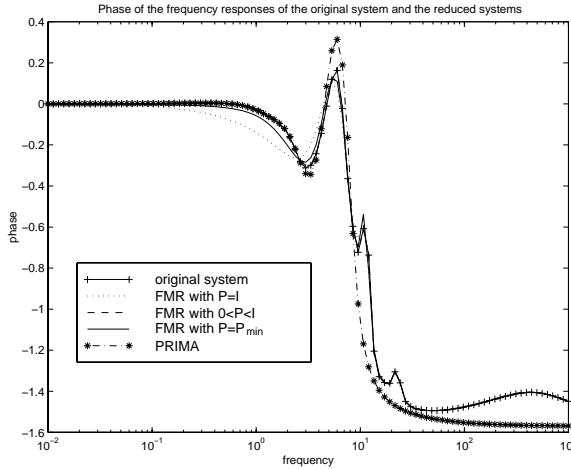


**Figure 2: Relative error of the reduced systems.**

The flop counts of these four model reduction methods are summarized in Table 1. We can see that FMR with  $P = I$  uses roughly



**Figure 3: Magnitude of the frequency responses of the original system and the reduced systems.**



**Figure 4: Phase of the frequency responses of the original system and the reduced systems.**

the same amount of flops as PRIMA. As expected, FMR with  $P = P_{\min}$  costs the most. FMR with  $0 < P_{\min} < P < I$  provides a good compromise; it improves the performance with some additional computational cost.

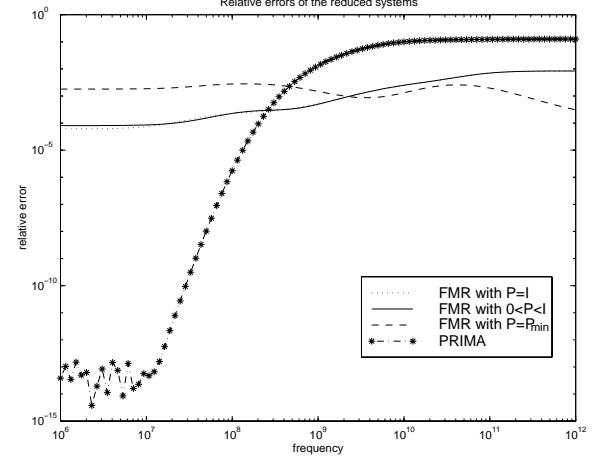
## Example 2

The second example is from [16]. This corresponds to an on-chip planar square spiral inductor suspended over a copper plane. The original system has 500 states. As before, we apply FMR on the system with the different solutions  $P$  to (2) (as defined in the first example) to obtain three reduced systems, order 5. Figures 5 and 6 compare the relative errors and the frequency responses of the three

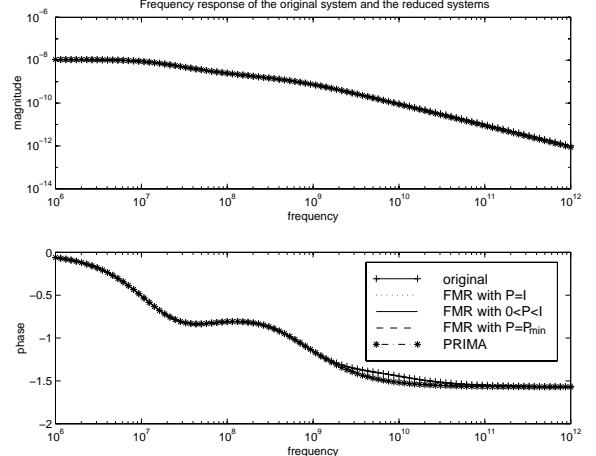
**Table 1: Flop counts comparison.**

	FMR on $P = P_{\min}$	FMR on $0 < P < I$	FMR on $P = I$	PRIMA
flops	$41 \times 10^6$	$2.4 \times 10^6$	$0.7 \times 10^6$	$0.5 \times 10^6$
ratio	79	4.7	1.3	1

reduced systems with those of the reduced system generated by PRIMA. The performance of the three FMR reduced systems follows the same trends exhibited in the preceding example. Moreover, the FMR reduced systems have lower peak errors than the PRIMA reduced system. The flop counts of these four model reduction methods are summarized in Table 2.



**Figure 5: Relative error of the reduced systems.**



**Figure 6: Frequency responses of the original system and the reduced systems.**

## 6. CONCLUSIONS

We have presented a factorization-based framework for passivity-preserving model reduction that includes PRIMA as a special case. We have shown that this framework provides a new interpretation for PRIMA, as well as enabling the derivation of an error bound for PRIMA in principle. In addition, we have a systematic method for improving the performance of PRIMA-like methods.

## 7. REFERENCES

[1] B. Anderson and S. Vongpanitlerd. *Network analysis and synthesis: a modern systems theory approach*. Prentice Hall, 1973.

**Table 2: Flop counts comparison for the RLC circuit.**

	FMR on $P = P_{\min}$	FMR on $0 < P < I$	FMR on $P = I$	PRIMA
flops	$334 \times 10^8$	$24 \times 10^8$	$4.7 \times 10^8$	$3.4 \times 10^8$
ratio	98	7.1	1.4	1

[2] V. Balakrishnan, Q. Su, and C.-K. Koh. Efficient balance-and-truncate model reduction for large scale systems. In *Proc. American Control Conf.*, pages 4746–4751, Arlington, VA, June 2001.

[3] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan. *Linear Matrix Inequalities in System and Control Theory*, volume 15 of *Studies in Applied Mathematics*. SIAM, Philadelphia, PA, June 1994.

[4] X. Chen and T. Wen. Positive realness preserving model reduction with  $h_{\text{infty}}$  norm error bounds. *IEEE Trans. on Circuits and Systems - I: Fundamental Theory and Applications*, 42(1):23–29, 1995.

[5] C.-K. Cheng, J. Lillis, S. Lin, and N. Chang. *Interconnect Analysis and Synthesis*. John Wiley and Sons, Inc., New York, 2000.

[6] E. Chiprout and M. Nakhla. Generalized moment-matching methods for transient analysis of interconnect networks. In *Proc. 29th ACM/IEEE Design Automation Conference*, pages 201–206, Anaheim, CA, June 1992.

[7] P. Feldmann and R. W. Freund. Efficient linear circuit analysis by Padé approximation via the Lanczos process. *IEEE Trans. Computer-Aided Design*, 14:639–649, May 1995.

[8] B. A. Francis. *A course in  $H_\infty$  Control Theory*, volume 88 of *Lecture Notes in Control and Information Sciences*. Springer-Verlag, 1987.

[9] R. W. Freund. Reduced-order modeling techniques based on Krylov subspaces and their use in circuit simulation. Technical Report 98-3-02, Numerical Analysis Manuscript, Bell Laboratories, Murray Hill, New Jersey, Feb. 1998.

[10] K. Gallivan, E. Grimme, and P. V. Dooren. Padé approximation of large-scale dynamic systems with Lanczos methods. In *Proc. IEEE Conf. on Decision and Control*, pages 443–448, Dec. 1994.

[11] M. Green and D. J. N. Limebeer. *Linear Robust Control*. Information and System sciences. Prentice Hall, Englewood Cliffs, NJ, 1995.

[12] E. Grimme, K. Gallivan, V. de Clippel, and P. V. Dooren. Approximate solves in Krylov-based modeling methods. In *Proc. IEEE Conf. on Decision and Control*, pages 3849–3854, Dec. 1997.

[13] P. K. Gunupudi and M. S. Nakhla. Model-reduction of nonlinear circuits using Krylov-space techniques. In *Proc. 29th ACM/IEEE Design Automation Conference*, pages 13–16, 1999.

[14] X. Huang. *Padé Approximation of Linear(ized) Circuit Responses*. PhD thesis, Carnegie Mellon University, Pittsburgh, PA, 1990.

[15] P. Lancaster and L. Rodman. *Algebraic Riccati Equations*. Oxford University Press, 1995.

[16] J. Li, F. Wang, and J. White. An efficient Lyapunov equation-based approach for generating reduced-order models of interconnect. In *Proc. Design Automation Conference*, pages 1–6, 1999.

[17] J. Li and J. White. Efficient model reduction of interconnect via approximate system Gramians. In *Proc. Int. Conf. on Computer-Aided Design*, pages 380–383, 1999.

[18] B. C. Moore. Principal component analysis in linear systems: Controllability, observability, and model reduction. *IEEE Trans. Aut. Control*, AC-26(1):17–32, Feb. 1981.

[19] A. Odabasioglu, M. Celik, and L. T. Pileggi. PRIMA: Passive reduced-order interconnect macromodeling algorithm. *IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems*, 17(8):645–654, 1998.

[20] L. T. Pillage and R. A. Rohrer. Asymptotic waveform evaluation for timing analysis. *IEEE Trans. Computer-Aided Design*, 9:352–366, 1990.

[21] L. M. Silveira, M. Kamon, and J. White. Efficient reduced-order modeling of frequency-dependent coupling inductances associated with 3D interconnect structures. In *Proc. 32nd ACM/IEEE Design Automation Conference*, pages 376–380, June 1995.

[22] R. A. Smith. Matrix equation  $XA + BX = C$ . *SIAM Journal on Applied Mathematics*, 16(1):198–201, 1968.

[23] Q. Su, V. Balakrishnan, and C.-K. Koh. Efficient approximate balanced truncation of general large-scale RLC systems via Krylov methods. In *Proc. ASPDAC/Int. Conf. VLSI Design*, Bangalore, India, Jan. 2002.

[24] J. C. Willems. Least squares stationary optimal control and the algebraic Riccati equation. *IEEE Trans. Aut. Control*, AC-16(6):621–634, Dec. 1971.

[25] K. Zhou. Remarks on stochastic model reduction. *IEEE Trans. Aut. Control*, 32(4):356–358, 1987.