

Efficient Positive-Real Balanced Truncation of Symmetric Systems Via Cross-Riccati Equations

Ngai Wong, *Member, IEEE*

Abstract—We present a highly efficient approach for realizing a positive-real balanced truncation (PRBT) of symmetric systems. The solution of a pair of dual algebraic Riccati equations in conventional PRBT, whose cost constrains practical large-scale deployment, is reduced to the solution of one cross-Riccati equation (XRE). The cross-Riccatian nature of the solution then allows a simple construction of PRBT projection matrices, using a Schur decomposition, without actual balancing. An invariant subspace method and a modified quadratic alternating-direction-implicit iteration scheme are proposed to efficiently solve the XRE. A low-rank variant of the latter is shown to offer a remarkably fast PRBT speed over the conventional implementations. The XRE-based framework can be applied to a large class of linear passive networks, and its effectiveness is demonstrated through numerical examples.

Index Terms—Alternating direction implicit (ADI), cross-Riccati equation (XRE), positive-real balanced truncation (PRBT), Schur decomposition, symmetric systems.

I. INTRODUCTION

VERY LARGE scale integration (VLSI) interconnect and package modelings typically involve massively coupled *RLC* components. The enormous orders of the models thus formed prohibit direct computer simulation. Model order reduction (MOR) has become a standard routine in simulators wherein the original system is approximated by a reduced-order model with little degradation in time/frequency-domain accuracy. In addition, the reduced-order model must preserve critical properties like stability and passivity [1] of the original model to ensure a correct global simulation [2]–[4]. In particular, a passive system is one that does not generate energy internally. A strictly passive system is a dissipative system and is automatically stable. In linear systems, passivity is equivalent to positive realness [1], [4], [5].

Numerous MOR schemes provide various tradeoffs between accuracy and computation. Among them, balanced truncation (BT) offers superior accuracy and closed-form error bounds [3]–[11]. The key idea of BT is to align and sort the internal states of the original model based on their importance in input–output state activities or energy transfer. The least important states are then truncated with little impact on the

system response. In a standard BT, the bottleneck is to solve two linear matrix equations, which are called the Lyapunov equations, for obtaining the controllability and observability Gramians. The cross product of these Gramians is then used to find low-order projection matrices to reduce (truncate) the original system. However, the standard BT does not guarantee passivity. Positive-real BT (PRBT) is another BT variant that preserves both passivity and stability [5] and has no special semidefiniteness constraints on the internal state-space matrices [3]. However, it requires solving a pair of algebraic Riccati equations (AREs) whose complexity is even higher than that of the Lyapunov equations.

In the standard BT of single-input–single-output (SISO) or symmetric multiple-input–multiple-output (MIMO) systems, information in the cross product of the controllability and observability Gramians can be directly obtained by solving a linear matrix equation called the Sylvester equation [6], [11]–[13]. Since only one matrix equation is solved, computation is practically halved, with further advantages like better consistency and numerical robustness [6], [11], [14]. The quadratic counterpart of the Sylvester equation, which is called the cross-Riccati equation denoted by XRE in this paper, appeared in [15]–[19] on control topics like feedback control, bounded-real systems, and discrete stochastic processes. Its integration with the PRBT, however, has not been fully appreciated by the Electronic Design Automation community. Moreover, a few research works have been done on the computationally competitive algorithms for solving large-scale XREs frequently encountered in the VLSI MOR.

By collecting distributed results and introducing new techniques, this paper generalizes, in the context of symmetric MIMO systems, the cross-Gramian framework in the standard BT [6] to the cross-Riccatian counterpart in the PRBT. To begin with, a standard invariant subspace method [15], [20] for solving the XRE, with new spectral characterization and derivations in relation to the dual AREs, is described. A Schur decomposition method, which is borrowed from the standard BT procedure [6], then allows a simple construction of projection matrices to obtain equivalent PRBT-reduced models without actual balancing. Next, to address the computational bottleneck in solving the XRE, a recently proposed fast ARE solver algorithm called the quadratic alternating-direction-implicit (QADI) iteration [8]–[10] is extended to XQADI (“X” for “cross” Riccati) which solves an XRE efficiently. A linear-fractional-transformation (LFT) analysis shows that the XQADI inherits the superlinear convergence of QADI and enjoys simple coding. Nonetheless, the real benefit of the XQADI comes from a mathematically equivalent but low-rank

Manuscript received January 10, 2007; revised June 3, 2007. This work was supported in part by the Hong Kong Research Grants Council under Project HKU 7173/04E and in part by the University Research Committee of The University of Hong Kong. This paper was recommended by Associate Editor L. M. Silveria.

The author is with the Department of Electrical and Electronic Engineering, The University of Hong Kong, Hong Kong, China (e-mail: nwong@eee.hku.hk).

Digital Object Identifier 10.1109/TCAD.2008.915534

implementation of the XQADI called the LRXQADI (“LR” for low rank or left–right). In parallel to the Cholesky-factor QADI (CFQADI) [8]–[10], the LRXQADI exploits and utilizes low-rank factored iterates to significantly speed up computation and reduce memory storage during the iterative XRE solution. The factored cross Riccatian (namely, the solution to the XRE) further permits rapid construction of PRBT projection matrices. The end result is a PRBT/XRE/LRXQADI flow that exhibits a remarkable efficiency comparable with the moment-matching projection-based algorithms such as PRIMA [2], but with the superior global accuracy and deterministic error bound pertinent to the PRBT. Although the proposed approach applies to symmetric systems only, its scope encompasses the large class of passive linear networks, which are commonly used in interconnect and package modelings, that always have symmetric admittance or impedance matrices due to reciprocity. A reciprocal system is the one whose voltage and current values at any two points in the network can be interchanged [21], [22].

This paper is an extension of the earlier work in [20] and is organized as follows. Section II briefly reviews the PRBT. The use of the Sylvester equation in the standard BT of symmetric systems is succinctly discussed. Section III presents the XRE and a standard invariant subspace approach for solving it, together with some new spectral characterizations. The Schur decomposition method for constructing the projection matrices and obtaining the PRBT-equivalent reduced-order model is also described. Section IV introduces the XQADI and the LRXQADI and shows how the low-rank nature of the latter gives rise to significant computational savings. The proposed PRBT/XRE flow is then applied to some practical examples in Section V. Finally, Section VI draws the conclusion.

II. BACKGROUND AND PRELIMINARIES

A. PRBT

Consider the state space of a minimal and passive MIMO square system

$$\dot{x} = A_0 x + B_0 u \quad (1a)$$

$$y = C_0 x + D_0 u \quad (1b)$$

where $A_0 \in \mathbb{R}^{n \times n}$, $B_0, C_0^T \in \mathbb{R}^{n \times m}$, $D_0 \in \mathbb{R}^{m \times m}$, and usually, $m \ll n$. The transfer function/matrix is $G(s) = D_0 + C_0(sI - A_0)^{-1}B_0$. The input u and output y are assumed power conjugated. For every entry of u that is a node voltage (branch current), the corresponding entry of y is a branch current (node voltage) so that $u^T y$ represents a power metric. We assume that A_0 is asymptotically stable, i.e., its spectrum is in the open left half-plane, denoted by $\text{spec}(A_0) \subset \mathbb{C}_-$. Let $M > 0$ (≥ 0) denote a positive definite (positive semidefinite) matrix M . We assume without loss of generality that $D_0 + D_0^T > 0$ (i.e., the system is strictly PR or strictly passive [1]); otherwise, the algorithm in [23] is used to achieve so in a reduced-order embedded system. In addition, an impulse-free system in the descriptor form [4] with a singular E_0 before \dot{x} can be put into the regular form in (1) [3]. The PR

lemma [1], [4] states that the linear system (1) is passive if and only if there exists an $X (\in \mathbb{R}^{n \times n}) > 0$ satisfying the linear matrix inequality

$$\begin{bmatrix} A_0^T X + X A_0 & X B_0 - C_0^T \\ B_0^T X - C_0 & -(D_0 + D_0^T) \end{bmatrix} \leq 0. \quad (2)$$

By Schur complement, (2) is equivalent to

$$A_0^T X + X A_0 + (X B_0 - C_0^T) (D_0 + D_0^T)^{-1} (B_0^T X - C_0) \leq 0. \quad (3)$$

Setting (3) to an equality results in an ARE. In the PRBT, the unique stabilizing solutions [1] $X_c (> 0)$ and $X_o (> 0)$ to the dual AREs

$$\begin{aligned} &A_0 X_c + X_c A_0^T + (X_c C_0^T - B_0) \\ &\quad \times (D_0 + D_0^T)^{-1} (C_0 X_c - B_0^T) = 0 \end{aligned} \quad (4a)$$

$$\begin{aligned} &A_0^T X_o + X_o A_0 + (X_o B_0 - C_0^T) \\ &\quad \times (D_0 + D_0^T)^{-1} (B_0^T X_o - C_0) = 0 \end{aligned} \quad (4b)$$

are solved. Their existence is guaranteed by the passivity assumption. To simplify notations, we define

$$\begin{aligned} B &:= B_0 (D_0 + D_0^T)^{-1/2}, \quad C := (D_0 + D_0^T)^{-1/2} C_0 \\ A &:= A_0 - B_0 (D_0 + D_0^T)^{-1} C_0 \end{aligned} \quad (5)$$

where “:=” denotes assignment, such that (4) also reads

$$A X_c + X_c A^T + X_c C^T C X_c + B B^T = 0 \quad (6a)$$

$$A^T X_o + X_o A + X_o B B^T X_o + C^T C = 0. \quad (6b)$$

Following the standard BT terminology in Section II-B, we call X_c the controllability Riccatian and X_o the observability Riccatian (although they are also sometimes known as the PR Gramians). By factoring out X_c and X_o in (6) for their coefficient matrices, we define

$$A_c := A + X_c C^T C \quad (7a)$$

$$A_o := A + B B^T X_o. \quad (7b)$$

Stabilizability of solutions implies that $\text{spec}(A_c) = \text{spec}(A_o) \subset \mathbb{C}_-$ [7]. Let $X_c = L_c L_c^T$ and $X_o = L_o L_o^T$, where $L_c, L_o \in \mathbb{R}^{n \times n}$, be any matrix square-root decompositions. Compute the singular value decomposition

$$L_c^T L_o = U \Sigma V^T, \quad \text{where } \Sigma = \text{diag}(\sigma_1, \dots, \sigma_n). \quad (8)$$

Here, Σ is a diagonal matrix with descending singular values

$$\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_r \gg \sigma_{r+1} \geq \dots \geq \sigma_n. \quad (9)$$

By defining the matrices

$$T_R := L_c U \Sigma^{-\frac{1}{2}} \quad \text{and} \quad T_L := T_R^{-1} = \Sigma^{-\frac{1}{2}} V^T L_o^T \quad (10)$$

and using “ \rightarrow ” to denote the corresponding similarity transform, we get the PR-balanced model $(\hat{A}_0, \hat{B}_0, \hat{C}_0, D)$, namely

$$(A_0, B_0, C_0, D_0) \rightarrow (\hat{A}_0, \hat{B}_0, \hat{C}_0, D) \\ = (T_L A_0 T_R, T_L B_0, C_0 T_R, D_0). \quad (11)$$

Moreover, in this PR-balanced state space, the controllability and observability Riccatians are simultaneously diagonalized

$$X_c \rightarrow T_L X_c T_L^T = \Sigma \quad \text{and} \quad X_o \rightarrow T_R^T X_o T_R = \Sigma \quad (12)$$

which satisfy

$$\hat{A}_0 \Sigma + \Sigma \hat{A}_0^T + (\Sigma \hat{C}_0^T - \hat{B}_0) (D_0 + D_0^T)^{-1} (\hat{C}_0 \Sigma - \hat{B}_0^T) = 0 \quad (13a)$$

$$\hat{A}_0^T \Sigma + \Sigma \hat{A}_0 + (\Sigma \hat{B}_0 - \hat{C}_0^T) (D_0 + D_0^T)^{-1} (\hat{B}_0^T \Sigma - \hat{C}_0) = 0. \quad (13b)$$

The state variables in $(\hat{A}_0, \hat{B}_0, \hat{C}_0, D_0)$ are aligned in descending importance in energy transfer [3]. Express $\Sigma = \text{diag}(\Sigma_b, \Sigma_s)$, where $\Sigma_b \in \mathbb{R}^{r \times r}$ holds the “bigger” singular values, and $\Sigma_s \in \mathbb{R}^{(n-r) \times (n-r)}$ holds the “smaller” ones. Partition columns of T_R and rows of T_L accordingly so that with respect to (12)

$$X_c = T_R \Sigma T_R^T = [T_{Rb} \quad T_{Rs}] \begin{bmatrix} \Sigma_b & \\ & \Sigma_s \end{bmatrix} \begin{bmatrix} T_{Rb}^T \\ T_{Rs}^T \end{bmatrix} \quad (14a)$$

$$X_o = T_L^T \Sigma T_L = [T_{Lb}^T \quad T_{Ls}^T] \begin{bmatrix} \Sigma_b & \\ & \Sigma_s \end{bmatrix} \begin{bmatrix} T_{Lb} \\ T_{Ls} \end{bmatrix}. \quad (14b)$$

The PRBT-reduced model is then obtained from the rank- r subspace projection

$$(\hat{A}_r, \hat{B}_r, \hat{C}_r, D) = (T_{Lb} A_0 T_{Rb}, T_{Lb} B_0, C_0 T_{Rb}, D_0) \quad (15)$$

where $\hat{A}_r \in \mathbb{R}^{r \times r}$ and $\hat{B}_r, \hat{C}_r^T \in \mathbb{R}^{r \times m}$. It is easily checked that the system in (15) is passive and stable. In addition, the transfer matrix $G_r(s) = D_0 + \hat{C}_r (sI - \hat{A}_r)^{-1} \hat{B}_r$ has an H_∞ -norm approximation error bound determined by the truncated subsystem [5].

B. Sylvester Equation in Symmetric-System Standard BT

The standard BT involves a pair of dual Lyapunov equations

$$A_0 W_c + W_c A_0^T + B_0 B_0^T = 0 \quad (16a)$$

$$A_0^T W_o + W_o A_0 + C_0^T C_0 = 0 \quad (16b)$$

whose solutions W_c and W_o are the controllability and observability Gramians, respectively. The rest of the standard BT procedure is the same as in the PRBT except that X_c and X_o are replaced with W_c and W_o , respectively. Previous work has studied the use of the cross Gramian in the standard BT of the symmetric MIMO systems (which trivially include the SISO systems) [11], [12], [14]. Specifically, the cross Gramian of a symmetric system W_{co} is solved from the Sylvester equation

$$A_0 W_{co} + W_{co} A_0 + B_0 C_0 = 0 \quad (17)$$

where W_{co} has the important property $W_{co}^2 = W_c W_o$. Subsequently, the range or image of $W_{co}(W_{co}^T)$ also spans the reachable (observable) space of (1). Moreover, it can be easily shown that the eigenvalues of W_{co} are invariant under similarity transforms and that their magnitudes are exactly the Hankel singular values of (1). In other words, the dominant left and right eigenspaces of W_{co} capture most of the input–output energy transfer defined in the sense of the Hankel norm [1] (with the input and output energy measures defined by $\int_{-\infty}^0 u^T u dt$ and $\int_0^\infty y^T y dt$, respectively). This fact allows the standard BT to proceed directly by utilizing the eigenvector bases of W_{co} without solving the two Lyapunov equations [6].

III. HAMILTONIAN-BASED SYMMETRIC-SYSTEM PRBT

We first define system symmetry. The MIMO system (A_0, B_0, C_0, D_0) in (1) is symmetric if $G(s) = G(s)^T$ or

$$D_0 + C_0(sI - A_0)^{-1} B_0 = D_0^T + B_0^T (sI - A_0^T)^{-1} C_0^T \quad (18)$$

for all $s \in \mathbb{C} \setminus \text{spec}(A_0)$, where “ \setminus ” denotes set subtraction. This necessitates $D_0 = D_0^T$. An alternative but equivalent definition for a symmetric system is that $D_0 = D_0^T$ and all of the system Markov parameters are symmetric, namely

$$C_0 A_0^i B_0 = (C_0 A_0^i B_0)^T = B_0^T (A_0^T)^i C_0^T \quad (19)$$

for all integers $i \geq 0$. The symmetry and minimality of the system means that (A_0, B_0, C_0) is similar to (A_0^T, C_0^T, B_0^T) , which is denoted by $(A_0, B_0, C_0) \sim (A_0^T, C_0^T, B_0^T)$, through a nonsingular and symmetric $T = T^T \in \mathbb{R}^{n \times n}$ (e.g., [14] and [15], Th. 6.2–4 in [24]) such that

$$A_0^T = T^{-1} A_0 T, \quad C_0^T = T^{-1} B_0, \quad \text{and} \quad B_0^T = C_0 T. \quad (20)$$

It can be checked that the second and third equalities in (20) imply one another. On the other hand, for the matrices defined in (5), we readily have $T = T^T$ and

$$A^T = T^{-1} A T, \quad C^T = T^{-1} B, \quad \text{and} \quad B^T = C T. \quad (21)$$

In other words, $(A, B, C) \sim (A^T, C^T, B^T)$. Analogous to (16) and (17), we formulate, with respect to (6), the XRE

$$A X_{co} + X_{co} A + X_{co} B C X_{co} + B C = 0. \quad (22)$$

Here, X_{co} is called the cross Riccattian as in [16] and [17]. The XRE-based PRBT of symmetric systems consists of two steps, namely, solving the XRE and constructing the projection matrices, which are described in the following.

A. XRE Solution Via Stable Invariant Subspace Identification

We first present a standard way, based on (21) and the Hamiltonian-like approach in [15], for solving (22) that arises from a symmetric system. We also give some new analytical derivations and results regarding the spectral properties of the XRE solution thus obtained [20]. First, define two

Hamiltonian matrices [1] H_c and H_o corresponding to (6a) and (6b), respectively

$$H_c := \begin{bmatrix} A^T & C^T C \\ -BB^T & -A \end{bmatrix}, \quad H_o := \begin{bmatrix} A & BB^T \\ -C^T C & -A^T \end{bmatrix}. \quad (23)$$

The spectral structure of a Hamiltonian matrix and the solutions of (6a) and (6b) by identifying the stable invariant subspaces of H_c and H_o , respectively, are well studied [1], [7]. We note that H_c and H_o share the same spectrum since $H_c = \begin{bmatrix} 0 & I \\ I & 0 \end{bmatrix} (-H_o) \begin{bmatrix} 0 & I \\ I & 0 \end{bmatrix}$. In [7], it has been shown that when the stable and unstable subspaces of H_o are separated (passivity implies no purely imaginary eigenvalues), i.e.,

$$H_o \begin{bmatrix} X_{11} & X_{12} \\ X_{21} & X_{22} \end{bmatrix} = \begin{bmatrix} X_{11} & X_{12} \\ X_{21} & X_{22} \end{bmatrix} \begin{bmatrix} \Phi_- & 0 \\ 0 & \Phi_+ \end{bmatrix} \quad (24)$$

with $\Phi_- \in \mathbb{R}^{n \times n}$ and $\Phi_+ \in \mathbb{R}^{n \times n}$ corresponding to the stable and unstable eigenvalues, respectively, of $\text{spec}(H_o)$, then the stabilizing solution to (6b) is $X_o = X_{21}X_{11}^{-1}$ and that to (6a) is $X_c = X_{12}X_{22}^{-1}$. Now, we show that this invariant subspace approach is also applicable to solving (22). First, define

$$H_{co} := \begin{bmatrix} A & BC \\ -BC & -A \end{bmatrix}. \quad (25)$$

It can be seen that $\text{spec}(H_{co}) = \text{spec}(H_o) = \text{spec}(H_c)$ since

$$H_{co} = \begin{bmatrix} I & 0 \\ 0 & T \end{bmatrix} H_o \begin{bmatrix} I & 0 \\ 0 & T^{-1} \end{bmatrix}. \quad (26)$$

By applying (26) to (24), we get

$$H_{co} \begin{bmatrix} X_{11} & X_{12} \\ TX_{21} & TX_{22} \end{bmatrix} = \begin{bmatrix} X_{11} & X_{12} \\ TX_{21} & TX_{22} \end{bmatrix} \begin{bmatrix} \Phi_- & 0 \\ 0 & \Phi_+ \end{bmatrix}. \quad (27)$$

Then, by noting $\begin{bmatrix} 0 & I \\ I & 0 \end{bmatrix} H_{co} \begin{bmatrix} 0 & I \\ I & 0 \end{bmatrix} = -H_{co}$, (27) is broken into two equations

$$H_{co} \begin{bmatrix} X_{11} \\ TX_{21} \end{bmatrix} = \begin{bmatrix} X_{11} \\ TX_{21} \end{bmatrix} \Phi_- \quad (28a)$$

$$H_{co} \begin{bmatrix} TX_{22} \\ X_{12} \end{bmatrix} = \begin{bmatrix} TX_{22} \\ X_{12} \end{bmatrix} (-\Phi_+). \quad (28b)$$

Simple algebra shows that $TX_{21}X_{11}^{-1} = TX_o$ and $X_{12}X_{22}^{-1}T^{-1} = X_cT^{-1}$ both solve (22). From the Hamiltonian structure, we further have $\text{spec}(\Phi_-) = \text{spec}(-\Phi_+)$. Thus, using $\text{span}(\circ)$ to denote the span (image) of a matrix

$$\text{span} \left(\begin{bmatrix} X_{11} \\ TX_{21} \end{bmatrix} \right) = \text{span} \left(\begin{bmatrix} TX_{22} \\ X_{12} \end{bmatrix} \right) \quad (29)$$

which implies that $TX_o = X_cT^{-1} =: X_{co}$ is a solution to (22) found by identifying the stable invariant subspace of H_{co} . The choice of the actual basis is immaterial as long as the spectrum restricted to it is the same, and from this, the uniqueness of X_{co} follows. Such invariant subspace identification for solving X_{co}

is usually done by the Schur decomposition which, in general, requires $O((2n)^3)$ work and $O((2n)^2)$ memory space.

Recalling (6) and (7), we generalize “stabilizability” in this symmetric-system cross-Riccatian sense. By factoring out X_{co} from both sides in (22) for the coefficient matrices and by noting $X_{co} = X_cT^{-1} = TX_o$ and (20), we get

$$A + X_{co}BC = A_c \quad \text{and} \quad A + BCX_{co} = A_o. \quad (30)$$

It is readily seen that $\text{spec}(A_c) = \text{spec}(A_o) = \text{spec}(\Phi_-) \subset \mathbb{C}_-$, and therefore, we call X_{co} the stabilizing solution to (22) analogously. Similar to the results in Section II-B, we have

$$X_{co}^2 = X_cX_o \quad (31)$$

and the actual T in (20) or (21) is immaterial. With the same notion as in Section II-A, the effect of the similarity transform on X_{co} (in fact, for arbitrary T_R and $T_L = T_R^{-1}$) is

$$X_{co} \rightarrow T_L X_{co} T_R \quad (32)$$

thus, $\text{spec}(X_{co})$ is invariant. Since there exists a similar system in which X_c and X_o are both diagonal [cf. (12)], we have

$$|\lambda_i(X_{co})| = \sigma_i, \quad i = 1, 2, \dots, n \quad (33)$$

where σ_i 's are the singular values in (9), and $\lambda_i(\circ)$ denotes the i th eigenvalue in descending magnitude without loss of generality. By contrasting with Section II-B, it can be concluded that X_{co} plays the same role in the PRBT as W_{co} does in the standard BT. A difference is that the input and output energy measures in the PRBT, by noting that u and y are power conjugated, are $\int_{-\infty}^0 u^T y dt$ and $-\int_0^\infty u^T y dt$, respectively (see [25] for the physical interpretation).

B. Constructing the Projection Matrices

The system balancing in the PRBT, corresponding to (8), (9), (10), and (11), can sometimes be numerically ill conditioned. This is particularly so for large-scale systems with nearly singular X_c and/or X_o , i.e., some states are nearly uncontrollable and/or unobservable [6], [11], [14]. To avoid ill conditionings, we borrow results from [6] on the standard BT and give an alternative way for obtaining a reduced-order model with the same transfer function as the PRBT-reduced model. From (31)

$$\text{span}(X_{co}) = \text{span}(X_c), \quad \text{span}(X_{co}^T) = \text{span}(X_o). \quad (34)$$

Therefore, the cross Riccatian X_{co} carries both the controllable and observable subspaces. By (14) and (31)

$$X_{co}^2 = [T_{Rb} \quad T_{Rs}] \begin{bmatrix} \Sigma_b^2 & 0 \\ 0 & \Sigma_s^2 \end{bmatrix} \begin{bmatrix} T_{Lb} \\ T_{Ls} \end{bmatrix}. \quad (35)$$

Knowing (33), X_{co} is block diagonalized into $V^{-1}X_{co}V = \text{diag}(X_{co}^b, X_{co}^s)$, where $V \in \mathbb{R}^{n \times n}$ is nonsingular [this V should not be confused with the one in (8)]. Here, $X_{co}^b \in \mathbb{R}^{r \times r}$ and $X_{co}^s \in \mathbb{R}^{(n-r) \times (n-r)}$ hold the bigger and smaller eigenvalues, in terms of magnitude, respectively. This block

diagonalization generally takes the $O(n^3)$ work. Next, partition the columns of V and the rows of $W = V^{-1}$ compatibly

$$X_{co} = [V_b \ V_s] \begin{bmatrix} X_{co}^b & 0 \\ 0 & X_{co}^s \end{bmatrix} \begin{bmatrix} W_b \\ W_s \end{bmatrix}. \quad (36)$$

We have that T_{Rb} and V_b span the same (right) eigenspace corresponding to X_{co}^b , whereas T_{Rs} and V_s span the same (right) eigenspace corresponding to X_{co}^s . Therefore, there exist nonsingular $M_1 \in \mathbb{R}^{r \times r}$ and $M_2 \in \mathbb{R}^{(n-r) \times (n-r)}$ such that

$$[T_{Rb} \ T_{Rs}] = [V_b \ V_s] \begin{bmatrix} M_1 & 0 \\ 0 & M_2 \end{bmatrix}. \quad (37)$$

By taking inverse on both sides, we also get

$$\begin{bmatrix} T_{Lb} \\ T_{Ls} \end{bmatrix} = \begin{bmatrix} M_1^{-1} & 0 \\ 0 & M_2^{-1} \end{bmatrix} \begin{bmatrix} W_b \\ W_s \end{bmatrix}. \quad (38)$$

By using (37) and (38), the reduced-order model by projection onto the “more significant” subspaces of (36)

$$(\tilde{A}_r, \tilde{B}_r, \tilde{C}_r, D) = (W_b A_0 V_b, W_b B_0, C_0 V_b, D_0) \quad (39)$$

is easily shown to be similar to (15), thereby producing the same transfer function as the PRBT-reduced model. In other words, an equivalent PRBT-reduced model can be obtained from the block diagonalization of X_{co} without actually balancing the system. To make this paper self-contained, such diagonalization process is included in Appendix I.

IV. QADI-BASED SYMMETRIC-SYSTEM PRBT

The recent work in [8]–[10] generalizes the ADI iteration [26]–[28], which is a highly efficient solver for the Lyapunov equations, to a QADI algorithm for the fast solution of AREs. Consider the ARE

$$A^T X + X A + X B B^T X + C^T C = 0 \quad (40)$$

where A is assumed stable, and a stabilizing solution $X \geq 0$ exists such that $\text{spec}(A + B B^T X) \subset \mathbb{C}_-$. The bounded real lemma [1] states that such an X exists if and only if $\sup \bar{\sigma}(C(j\omega - A)^{-1} B) < 1, \forall \omega \in \mathbb{R}$, where $\bar{\sigma}(\cdot)$ denotes the largest singular value. In its original form, the QADI solves (40) by iterating on two half-steps

$$\begin{aligned} (A^T + X_{j-1}^T B B^T + p_j I) X_{j-\frac{1}{2}}^T \\ = -C^T C - X_{j-1}^T (A - p_j I) \end{aligned} \quad (41a)$$

$$\begin{aligned} (A^T + X_{j-\frac{1}{2}}^T B B^T + p_j I) X_j \\ = -C^T C - X_{j-\frac{1}{2}}^T (A - p_j I) \end{aligned} \quad (41b)$$

where $X_0 = 0$, and the shifts $p_j \in \mathbb{C}_-, j = 1, 2, \dots$, are either real or complex conjugate pairs. For complex conjugate p_j 's, real-value arithmetic, and thus higher numerical efficiency and robustness, is maintained by combining two runs of (41) into one [9]. Furthermore, it has been shown that X_j converges to the stabilizing X superlinearly [8]–[10].

A. XRE Solution Via XQADI

By assuming the existence of a stabilizing solution to (22) and by comparing it to (40), we modify QADI to XQADI that solves an XRE with the following two half-steps:

$$\begin{aligned} (A^T + X_{j-1}^T (BC)^T + p_j I) X_{j-\frac{1}{2}}^T \\ = -(BC)^T - X_{j-1}^T (A^T - p_j I) \end{aligned} \quad (42a)$$

$$\begin{aligned} (A + X_{j-\frac{1}{2}} BC + p_j I) X_j \\ = -BC - X_{j-\frac{1}{2}} (A - p_j I) \end{aligned} \quad (42b)$$

where $X_0 = 0$, and $p_j \in \mathbb{C}_-, j = 1, 2, \dots$, are either real or complex conjugate pairs. Now, define the (lower) LFT [1] as

$$F_1 \left(\begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix}, \Delta \right) = P_{11} + P_{12} \Delta (I - P_{22} \Delta)^{-1} P_{21}$$

where the matrix dimensions and the matrix inverse are implicitly assumed compatible and well defined, respectively. As in [9], we recognize that (42a) and (42b) are expressible as LFTs. Specifically, let $S_j := (A + p_j I)^{-1}$ and $T_j := (A - p_j I)$

$$X_{j-\frac{1}{2}} = F_1 \left(\begin{bmatrix} -BCS_j & -T_j + BCS_j BC \\ S_j & -S_j BC \end{bmatrix}, X_{j-1} \right) \quad (43a)$$

$$X_j = F_1 \left(\begin{bmatrix} -S_j BC & S_j \\ -T_j + BCS_j BC & -BCS_j \end{bmatrix}, X_{j-\frac{1}{2}} \right). \quad (43b)$$

The LFTs in (43) can be nested, using the Redheffer Star Product [9], to get

$$\begin{aligned} X_j = F_1 \left(\begin{bmatrix} M_{11}^{(j)} & M_{12}^{(j)} \\ M_{12}^{(j)} & M_{11}^{(j)} \end{bmatrix}, X_{j-1} \right) \\ = M_{11}^{(j)} + M_{12}^{(j)} X_{j-1} (I - M_{11}^{(j)} X_{j-1})^{-1} M_{12}^{(j)} \end{aligned} \quad (44)$$

where

$$\begin{aligned} M_{11}^{(j)} &= -2p_j S_j B (I - CS_j BCS_j B)^{-1} CS_j \\ &= -2p_j S_j B (I - CS_j B B^T S_j^T C^T)^{-1} CS_j \end{aligned} \quad (45a)$$

$$\begin{aligned} M_{12}^{(j)} &= I - 2p_j S_j (I - BCS_j BCS_j)^{-1} \\ &= I - 2p_j S_j + S_j B C M_{11}^{(j)}. \end{aligned} \quad (45b)$$

To arrive at (45a), we have used the properties that $S_j^T = T^{-1} S_j T$ and $T_j^T = T^{-1} T_j T$ [cf. (21)], and therefore, $CS_j B = (CS_j B)^T$, while the matrix inversion lemma has been used in the simplification of (45b). The matrix inverse in (45a) is well defined since the existence of a stabilizing solution to (22), or equivalently to (40) [see (46)], ensures $\bar{\sigma}(CS_j B) < 1$ due to the bounded real lemma. Our deployment of XQADI consists of an implementation of (44) and (45), for which the matrix inverse in finding S_j has $O(n^3)$ complexity but that in $(I - M_{11}^{(j)} X_{j-1})^{-1}$ is $O(n^2)$ (or, more precisely, $O(n^2 m)$, where m is fixed) by exploiting the usually low-rank $M_{11}^{(j)}$ via matrix inversion lemma. Storage of the X_j iterates, on the other hand, requires $O(n^2)$ memory space.

The convergence of X_j to X_{co} in XQADI can be drawn from that of QADI [8], [10]. To see this, we note that the QADI computes the stabilizing solution, which is denoted by

X_o [see (6b)], to (40). From Section III, we also learn that for a symmetric system, there exists a nonsingular $T \in \mathbb{R}^{n \times n}$ such that (21) holds and the stabilizing solution to (22) is related to the stabilizing solution of (40) by $X_{co} = TX_o$. Subsequently

$$\begin{aligned} & A^T X_o + X_o A + X_o B B^T X_o + C^T C = 0 \\ \Leftrightarrow & T^{-1} A T X_o + X_o A + X_o B C T X_o + T^{-1} B C = 0 \\ \Leftrightarrow & A(T X_o) + (T X_o) A + (T X_o) B C (T X_o) + B C = 0 \\ \Leftrightarrow & A X_{co} + X_{co} A + X_{co} B C X_{co} + B C = 0. \end{aligned} \quad (46)$$

Similar trick also shows that (41a) \Leftrightarrow (42a) and (41b) \Leftrightarrow (42b) by transforming X_{j-1} , $X_{j-(1/2)}$, and X_j in (41) into $T X_{j-1}$, $T X_{j-(1/2)}$, and $T X_j$, respectively. As a result, the XQADI inherits all the properties, including the well posedness and convergence, of the QADI [8], [10]. In particular, following the same techniques as in [10], it can be shown that

$$\|X_j - X_{co}\|_F \leq K_c k(\mathbf{p})^2 \|X_{co}\|_F$$

$$k(\mathbf{p}) = \max_{\forall \lambda_i \in \text{spec}(H_{co}) \cap \mathbb{C}_-} \left| \prod_{k=1}^j \frac{p_k - \lambda_i}{p_k + \lambda_i} \right| \quad (47)$$

where K_c is a positive constant, and $\|\cdot\|_F$ denotes the Frobenius norm. To achieve fast convergence using, for example, L shifts (usually used in a cyclic manner [27]), they are chosen (or approximately chosen) according to the minimax problem

$$\min_{\{p_1, p_2, \dots, p_L\}} \left(\max_{\forall \lambda_i \in \text{spec}(H_{co}) \cap \mathbb{C}_-} \left| \prod_{k=1}^L \frac{p_k - \lambda_i}{p_k + \lambda_i} \right| \right) \quad (48)$$

(see [8]–[10] for details). After getting the converged X_{co} , the construction of the projection matrices can be done in the same manner as in Section III-B.

B. XRE Solution Via LRXQADI

Similar to the CFQADI in [8]–[10], it turns out that a low-rank version of XQADI can also be derived which enables, by utilizing the low-rank iterates, further acceleration and memory savings. The algorithm called LRXQADI, in which “LR” simultaneously stands for low rank and left–right, directly iterates on $Z_L^{(j)}$ and $Z_R^{(j)}$, where $X_j = Z_L^{(j)} Z_R^{(j)}$. In particular, by making use of (44) and (45) and by setting $Z_L^{(0)} = 0$ and $Z_R^{(0)} = 0$, for $j = 1, 2, \dots$,

$$M_{11L}^{(j)} = \sqrt{-2p_j} S_j B (I - C S_j B B^T S_j^T C^T)^{-\frac{1}{2}} \quad (49a)$$

$$M_{11R}^{(j)} = \sqrt{-2p_j} (I - C S_j B B^T S_j^T C^T)^{-\frac{1}{2}} C S_j \quad (49b)$$

$$M_{12}^{(j)} = I - 2p_j S_j + S_j B C M_{11L}^{(j)} M_{11R}^{(j)} \quad (49c)$$

$$Z_L^{(j)} = \left[M_{11L}^{(j)} \ M_{12}^{(j)} \ Z_L^{(j-1)} \left(I - Z_R^{(j-1)} M_{11L}^{(j)} M_{11R}^{(j)} Z_L^{(j-1)} \right)^{-\frac{1}{2}} \right] \quad (49d)$$

$$Z_R^{(j)} = \left[\begin{array}{c} M_{11R}^{(j)} \\ \left(I - Z_R^{(j-1)} M_{11L}^{(j)} M_{11R}^{(j)} Z_R^{(j-1)} \right)^{-\frac{1}{2}} Z_R^{(j-1)} M_{12}^{(j)} \end{array} \right] \quad (49e)$$

Note that $Z_R^{(j-1)} M_{11L}^{(j)} M_{11R}^{(j)} Z_L^{(j-1)} \geq 0$ in (49d) and (49e) since we always have $M_{11R}^{(j)} Z_L^{(j-1)} = (Z_R^{(j-1)} M_{11L}^{(j)})^T$, as proved in Appendix II. Well posedness of the matrix inverses in (49) also parallels the proof in the QADI [8], [10]. It can be easily seen that each sweep of (49) increases the number of columns in $Z_L^{(j)}$ or rows in $Z_R^{(j)}$ by m , i.e., the number of input or output ports. Low-rank B and C also allow the use of matrix inversion lemma in (49) to reduce the computation.

A good stopping criterion for the LRXQADI is the cross-product metric in [29], which monitors the computationally simple Frobenius norm update in $Z_R^{(j)} Z_L^{(j)}$. To see this, basic matrix algebra dictates that $\text{spec}(Z_R^{(j)} Z_L^{(j)}) \cup \{0\} = \text{spec}(Z_L^{(j)} Z_R^{(j)})$, and with reference to (36), the LRXQADI should be terminated once the eigenvalue update is smaller than a preset tolerance. By using (47) and similar techniques as in [29], it is straightforward to show that $Z_R^{(j)} Z_L^{(j)} \rightarrow X_{co}$ asymptotically. For practical circuits, the iterates $Z_L^{(j)}$ and $Z_R^{(j)}$ usually converge in just several to tens of runs of (49) (see numerical examples later); thus, these low-rank factors can provide further computational and memory savings [14]. The matrix inverse operation in S_j , assuming that no structures are exploited, still costs $O(n^3)$ work and constitutes the most expensive step in the LRXQADI, but other than that, the remaining operations are just matrix vector multiplications and small size matrix inverses. Moreover, the factored iterates consume only an $O(nm)$ memory space, thus representing a significant reduction over the invariant subspace and XQADI approaches.

C. Projection Matrices From Low-Rank Factors

Suppose that $Z_L^{(j)}$ and $Z_R^{(j)}$ in LRXQADI converge to Z_L and Z_R , respectively. Then, X_{co} can be numerically equated to, or closely approximated by, the product $Z_L Z_R$, where $Z_L \in \mathbb{R}^{n \times k}$ and $Z_R \in \mathbb{R}^{k \times n}$ and, often, $k \ll n$. In other words

$$\text{span}(X_{co}) = \text{span}(Z_L), \quad \text{span}(X_{co}^T) = \text{span}(Z_R^T). \quad (50)$$

This section presents an effective way, by exploiting the low-rank nature of Z_L and Z_R , to construct the projection matrices without actual balancing. From the results in Section III-B, the projection bases are, in fact, the dominant (left and right) eigenspaces of X_{co} or, equivalently, those of X_{co}^2 . Suppose that the cross product $Z_R Z_L \in \mathbb{R}^{k \times k}$ is block diagonalized, using the same mechanics as in Section III-B and Appendix I (but with the much less $O(k^3)$ work), such that

$$\begin{aligned} Z_R Z_L &= \hat{V} \text{diag} \left(\hat{X}_{co}^b, \hat{X}_{co}^s \right) \hat{W} \\ &= [\hat{V}_b \ \hat{V}_s] \begin{bmatrix} \hat{X}_{co}^b & 0 \\ 0 & \hat{X}_{co}^s \end{bmatrix} \begin{bmatrix} \hat{W}_b \\ \hat{W}_s \end{bmatrix} \end{aligned} \quad (51)$$

where $\hat{W} = \hat{V}^{-1}$, and $\hat{V}_b \hat{X}_{co}^b \hat{W}_b$ represents the “more significant” eigendecomposition of $Z_R Z_L$. Then

$$X_{co}^2 = Z_L Z_R Z_L Z_R \approx Z_L \hat{V}_b \left(\hat{X}_{co}^b \right)^{-1} \left(\hat{X}_{co}^b \right)^2 \hat{W}_b Z_R. \quad (52)$$

Note that since $\text{spec}(Z_R^{(j)} Z_L^{(j)}) \cup \{0\} = \text{spec}(Z_L^{(j)} Z_R^{(j)})$, $(\hat{X}_{co}^b)^2$ is holding the “more significant” eigenvalues of X_{co}^2 . By defining $W_b' := \hat{W}_b Z_R$ and $V_b' := Z_L \hat{V}_b (\hat{X}_{co}^b)^{-1}$, it is straightforward to verify that $W_b' V_b' = I$. Thus, the projection matrices corresponding to the dominant eigenspaces of X_{co}^2 are captured in W_b' and V_b' , respectively. Moreover, similar to (39), the PRBT-reduced model can be obtained by

$$(\tilde{A}_r, \tilde{B}_r, \tilde{C}_r, D) = (W_b' A_0 V_b', W_b' B_0, C_0 V_b', D_0). \quad (53)$$

Therefore, besides the reduced computation and storage due to the low-rank iterates in the LRXQADI, further savings can be achieved in the projection phase because usually, only a low-dimensional matrix decomposition is required in (51) in contrast to the full decomposition in (36).

V. NUMERICAL EXAMPLES

In the context of symmetric systems, the proposed XRE-based PRBT approaches described in Sections III and IV are implemented and compared with the conventional PRBT realizations wherein two AREs are solved. All experiments are done in the Matlab R2006a environment and executed on a 1.8-GHz notebook with 2-GB memory. The AREs are solved with the Matlab routine `aresolv` with `eigen` flag enabled (generally faster than its `schur` option) and also the SLICOT [30] Fortran routine `slcares` (which is faster than its generalized Schur counterpart `slcaregs`) invoked through a Matlab gateway. Moreover, the recent PRBT/CFQADI flow [8], [10], in which the dual AREs are solved by the CFQADI (coded with Matlab m-script files), is contrasted. For the proposed XRE-based PRBT flow, we first solve (22) by identifying the stable invariant subspace as in Section III-A, and the projection matrices were constructed through a block diagonalization of X_{co} as in Section III-B. Both steps essentially make use of the Matlab routine `schur`. As for the XQADI implementations, the original version in (44) and the LRXQADI in (49) are coded and run as Matlab m-script files. The construction of the projection matrices from X_{co} out of XQADI follows that in Section III-B, whereas that from $Z_L Z_R$ out of LRXQADI makes use of the procedures in Section IV-C. The LRXQADI iteration is stopped once the update of eigenvalues in $Z_R^{(j)} Z_L^{(j)}$ is insignificant, as discussed in Section IV-B. For simplicity, a single shift p is used in CFQADI, XQADI, and LRXQADI. The effect of multiple shifts follows from the results in [9] but is not elaborated since this is beyond the theme of this paper. The popular PRIMA algorithm [2], which is expanded at $s = 0$, is also included for comparison. We note that PRIMA is a passivity-preserving moment-matching Krylov-subspace MOR scheme that poses semidefiniteness constraints on the state-space matrices (see [3, Section II-C]) which are unnecessary in the PRBT. Moreover, unlike the automated process in the BT schemes (once an error tolerance is specified), the final model order and the moment-matching frequency expansion points in PRIMA need to be determined *a priori* and may require nontrivial algorithms [31].

We try out five real-life examples [7], [8]. The first is a transmission-line model on the order of 256, the second is

TABLE I
CPU TIMES (IN SECONDS) FOR VARIOUS PRBT
IMPLEMENTATIONS AND PRIMA

	tline256	spiral500	rlc800	rlc2000	rlc3000
2×AREs/ slcares	(3.20)† (0.48) 3.69	(42.13) (3.50) 45.63	(178.95) (15.64) 194.59	N/A	N/A
2×AREs/ aresolv	(4.36) (1.80) 6.16	(43.84) (12.84) 56.69	(217.41) (59.38) 276.78	N/A	N/A
2×AREs/ CFQADI	(4.72) (0.13) 4.84	(2.02) (0.03) 2.05	(1.25) (0.02) 1.27	(43.11) (0.20) 43.31	(59.05) (0.50) 59.55
1×XRE/ invar	(1.86) (0.47) 2.33	(19.25) (3.91) 23.16	(100.42) (29.84) 130.27	(2154.7) (356.3) 2510.9	N/A
1×XRE/ XQADI	(4.72) (0.78) 5.50	(15.53) (4.39) 19.92	(43.55) (20.19) 63.73	N/A	N/A
1×XRE/ LRXQADI	(2.80) (0.36) 3.16	(1.61) (0.13) 1.73	(1.09) (0.11) 1.20	(41.20) (0.13) 41.33	(50.27) (0.36) 50.63
PRIMA	0.13	0.64	2.53	6.25	12.83

† The brackets are time breakdowns: (top) time for solving 2×AREs or 1×XRE; (middle) time for computing the PRBT projection matrices. N/A means out of memory or impractically slow convergence.

a spiral inductor on the order of 500, and the rest are *RLC* ladders on the orders of 800, 2000, and 3000. The CPU times for various PRBT implementations are shown in Table I. We remark that ARE or XRE solutions are unavailable in some high-order cases due to memory exhaustion, and in the XQADI approach, the large-size Matlab matrix multiplications and inversions are too slow that convergence cannot be sought in practical time. Since the ARE solvers `slcares` and `aresolv` and the invariant subspace method are all based on the Hamiltonian matrix or alike, it is not surprising that the latter is the fastest among the three as only one matrix equation needs to be solved. In fact, when the time measurements are available, the PRBT/XRE/invar time is approximately half that of the standard PRBT/ARE/aresolv flow. Nonetheless, these Hamiltonian-based schemes still suffer from a prohibitive computation compared with PRIMA, particularly for large systems. Generally, faster PRBT times are observed in the PRBT/XRE/XQADI method, but they are still relatively expensive and do not scale well with model orders.

The real breakthrough of the PRBT complexity barrier actually comes from the low-rank implementations, namely, CFQADI and LRXQADI, primarily due to their utilization of the low-rank iterates and the fast-decaying singular values inherent in many physical systems. From Table I, it can be seen that the speed gain of PRBT/XRE/LRXQADI is more significant for large systems, and in the third example, its speed is even faster than that of PRIMA. This is inferred by Fig. 1 which shows the quick attenuating cross-Riccatian singular values [cf. (33)] in all the test examples. In our experiments, we terminate CFQADI, XQADI, and LRXQADI once the update of singular values (or eigenvalue magnitudes) drops below $10^{-5} \sim 10^{-10}$; thus, the number of iterations and ranks of the low-rank factors is consistently on the order of tens. Fig. 2 shows $\|X_j - X_{co}\|_F / \|X_{co}\|_F$ [see (47)] for the XRE solution by XQADI or, equivalently, LRXQADI in some test examples.

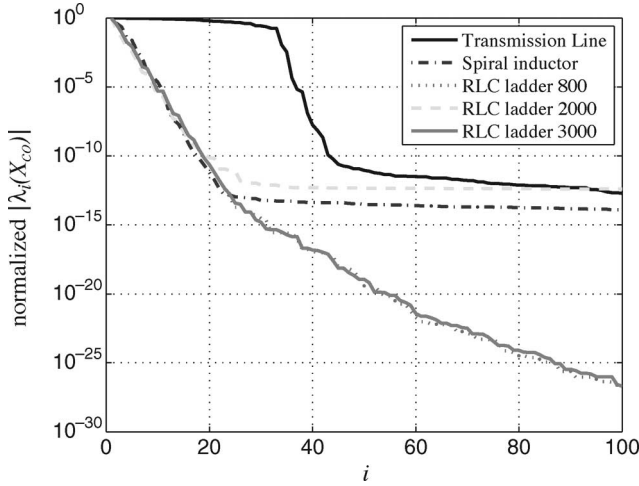
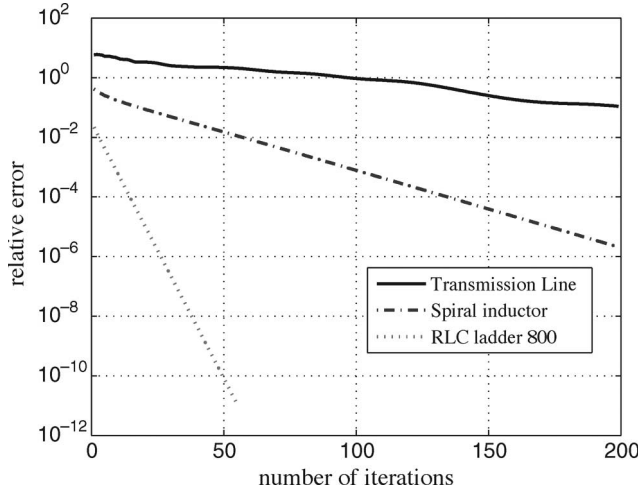


Fig. 1. Cross-Riccatian singular value decay in the examples.

Fig. 2. LRXQADI: Convergence of X_j to the stabilizing X_{co} at several orders.

Linear convergence of LRXQADI can be observed from these virtually straight curves.

Figs. 3(a)–5(a) show the frequency responses of the original systems and the reduced-order models, whereas Figs. 3(b)–5(b) show the relative errors. The plots for the higher order *RLC* ladders are similar to Fig. 5 and, hence, omitted. It is seen that the PRIMA curves are consistently less accurate, which highlights the superior global accuracy of the PRBT. In addition, the curves from the XRE-based approaches practically overlap with those from the conventional PRBT, thereby verifying the validity of the proposed methods in generating equivalent PRBT-reduced response. In fact, in experiments not reported here, we have further tried multipoint moment matching by PRIMA. Besides the extra computation (in expansion-point selection and Krylov iterations) and higher order reduced models, it has been found that the approximation accuracy of multipoint PRIMA is still incomparable to that of the PRBT.

Some remarks are in order.

- 1) Due to finite arithmetic or possible early termination of XQADI or LRXQADI, the cross-Riccatian X_{co} or $Z_L Z_R$ thus obtained may not be exact, thus leaving the question as to whether the reduced-order model is passive. This

issue, however, remains with every numerical implementation of iterative algorithms for the PRBT. Indeed, even with “noniterative” algorithms like the Hamiltonian-based approaches, there are inner iterations which may terminate early in case of ill-conditioned data, resulting in potential problems. This is why passivity check and enforcement, if necessary [32], [33], are still required in practical implementations. To this end, we note the following.

- a) The passivity “check” is usually fast because only the low-order reduced models are involved. The conventional checking procedure is to form a Hamiltonian matrix and then search for imaginary axis eigenvalues [32], [33].
- b) The passivity “enforcement” is much more expensive as it involves finding regions of passivity violation and performing passivity compensation, for example, by first-order perturbation or convex programming [32]–[34]. Moreover, the compensation process will always introduce errors to the approximation accuracy of the reduced-order model and, therefore, should be kept to be as sparing as possible.

Consequently, for algorithms that do not tend to guarantee passivity [like standard BT, Padé via Lanczos, etc.], there can be multiple frequency bands of passivity violation. The “enforcement” step must be invoked iteratively to ensure a passive model at least for the frequencies of interest, and then, the approximation error thus incurred can be high. The PRBT theoretically guarantees a passive reduced model and tends to achieve so in practice. In fact, in our numerical examples, all the reduced-order models are found to be strictly passive. Therefore, in most PRBT implementations, even when nonpassivity is present, passivity enforcement requires only minor correction, and the disturbance on approximation accuracy would be small.

- 2) For simplicity, only a single shift has been used in CFQADI, XQADI, and LRXQADI in our examples. Referring to (48), we have chosen $p = -\sqrt{\rho(H_{co})/\rho(H_{co}^{-1})}$ [7], [9], where $\rho(\circ)$ denotes spectral radius. When computing the shifts p_j ’s, one often makes use of the power iterations of H_{co} and its inverse for approximating the maximal and minimal eigenvalues in $\text{spec}(H_{co})$. When B and C are of low ranks, it is easy to see that the multiplication of H_{co} onto a vector, which is a basic operation in power iteration, scales by $O(n^2)$ rather than $O((2n)^2)$. We can also make use of the matrix inversion lemma to verify that

$$\begin{aligned} H_{co}^{-1} &= \begin{bmatrix} I & -A^{-1}BC \\ 0 & I \end{bmatrix} \begin{bmatrix} A^{-1} & 0 \\ 0 & S^{-1} \end{bmatrix} \begin{bmatrix} I & 0 \\ BCA^{-1} & I \end{bmatrix} \\ S^{-1} &= (-A + BCA^{-1}BC)^{-1} \\ &= -A^{-1} - A^{-1}BCA^{-1}B \\ &\quad \times (I - CA^{-1}BCA^{-1}B)^{-1}CA^{-1} \end{aligned}$$

thus allowing the power iteration of H_{co}^{-1} to be done efficiently with only an $O(n^2)$ work.

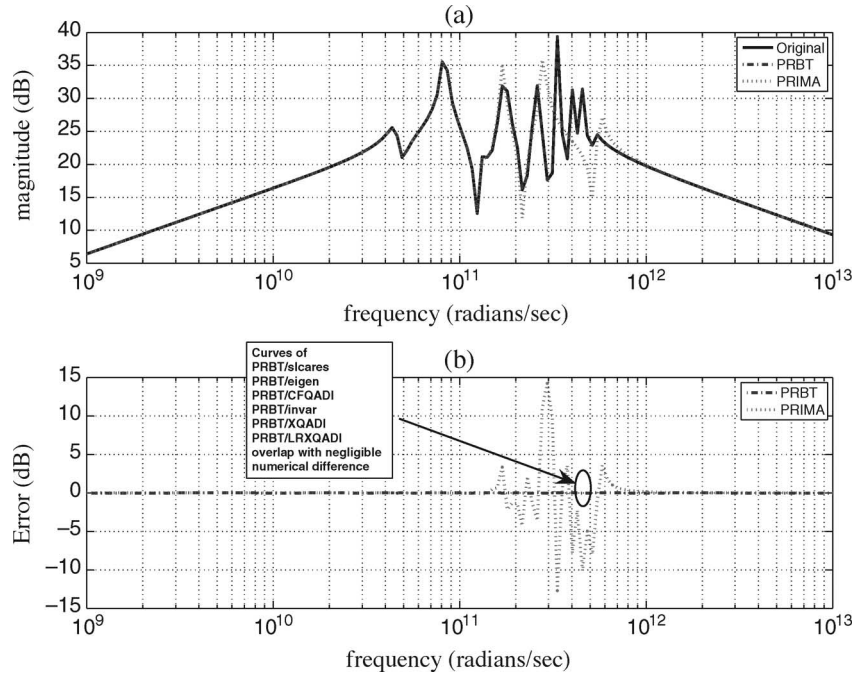


Fig. 3. (a) Frequency responses of the transmission line (order = 256) and reduced models (order = 36). (b) Relative error from the original response.

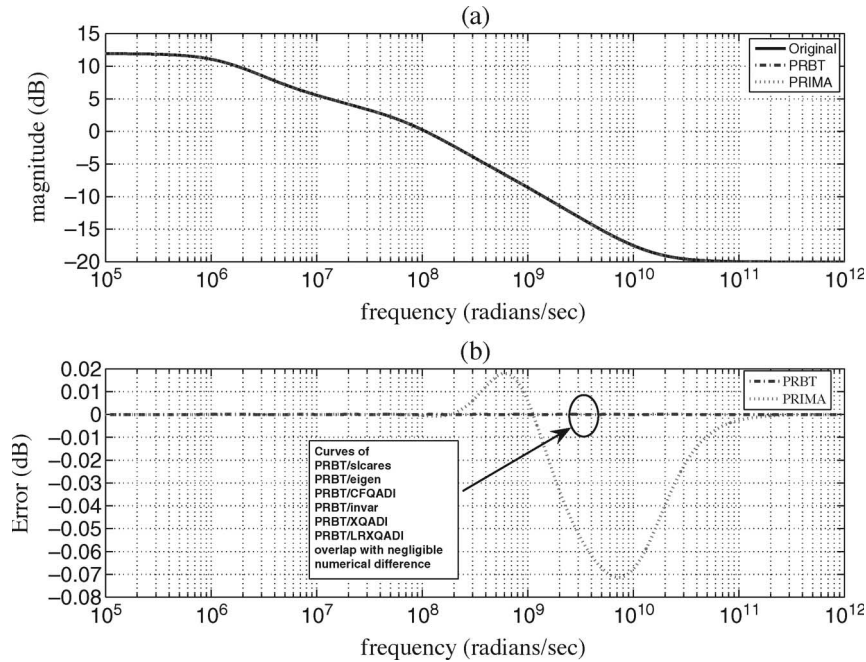


Fig. 4. (a) Frequency responses of the spiral inductor (order = 500) and reduced models (order = 9). (b) Relative error from the original response.

3) In practice, the time for PRBT/LRXQADI is approximately of the same order as that for PRIMA, but the reduced-order model from the PRBT always has superior accuracy. Although a two-stage reduction based on PRIMA and then PRBT is possible [3], [7], any first-stage reduction by PRIMA would cause irremediable loss of approximation accuracy (because PRBT can then at most preserve the fidelity of the intermediate PRIMA-reduced model), and it is hard to determine the intermediate order to switch from PRIMA to PRBT. Given that PRBT/LRXQADI largely reduces the PRBT computation

to an extent comparable to PRIMA, it is justifiable to deploy PRBT/LRXQADI directly on the initial model.

4) PRBT/LRXQADI scales well with the problem size as the termination of LRXQADI is determined by the drop of singular values (eigenvalue magnitudes of the cross Riccadian) pertinent to a physical system, and it is fairly independent of the modeling order. In the same vein, the XRE and the LRXQADI formulations also provide a theoretical tool for analyzing the eigenvalue decay in the cross Riccadian, whose physical importance parallels its Lyapunov counterpart [35].

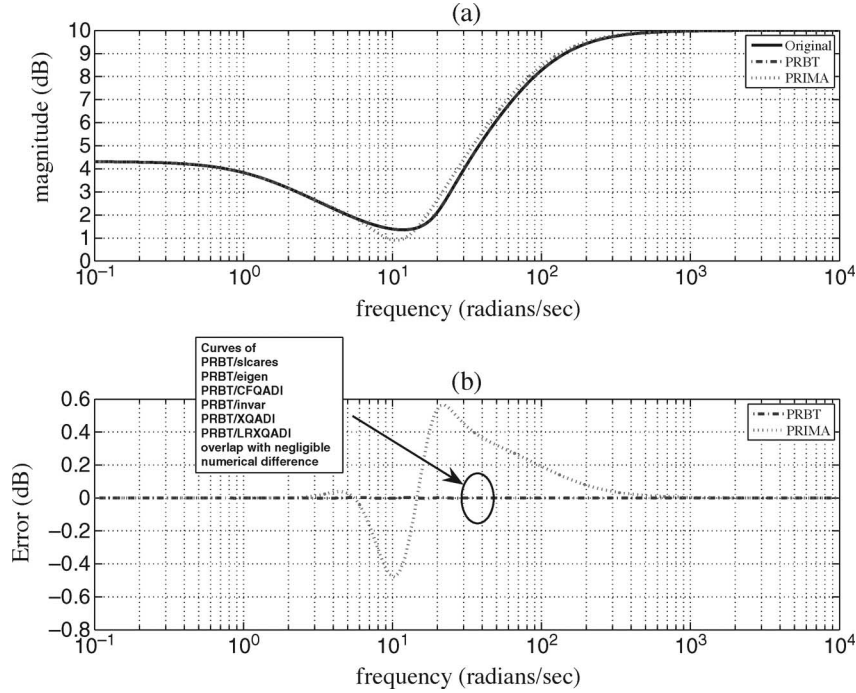


Fig. 5. (a) Frequency responses of the *RLC* ladder (order = 800) and reduced models (order = 8). (b) Relative error from the original response.

VI. CONCLUSION

This paper has presented a computationally efficient cross-Riccatian PRBT implementation applicable to symmetric MIMO systems such as linear passive electrical networks. Instead of solving two AREs in the conventional PRBT, only one XRE has to be solved, which results in good numerical robustness and reduced computation and storage. Theoretical connection between the XRE and the PRBT has been elaborated. An invariant subspace approach, an iterative XQADI algorithm, and an LRXQADI scheme have been described to efficiently solve the XRE. A simple construction of the PRBT projection matrices through the Schur block diagonalization has been adapted to avoid the possibly ill-conditional PR balancing. Application examples have demonstrated the remarkable efficacy of the PRBT/XRE/LRXQADI integration over the traditional PRBT realizations.

APPENDIX I

BLOCK DIAGONALIZATION AND FURTHER PROPERTIES

The block diagonalization of X_{co} follows closely from the work in [6]. To arrive at (36), we first compute an intermediate-ordered real Schur form of X_{co} , namely

$$X_{co} = [Q_b \quad Q_s] \begin{bmatrix} X_{co}^b & \Omega \\ 0 & X_{co}^s \end{bmatrix} \begin{bmatrix} Q_b^T \\ Q_s^T \end{bmatrix} \quad (54)$$

where $X_{co}^b \in \mathbb{R}^{r \times r}$ contains the r bigger magnitude eigenvalues of X_{co} , and X_{co}^b and X_{co}^s are block upper triangular. Next, solve for $\Gamma \in \mathbb{R}^{r \times (n-r)}$ in the Sylvester equation

$$X_{co}^b \Gamma - \Gamma X_{co}^s + \Omega = 0. \quad (55)$$

It can be easily verified that

$$X_{co} = [Q_b \quad Q_b \Gamma + Q_s] \begin{bmatrix} X_{co}^b & 0 \\ 0 & X_{co}^s \end{bmatrix} \begin{bmatrix} Q_b^T - \Gamma Q_s^T \\ Q_s^T \end{bmatrix}. \quad (56)$$

Therefore, with respect to (36), the projection matrices are $V_b = Q_b$ and $W_b = Q_b^T - \Gamma Q_s^T$, respectively. Moreover, by using the notations as in Section III-B and supposing that $\tilde{X}_c, \tilde{X}_o \in \mathbb{R}^{r \times r}$ are the controllability and observability Riccati solutions of $(\tilde{A}_r, \tilde{B}_r, \tilde{C}_r, D)$ in (39), some interesting properties in the standard BT [6] can be generalized to this cross-Riccatian framework

$$\tilde{X}_c = M_1 \Sigma_b M_1^T = W_b X_c W_b^T \quad (57a)$$

$$\tilde{X}_o = M_1^{-T} \Sigma_b M_1^{-1} = V_b^T X_o V_b. \quad (57b)$$

The proof is similar to that in [6] and is therefore omitted.

APPENDIX II

POSITIVE-SEMIDEFINITENESS PROOF IN LRXQADI

Referring to the LRXQADI steps in Section IV-B and (49), we now show that $Z_R^{(j-1)} M_{11L}^{(j)} M_{11R}^{(j)} Z_L^{(j-1)} \geq 0$, $j = 2, 3, \dots$. Obviously, it is sufficient to prove that $(M_{11L}^{(j)})^T (Z_R^{(j-1)})^T = M_{11R}^{(j)} Z_L^{(j-1)}$. Again, using (21) on (49), we have for all j 's

$$\begin{aligned} (M_{11L}^{(j)})^T &= \sqrt{-2p_j} (I - CS_j B B^T S_j^T C^T)^{-\frac{1}{2}} B^T S_j^T \\ &= \sqrt{-2p_j} (I - CS_j B B^T S_j^T C^T)^{-\frac{1}{2}} C T T^{-1} S_j T \\ &= M_{11R}^{(j)} T \end{aligned} \quad (58)$$

and similarly

$$\left(M_{12}^{(j)}\right)^T = T^{-1} M_{12}^{(j)} T. \quad (59)$$

Since $Z_L^{(1)} = M_{11L}^{(1)}$ and $Z_R^{(1)} = M_{11R}^{(1)}$, we have $(Z_R^{(1)})^T = T^{-1} Z_L^{(1)}$. By using the aforementioned results, it is easy to prove that for $j = 2$, $(M_{11L}^{(2)})^T (Z_R^{(1)})^T = M_{11R}^{(2)} Z_L^{(1)}$. Subsequently, we have $Z_R^{(1)} M_{11L}^{(2)} M_{11R}^{(2)} Z_L^{(1)} \geq 0$ and

$$\begin{aligned} \left(Z_R^{(2)}\right)^T &= \left[M_{11R}^{(2)}\right]^T \\ &= \left(M_{12}^{(2)}\right)^T \left(Z_R^{(1)}\right)^T \left(I - Z_R^{(1)} M_{11L}^{(2)} M_{11R}^{(2)} Z_L^{(1)}\right)^{-\frac{1}{2}} \\ &= T^{-1} Z_L^{(2)} \end{aligned}$$

which, in turn, implies that $(M_{11L}^{(3)})^T (Z_R^{(2)})^T = M_{11R}^{(3)} Z_L^{(2)}$ and $Z_R^{(2)} M_{11L}^{(3)} M_{11R}^{(3)} Z_L^{(2)} \geq 0$. The proof continues this way and extends to all j 's.

REFERENCES

- [1] K. Zhou, J. C. Doyle, and K. Glover, *Robust and Optimal Control*. Upper Saddle River, NJ: Prentice-Hall, 1996.
- [2] A. Odabasioglu, M. Celik, and L. T. Pileggi, "PRIMA: Passive reduced-order interconnect macromodeling algorithm," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 17, no. 8, pp. 645–654, Aug. 1998.
- [3] J. R. Phillips, L. Daniel, and L. M. Silveira, "Guaranteed passive balancing transformations for model order reduction," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 22, no. 8, pp. 1027–1041, Aug. 2003.
- [4] Z. Bai, P. M. Dewilde, and R. W. Freund, "Reduced-order modeling," in *Numerical Analysis Manuscript No. 02-4-13*. Murray Hill, NJ: Bell Lab., Mar. 2002.
- [5] X. Chen and J. T. Wen, "Positive realness preserving model reduction with H_∞ norm error bounds," *IEEE Trans. Circuits Syst.*, vol. 42, no. 1, pp. 23–29, Jan. 1995.
- [6] R. W. Aldaheri, "Model order reduction via real Schur-form decomposition," *Int. J. Control*, vol. 53, no. 3, pp. 709–716, Mar. 1991.
- [7] N. Wong, V. Balakrishnan, C.-K. Koh, and T. S. Ng, "Two algorithms for fast and accurate passivity-preserving model order reduction," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 25, no. 10, pp. 2062–2075, Oct. 2006.
- [8] N. Wong and V. Balakrishnan, "Fast balanced stochastic truncation via a quadratic extension of the alternating direction implicit iteration," in *Proc. Int. Conf. Comput.-Aided Des.*, Nov. 2005, pp. 801–805.
- [9] N. Wong and V. Balakrishnan, "Multi-shift quadratic alternating direction implicit iteration for high-speed positive-real balanced truncation," in *Proc. IEEE Des., Autom. Conf.*, Jul. 2006, pp. 257–260.
- [10] N. Wong and V. Balakrishnan, "Fast positive-real balanced truncation via quadratic alternating direction implicit iteration," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 26, no. 9, pp. 1725–1731, Sep. 2007.
- [11] A. C. Antoulas, D. C. Sorensen, and S. Gugercin, "A survey of model reduction methods for large scale systems," *Contemp. Math.*, vol. 280, pp. 193–219, 2001.
- [12] K. V. Fernando and H. Nicholson, "On the structure of balanced and other principal representations of SISO systems," *IEEE Trans. Autom. Control*, vol. AC-28, no. 2, pp. 228–231, Feb. 1983.
- [13] K. V. Fernando and H. Nicholson, "On the cross-Gramian for symmetric MIMO systems," *IEEE Trans. Circuits Syst.*, vol. CAS-32, no. 5, pp. 487–489, May 1985.
- [14] D. C. Sorensen and A. C. Antoulas, "The Sylvester equation and approximate balanced reduction," *Linear Algebra Appl.*, vol. 351/352, pp. 671–700, Aug. 2002.
- [15] L. Fortuna, A. Gallo, C. Guglielmino, and G. Nunnari, "On the solution of a nonlinear matrix equation for MIMO realizations," *Syst. Control Lett.*, vol. 11, no. 1, pp. 79–82, Jul. 1988.
- [16] J. A. Ramos and E. I. Verriest, "A note on a cross Riccati and related properties for symmetric stochastic realizations," *IEEE Trans. Autom. Control*, vol. 34, no. 5, pp. 548–551, May 1989.
- [17] L. Fortuna, A. Gallo, and G. Nunnari, "A new scheme for the approximation of linear systems with bounded stable symmetric transfer matrix," in *Proc. Int. Symp. Circuits Syst.*, 1989, pp. 523–526.
- [18] L. Fortuna, A. Gallo, and G. Nunnari, "New results about the symmetric systems with bounded real transfer matrix," in *Proc. Int. Symp. Circuits Syst.*, 1990, pp. 1760–1763.
- [19] L. Fortuna, G. Muscato, and G. Nunnari, "On H_∞ -control for symmetric systems," in *Proc. IEEE Conf. Decis. Control*, 1992, pp. 3723–3725.
- [20] N. Wong, "Fast positive-real balanced truncation of symmetric systems using cross Riccati equations," in *Proc. Des., Autom. Test Eur.*, Apr. 2007, pp. 1496–1501.
- [21] R. W. Freund and P. Feldmann, "Reduced-order modeling of large passive linear circuits by means of the SyPVL algorithm," in *Proc. Int. Conf. Comput.-Aided Des.*, Nov. 1996, pp. 280–287.
- [22] B. N. Sheehan, "ENOR: Model order reduction of RLC circuits using nodal equations for efficient factorization," in *Proc. IEEE Des., Autom. Conf.*, Jun. 1999, pp. 17–21.
- [23] H. Weiss, Q. Wang, and J. L. Speyer, "System characterization of positive real conditions," *IEEE Trans. Autom. Control*, vol. 39, no. 3, pp. 540–544, Mar. 1994.
- [24] T. Kailath, *Linear Systems*. Englewood Cliffs, NJ: Prentice-Hall, 1980.
- [25] J. C. Willems, "Dissipative dynamical systems II: Linear systems with quadratic supply rates," *Arch. Ration. Mech. Anal.*, vol. 45, no. 5, pp. 353–393, Jan. 1972.
- [26] A. Lu and E. L. Wachspress, "Solution of Lyapunov equations by alternating direction implicit iteration," *Comput. Math. Appl.*, vol. 21, no. 9, pp. 43–58, 1991.
- [27] T. Penzl, "A cyclic low rank Smith method for large sparse Lyapunov equations with applications in model reduction and optimal control," *SIAM J. Sci. Comput.*, vol. 21, no. 4, pp. 1401–1418, 2000.
- [28] J. Li and J. White, "Low-rank solution of Lyapunov equations," *SIAM Rev.*, vol. 46, no. 4, pp. 693–713, 2004.
- [29] V. Balakrishnan, Q. Su, and C.-K. Koh, "Efficient balance-and-truncate model reduction for large scale systems," in *Proc. Amer. Control Conf.*, Jun. 2001, pp. 4746–4751.
- [30] P. Benner, V. Mehrmann, V. Sima, S. Van-Huffel, and A. Varga, "SLICOT—A subroutine library in systems and control theory," *Appl. Comput. Control, Signals, Circuits*, vol. 1, pp. 499–539, 1999.
- [31] E. Chiprout and M. S. Nakhla, "Analysis of interconnect networks using complex frequency hopping (CFH)," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 14, no. 2, pp. 186–200, Feb. 1995.
- [32] S. Grivet-Talocia, "Passivity enforcement via perturbation of Hamiltonian matrices," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 51, no. 9, pp. 1755–1769, Sep. 2004.
- [33] D. Saraswat, R. Achar, and M. S. Nakhla, "Fast passivity verification and enforcement via reciprocal systems for interconnects with large order macromodels," *IEEE Trans. VLSI Syst.*, vol. 15, no. 1, pp. 48–59, Jan. 2007.
- [34] C. P. Coelho, J. R. Phillips, and L. M. Silveira, "A convex programming approach for generating guaranteed passive approximations to tabulated frequency data," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 48, no. 9, pp. 293–301, Sep. 2004.
- [35] A. C. Antoulas, D. C. Sorensen, and Y. Zhou, "On the decay rate of Hankel singular values and related issues," *Syst. Control Lett.*, vol. 46, no. 5, pp. 323–342, Aug. 2002.



Ngai Wong (S'98–M'02) received the B.Eng. (with first class honors) and Ph.D. degrees in electrical and electronic engineering from The University of Hong Kong, Hong Kong, China, in 1999 and 2003, respectively.

He was an Intern with Motorola, Inc., Hong Kong, from 1997 to 1998, specializing in product testing. He was a Visiting Scholar at Purdue University, West Lafayette, IN, in 2003. Currently, he is an Assistant Professor with the Department of Electrical and Electronic Engineering, The University of Hong Kong. His research interests include very large scale integration (VLSI) model order reduction and simulation, digital filter design, sigma-delta modulators, and optimization problems in communication and VLSI applications.

Dr. Wong was the recipient of the P. K. Yu Memorial Scholarship in 2000, the Sir Edward Youde Memorial Fellowship, and the Leung Wai Sun Fellowship in 2002.