

Recent Advances in Structure-Preserving Model Order Reduction

Roland W. Freund

Abstract In recent years, model order reduction techniques based on Krylov subspaces have become the methods of choice for generating small-scale macromodels of the large-scale multi-port RCL networks that arise in VLSI interconnect analysis. A difficult and not yet completely resolved issue is how to ensure that the resulting macromodels preserve all the relevant structures of the original large-scale RCL networks. In this paper, we present a brief review of how Krylov subspace techniques emerged as the algorithms of choice in VLSI circuit simulation, describe the current state-of-art of structure preservation, discuss some recent advances, and mention open problems.

1 Introduction

In 1990, *Asymptotic Waveform Evaluation* (AWE) was introduced as a generalization of the classical Elmore delay to obtain more accurate timing estimates for the increasingly larger RCL networks arising in VLSI interconnect modeling. It was quickly realized that the mathematical concept behind AWE is equivalent to model order reduction based on Padé approximation. The connection between Krylov subspaces and Padé approximation then provided the basis for devising Krylov subspace-based methods for generating macromodels of large-scale multi-port RCL networks. These algorithms satisfy some of the same key properties as AWE, such as moment matching, but they avoid AWE's unstable numerical behavior due to round-off errors and explicit moment computations.

Generating small-scale macromodels of large-scale RCL networks is a special instance of the problem of *Model Order Reduction* (MOR) of large-scale systems, which is a classical and well-studied subject of the area of *Control Theory* in mathe-

Roland W. Freund
Department of Mathematics, University of California at Davis, One Shields Avenue, Davis, CA 95616, USA, e-mail: freund@math.ucdavis.edu

matics. However, in control theory, the systems to be reduced are of relatively small or at most moderate size, and the methods developed in this area typically are too computationally expensive to be viable for large-scale systems. The need to efficiently reduce the ever-increasing sizes of the RCL networks used to model VLSI interconnect led to a renewed interest in MOR and especially in the development of algorithms that are applicable to large-scale systems.

Generally, Krylov subspace-based MOR algorithms have emerged as powerful and versatile tools for model order reduction of large-scale systems. Even though many of these algorithms were originally proposed in the context of VLSI interconnect analysis, they have found applications in many other areas, such as structural analysis and computational acoustics. On the other hand, the problem of MOR of large-scale RCL networks has brought the issue of structure preservation to prominence. The ideal MOR algorithm for large-scale RCL networks would generate small-scale macromodels in the form of RCL networks. Krylov subspace-based MOR algorithms, however, are linear algebra techniques, and they produce macromodels described by data matrices, rather than networks. A more modest goal then is to generate macromodels that preserve essential properties of RCL networks, such as passivity and reciprocity.

The *Passive Reduced-order Interconnect Macromodeling Algorithm* (PRIMA) was the first Krylov subspace-based method that produces passive macromodels of RCL networks. The key insight behind PRIMA is that passivity can be preserved by using explicit projection of the data matrices describing the original RCL network onto Krylov subspaces and by giving up about half the moment matching satisfied by MOR based on Padé approximation. On the other hand, PRIMA does not preserve other structures inherent to RCL networks, such as reciprocity, which makes it harder to synthesize the PRIMA macromodels as actual electrical networks.

The *Structure-Preserving Reduced-order Interconnect Macromodeling* (SPRIM) algorithm was introduced as a structure-preserving variant of PRIMA that overcomes many of the shortcomings of PRIMA and at the same time, is more accurate than PRIMA. The purpose of this paper is twofold. First, we review PRIMA and SPRIM and their basic properties. Second, we discuss some recent advances in structure-preserving MOR. In particular, we describe a thick-restart Krylov subspace approach that allows the use of complex and multiple expansion points for the underlying moment-matching property, and we discuss some techniques designed for the efficient implementation of this approach.

This paper is organized as follows. In Section 2, we summarize the standard description of general RCL networks and some of their key properties. In Section 3, we present a brief review of the evolution of Krylov subspace-based model order reduction from moment matching to the currently dominating paradigm of projection onto Krylov subspaces. In Sections 4 and 5, we describe PRIMA and SPRIM, respectively. In Section 5, we discuss some recent advances in adapting thick-restart Krylov subspace techniques to the problem of model order reduction. Section 6 addresses some issues arising in moment matching with complex expansion points. Finally, we make some concluding remarks in Section 7.

Throughout this paper the following notation is used. The set of real and complex numbers is denoted by \mathbb{R} and \mathbb{C} , respectively. For general (real or complex) matrices $M = [m_{jk}]$, $M^T := [m_{kj}]$ is the *transpose* of M , and $M^H := [\overline{m_{kj}}]$ is the *Hermitian* (or *complex conjugate*) of M . The $n \times n$ identity matrix is denoted by I_n , or simply I if its size is apparent from the context. The zero matrix is denoted by 0 ; its actual size will always be apparent from the context. The notation $M \succeq 0$ ($M \succ 0$) is used to indicate that a real or complex square matrix M is *Hermitian positive semidefinite* (*positive definite*). If all entries of the matrix $M \succeq 0$ ($M \succ 0$) are real, then M is said to be *symmetric positive semidefinite* (*positive definite*). Finally, for a matrix $V \in \mathbb{R}^{N \times n}$, we denote by $\text{colspan } V$ the linear subspace of \mathbb{R}^N spanned by the columns of V . Note that $\text{colspan } V$ has dimension n if, and only if, $\text{rank } V = n$.

2 Description of RCL Networks

An *RCL network* is an electronic circuit with linear resistors, capacitors, inductors, and independent voltage and current sources as its only elements. In this section, we briefly review the standard description and some key properties of such RCL networks.

2.1 RCL Network Equations

A system of equations describing any electronic circuit can be obtained by combining *Kirchhoff's current laws* (KCLs), *Kirchhoff's voltage laws* (KVLs), and the *branch constitutive relations* (BCRs). The BCRs are the equations that describe the electronic behavior of each of the circuit elements. For example, the BCR of a resistor is Ohm's law.

An elegant approach to formulating KCLs and KVLs for a given electronic circuit is based on the representation of the circuit topology as a directed graph; see, e.g., [9, 33, 35, 32]. The edges of such a graph correspond to the elements of the circuit, and the nodes correspond to the nodes of the circuit, which represent the interconnections of the circuit elements. Each edge e can be expressed as an ordered pair $e = (n_1, n_2)$ of nodes n_1 and n_2 , where the direction of e is from n_1 to n_2 . For circuit elements for which the direction of the electric current through the element is known beforehand, the direction of the associated edge is chosen accordingly. For any other circuit element, an arbitrary direction is assigned to the associated edge. If the computed electric current through an element is nonnegative, then the current flow is in the direction of the edge; otherwise, the actual current flow is against the direction of the edge. Such a directed graph can be described by its *incidence matrix* the rows and columns of which correspond to the nodes and edges, respectively, and the entries of which are defined as follows. The column associated with

edge $e = (n_1, n_2)$ contains the entry “1” in the row position corresponding to node n_1 , the entry “-1” in the row position corresponding to node n_2 , and zero entries otherwise. Since the resulting matrix is always rank deficient, one deletes the row associated with the *ground node* of the circuit. We refer to the matrix obtained after this deletion as the incidence matrix \mathcal{A} of the circuit. In terms of \mathcal{A} , all of the circuit’s KCLs and KVLs can be stated compactly as follows:

$$\mathcal{A} i_{\mathcal{E}} = 0 \quad \text{and} \quad \mathcal{A}^T v = v_{\mathcal{E}}. \quad (1)$$

Here, $i_{\mathcal{E}}$ is the vector the entries of which are the currents through all the circuit elements, $v_{\mathcal{E}}$ is the vector the entries of which are the voltages across all the circuit elements, and v is the vector the entries of which are the voltages at the nodes of the circuit, except for the ground node at which the voltage is zero.

We now assume that the given electronic circuit is an RCL network. We use subscripts r , c , l , v , and i , to refer to edge quantities corresponding to the resistors, capacitors, inductors, voltage sources, and current sources, respectively, of the given RCL network. Moreover, we assume that the edges of the associated directed graph are ordered according to element type. Consequently, the graph’s incidence matrix \mathcal{A} and the vectors $i_{\mathcal{E}}$ and $v_{\mathcal{E}}$ can be partitioned as follows:

$$\mathcal{A} = [\mathcal{A}_r \quad \mathcal{A}_c \quad \mathcal{A}_l \quad \mathcal{A}_v \quad \mathcal{A}_i], \quad i_{\mathcal{E}} = \begin{bmatrix} i_r \\ i_c \\ i_l \\ i_v \\ i_i \end{bmatrix}, \quad v_{\mathcal{E}} = \begin{bmatrix} v_r \\ v_c \\ v_l \\ v_v \\ v_i \end{bmatrix}.$$

Furthermore, the BCRs for the resistors, capacitors, and inductors can be stated in the following compact form:

$$v_r(t) = R i_r(t), \quad i_c(t) = C \frac{d}{dt} v_c(t), \quad v_l(t) = L \frac{d}{dt} i_l(t). \quad (2)$$

Here, R and C are diagonal matrices, the diagonal entries of which are the resistances of the resistors and the capacitances of the capacitors, respectively, and in particular, $R \succ 0$ and $C \succ 0$. The matrix L contains the inductances between the inductors as its entries. If mutual inductances are included, then L is a full matrix; otherwise, L is also a diagonal matrix. In both cases, $L \succ 0$. Therefore, the matrices in (2) always satisfy

$$R \succ 0, \quad C \succ 0, \quad \text{and} \quad L \succ 0. \quad (3)$$

Finally, the BCRs for the independent voltage and current sources of the RCL network simply state that

$$v_v(t) \quad \text{and} \quad i_i(t) \quad (4)$$

are given functions that can be chosen as inputs to the network, whereas

$$v_i(t) \quad \text{and} \quad i_v(t) \quad (5)$$

are unknown output functions that need to be determined as part of the problem of solving the system of equations describing the given RCL network.

The KCLs and KVLs (1) together with the BCRs (2) yield a system of equations for the unknown circuit quantities $v(t)$, $v_{\mathcal{E}}$, and $i_{\mathcal{E}}$. The size of this system can be reduced significantly by using some of the relations in (1) and (2) to eliminate v_r , v_c , v_l , i_r , and i_c . This process is known as modified nodal analysis and results in the system of equations

$$\begin{aligned} \mathcal{A}_c C \mathcal{A}_c^T \frac{d}{dt} v(t) + \mathcal{A}_r R^{-1} \mathcal{A}_r^T v(t) + \mathcal{A}_l i_l(t) + \mathcal{A}_v i_v(t) &= -\mathcal{A}_i i_i(t), \\ L \frac{d}{dt} i_l(t) - \mathcal{A}_l^T v(t) &= 0, \\ \mathcal{A}_v^T v(t) &= v_v(t) \end{aligned} \quad (6)$$

for the remaining unknown quantities $v(t)$, $i_l(t)$, and $i_v(t)$; see, e.g., [6, Section 5.2] or [21]. Recall from (4) that the functions $i_i(t)$ and $v_v(t)$ appearing on the right-hand side of (6) are given. Furthermore, the unknown output function $v_i(t)$ in (5) can be easily obtained from $v(t)$ via the relation

$$v_i(t) = \mathcal{A}_i^T v(t), \quad (7)$$

which follows from the KVLs (1).

For the purpose of model order reduction, it is convenient to state the equations (6) and (7) as a *descriptor system* of the form

$$\begin{aligned} E \frac{d}{dt} x(t) &= A x(t) + B u(t), \\ y(t) &= B^T x(t). \end{aligned} \quad (8)$$

Here, we have set

$$u(t) := \begin{bmatrix} -i_i(t) \\ v_v(t) \end{bmatrix}, \quad y(t) := \begin{bmatrix} v_i(t) \\ -i_v(t) \end{bmatrix}, \quad x(t) := \begin{bmatrix} v(t) \\ i_l(t) \\ i_v(t) \end{bmatrix}, \quad (9)$$

and

$$A := \begin{bmatrix} A_{11} & -\mathcal{A}_l & -\mathcal{A}_v \\ \mathcal{A}_l^T & 0 & 0 \\ \mathcal{A}_v^T & 0 & 0 \end{bmatrix}, \quad E := \begin{bmatrix} E_{11} & 0 & 0 \\ 0 & L & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad B := \begin{bmatrix} \mathcal{A}_i & 0 \\ 0 & 0 \\ 0 & -I \end{bmatrix}, \quad (10)$$

where

$$A_{11} := -\mathcal{A}_r R^{-1} \mathcal{A}_r^T \quad \text{and} \quad E_{11} := \mathcal{A}_c C \mathcal{A}_c^T. \quad (11)$$

The vector-valued functions $u(t)$, $y(t)$, and $x(t)$ are called *input vector*, *output vector*, and *state-space vector* of the descriptor system (8). We denote by m the length

of the input and output vectors and by N the length of the state-space vector. The integer N is called the *state-space dimension* of (8). In view of (9), m is the total number of voltage and current sources of the given RCL network, and N is the sum of the number of nodes (excluding the ground node), the number of inductors, and the number of voltage sources. Note that $A, E \in \mathbb{R}^{N \times N}$, $B \in \mathbb{R}^{N \times m}$, and the matrix E is singular in general. Finally, we remark that A and E satisfy the semidefiniteness conditions

$$A + A^T = \begin{bmatrix} 2A_{11} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \succeq 0 \quad \text{and} \quad E = E^T \succeq 0, \quad (12)$$

which readily follow from (3), (10), and (11).

2.2 RCL Transfer Functions

The first relation of the descriptor system (8) represents a linear system of differential-algebraic equations. In Laplace domain, this relation is transformed into a purely algebraic linear system that can be used to formally eliminate the state-space vector. To this end, we denote by $\hat{x}(s)$, $\hat{u}(s)$, and $\hat{y}(s)$ the Laplace transform of $x(t)$, $u(t)$, and $y(t)$, respectively. Application of the Laplace transform to the time-domain system (8) results in the Laplace-domain system

$$\begin{aligned} sE\hat{x}(s) &= A\hat{x}(s) + B\hat{u}(s), \\ \hat{y}(s) &= B^T\hat{x}(s), \end{aligned} \quad (13)$$

where $s \in \mathbb{C}$.

From now on, we assume that the *matrix pencil* $sE - A$ is *regular*, i.e., the matrix $sE - A$ is singular only for finitely many values of $s \in \mathbb{C}$. We can then eliminate $\hat{x}(s)$ from (13) to obtain the input-output relation

$$\hat{y}(s) = H(s)\hat{u}(s), \quad \text{where} \quad H(s) := B^T(sE - A)^{-1}B. \quad (14)$$

The function H is called the *transfer function* of the descriptor system (8). We remark that H is a rational $m \times m$ -matrix-valued function and that the potential poles of H are the values of $s \in \mathbb{C}$ for which the matrix $sE - A$ is singular. For RCL networks with only current sources, $\hat{u}(s)$ and $\hat{y}(s)$ are the input currents and output voltages of these sources, and thus $H(s)$ is the *impedance matrix* of the network. Similarly, $H(s)$ is the *admittance matrix* for RCL networks with only voltage sources. For RCL networks with both voltage and current sources, $H(s)$ is called a *hybrid matrix* of the network; see, e.g., [1, Section 2.4].

Finally, we remark that the regularity assumption for the matrix pencil $sE - A$ is satisfied for any realistic RCL network. Indeed, $sE - A$ is regular if, and only if, the

subnetwork consisting of only the voltage sources of the RCL network has no closed (undirected) loops and the (undirected) graph corresponding to the subnetwork obtained from the RCL network by deleting all its current sources is connected; see, e.g., [20, Theorem 1].

2.3 Passivity

A physical system is said to be *passive* if it only consumes energy. Clearly, any RCL network is passive.

It is well known [1] that passivity of a descriptor system (8) is equivalent to positive realness of its associated transfer function (14), H . Here, a transfer function H is said to be *positive real* if it has no poles in the right half

$$\mathbb{C}_+ := \{s \in \mathbb{C} \mid \operatorname{Re} s > 0\}$$

of the complex plane and

$$H(s) + (H(s))^H \succeq 0 \quad \text{for all } s \in \mathbb{C}_+.$$

Since RCL networks are passive, the associated transfer function H in (14) is positive real. This fact can also be deduced directly from the semidefiniteness conditions (12) by employing the following result, which can be found as Theorem 13 in [16].

Theorem 1. *Let $A, E \in \mathbb{R}^{N \times N}$ and $B \in \mathbb{R}^{N \times m}$ be given matrices, and assume that $A + A^T \preceq 0$, $E = E^T \succeq 0$, and that the matrix pencil $sE - A$ is regular. Then, the function*

$$H(s) := B^T (sE - A)^{-1} B$$

is positive real.

2.4 Reciprocity

A second important property of RCL networks is *reciprocity*, which represents a certain symmetry of the input-output behavior of the network; see, e.g., [36] and [1, Section 2.8]. For RCL networks with only current sources, reciprocity is equivalent to the symmetry of the impedance matrix $H(s)$, i.e., $H(s) = H(s)^T$ for all $s \in \mathbb{C}$. Similarly, reciprocity is equivalent to the symmetry of the admittance matrix $H(s)$ for RCL networks with only voltage sources.

For general RCL networks with m_i current and m_v voltages sources, reciprocity is equivalent to the transfer function $H(s)$ being symmetric with respect to the *signature matrix*

$$\Sigma := \begin{bmatrix} I_{m_i} & 0 \\ 0 & -I_{m_v} \end{bmatrix} \in \mathbb{R}^{m \times m}, \quad m = m_i + m_v. \quad (15)$$

Here, H is said to be *symmetric with respect to* Σ if $H(s)\Sigma = \Sigma(H(s))^T$ for all $s \in \mathbb{C}$. Note that the definition (15) of the signature matrix assumes the ordering (9) of the input and output vectors u and y . If a different ordering for u and y is used, then the diagonal entries of Σ need to be permuted accordingly.

It turns out that reciprocity follows easily from the block structure of the matrices A , E , and B defined in (10) and from the symmetry of the diagonal blocks of A and E . In fact, we have the following theorem the proof of which is straightforward and left as an exercise to the reader.

Theorem 2. *Let $A, E \in \mathbb{R}^{N \times N}$ and $B \in \mathbb{R}^{N \times m}$ be matrices with block structures of the form*

$$A = \begin{bmatrix} A_{11} & -A_{12} & -A_{13} \\ A_{12}^T & 0 & 0 \\ A_{13}^T & 0 & 0 \end{bmatrix}, \quad E = \begin{bmatrix} E_{11} & 0 & 0 \\ 0 & E_{22} & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} B_{11} & 0 \\ 0 & 0 \\ 0 & B_{32} \end{bmatrix},$$

where the sizes of the blocks are the same as in (10) and

$$A_{11} = A_{11}^T, \quad E_{11} = E_{11}^T, \quad E_{22} = E_{22}^T.$$

Then, the function

$$H(s) := B^T (sE - A)^{-1} B$$

is symmetric with respect to the signature matrix Σ defined in (15).

3 A Brief Review of Krylov Subspace-Based Model Order Reduction

In this section, we present a brief review of the evolution of Krylov subspace-based model order reduction from moment matching to the currently dominating paradigm of projection onto Krylov subspaces.

3.1 Moment Matching and Padé Approximation

The Elmore delay [11] is a classical, simple metric for estimating signal delays in electronic networks. In 1990, *Asymptotic Waveform Evaluation (AWE)* [31] was introduced as a generalization of the Elmore delay to obtain more accurate timing estimates for the large RCL networks arising in VLSI interconnect modeling; see, e.g., [6].

To illustrate the AWE approach, we consider the special case of RC networks driven by a single voltage source. The transfer function H is then scalar-valued and represents the admittance of the RC network. The Taylor expansion of H about $s_0 = 0$ is given by

$$H(s) = \mu_0 + \mu_1 s + \mu_2 s^2 + \cdots + \mu_j s^j + \cdots, \quad (16)$$

where the μ_j 's are the so-called *moments*. Suppose we are trying to approximate H by a rational function H_1 of the form

$$H_1(s) = \frac{a_1}{s - b_1}, \quad \text{where } a_1, b_1 \in \mathbb{R}. \quad (17)$$

Given the first two moments, μ_0 and μ_1 , of the transfer function (16), H , we can determine values of the parameters a_1, b_1 in (17) such that the Taylor expansions of H_1 and H about $s_0 = 0$ agree in the first two moments, i.e.,

$$H_1(s) = H(s) + \mathcal{O}(s^2). \quad (18)$$

The last condition is satisfied if, and only if, $a_1 = -\mu_0^2/\mu_1$ and $b_1 = \mu_0/\mu_1$. Moreover, one can show that $\mu_0 > 0$ and $\mu_1 < 0$, and thus $a_1 > 0$ and $b_1 < 0$. It follows that

$$R := -\frac{b_1}{a_1} > 0 \quad \text{and} \quad C := \frac{a_1}{b_1^2} > 0.$$

Furthermore, it is easy to verify that for these values of R and C , the approximate transfer function (17), H_1 , is the admittance of the simple RC network shown in Figure 1. The impulse response of this network is given by $i(t) = i(0) \exp(-t/\tau)$, where $\tau := RC$ is the Elmore delay of the original given RC network. Note that the

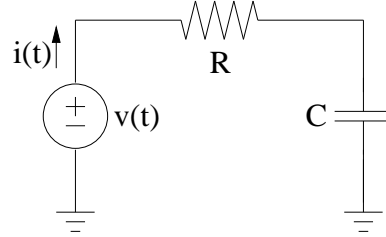


Fig. 1 Synthesis of the approximate transfer function (17), H_1 , as an RC network

RC network in Figure 1 represents a synthesis of the approximation defined by the moment-matching property (18).

AWE generalizes the ‘one-pole’ approximation (17) of the transfer function (16), H , of the given RC network to a general approximation H_n with n poles. More precisely, H_n is of the form

$$H_n(s) = \frac{a_1}{s - b_1} + \frac{a_2}{s - b_2} + \cdots + \frac{a_n}{s - b_n}, \quad (19)$$

where the $2n$ free parameters a_1, a_2, \dots, a_n and b_1, b_2, \dots, b_n are now chosen such that the Taylor expansions of H_n and H about $s_0 = 0$ agree in the first $2n$ moments, i.e.,

$$H_n(s) = H(s) + \mathcal{O}(s^{2n}). \quad (20)$$

The property (20) is often referred to as *moment matching*. The corresponding transfer function H_n is an optimal approximation to the original transfer function H in the sense that among all functions of the form (19), H_n matches the maximum possible number of moments. Such optimal approximations are called *Padé approximants* [5].

There are two major issues with the original AWE approach. First, the computation of H_n via explicitly generating the $2n$ moments $\mu_0, \mu_1, \dots, \mu_{2n-1}$ is extremely sensitive to numerical round-off errors and thus is viable only for very small values of n , typically $n \leq 10$; we refer the reader to [12, 13] for a discussion of this issue and for numerical examples. The problem can be resolved by generating the necessary moment information implicitly, by exploiting the connection of moment matching and Krylov subspace methods; see Section 3.3.

The second, more fundamental issue is that for general RCL networks, reduced-order transfer functions defined via Padé approximation do not preserve essential properties, such as passivity and reciprocity, of the original network. In fact, preservation of these properties can only be guaranteed for the special cases of RC, RL, and LC networks; see [22, 4, 23]. For general RCL networks, this issue can be resolved by relaxing the moment-matching property and using certain *Padé-type approximants*, instead of Padé approximants; see Section 3.5.

3.2 Reduced-Order Models

We now return to the case of general RCL networks with a total number of m independent current and voltage sources. Recall that in time domain, the RCL network equations can be stated as a descriptor system (8) with data matrices $A, E \in \mathbb{R}^{N \times N}$ and $B \in \mathbb{R}^{N \times m}$ of the form (10), where N is the state-space dimension of the system (8). In Laplace domain, the RCL network is described by its transfer function (14), H .

A general *reduced-order model* of (8) is a descriptor system of the same form as (8), but with state-space dimension $n (< N)$, instead of N . Thus a reduced-order model is of the form

$$\begin{aligned} E_n \frac{d}{dt} x_n(t) &= A_n x_n(t) + B_n u(t), \\ y_n(t) &= B_n^T x_n(t), \end{aligned} \quad (21)$$

where $A_n, E_n \in \mathbb{R}^{n \times n}$ and $B_n \in \mathbb{R}^{n \times m}$. Note that the input vector $u(t)$ in (21) is the same as in (8). In particular, the number m is unchanged from (8). The output vector $y_n(t)$ of (21), however, is only an approximation to the original output vector

$y(t)$ of (8). In fact, the problem of model order reduction is to find a sufficiently large reduced state-space dimension n and matrices A_n , E_n , and B_n such that the output vector of the reduced-order model (21) is a ‘sufficiently good’ approximation to the output vector of the original system (8).

Provided that the matrix pencil

$$sE_n - A_n, \quad s \in \mathbb{C}, \quad (22)$$

associated with the reduced-order model (21) is regular, we have, in analogy to (14), the Laplace-domain input-output relation

$$\hat{y}_n(s) = H_n(s)\hat{u}(s), \quad \text{where} \quad H_n(s) := B_n^T (sE_n - A_n)^{-1} B_n, \quad (23)$$

for the reduced-order model (21). Together with (14), it follows that

$$\hat{y}_n(s) - \hat{y}(s) = (H_n(s) - H(s))\hat{u}(s).$$

This relation shows that in terms of transfer functions, the problem of model order reduction is to find a sufficiently large reduced state-space dimension n and matrices A_n , E_n , and B_n such that the transfer function (23), H_n , of the reduced-order model is a ‘sufficiently good’ approximation to the transfer function (14), H , of the original system:

$$H_n(s) \approx H(s) \quad \text{in ‘some sense’}.$$

There are many general-purpose methods for constructing suitable approximations H_n ; see, e.g., [3, 2]. For the special case of systems describing RCL networks, techniques based on moment matching are the most widely used and in fact, for very large N , are often the only computationally viable approaches.

3.3 Moment Matching Via Krylov Subspace Methods

Recall that for general RCL networks with a total number of m independent current and voltage sources, the transfer functions H and H_n are $m \times m$ -matrix-valued. We now consider the extension of the moment-matching property (20) to the case of such general RCL networks. Furthermore, instead of the expansion point $s_0 = 0$ in (20), we allow general expansion points s_0 subject to the constraint that the matrix $s_0 E - A$ is nonsingular. This condition guarantees that s_0 is not a pole of the transfer function (14), H . For now, we restrict ourselves to real expansion points $s_0 \in \mathbb{R}$; the case of general complex expansion points $s_0 \in \mathbb{C}$ is discussed in Section 7.

In this general setting, moment matching takes on the following form:

$$H_n(s) = H(s) + \mathcal{O}((s - s_0)^q). \quad (24)$$

Here, $q = q(n, m)$ is an integer that depends on the reduced state-space dimension n , on m , and on the particular method for constructing the reduced-order model associated with H_n . In this setting, the AWE approach is obtained by choosing a method that gives the maximum value of q in (24). The corresponding function H_n is then called an n -th *Padé approximant* of H and the system (21) a *Padé reduced-order model* of (8). We denote by $\tilde{q}(n, m)$ the maximum value of q in (24). It is well known (see, e.g., [14, 15]) that

$$\tilde{q}(n, m) \geq 2 \left\lfloor \frac{n}{m} \right\rfloor, \quad (25)$$

where equality holds true in the generic case.

As mentioned before, generating Padé reduced-order models directly from the first $\tilde{q} = \tilde{q}(n, m)$ moments μ_j , $j = 0, 1, \dots, \tilde{q} - 1$, in the expansion

$$H(s) = \mu_0 + \mu_1(s - s_0) + \mu_2(s - s_0)^2 + \dots + \mu_{\tilde{q}-1}(s - s_0)^{\tilde{q}-1} + \dots \quad (26)$$

is extremely sensitive to numerical round-off errors. The remedy is to avoid the computation of the moments and instead, to generate the information contained in the moments via Krylov subspace techniques. To this end, we rewrite the transfer function (14), H , as follows:

$$\begin{aligned} H(s) &= B^T (s_0 E - A + (s - s_0)E)^{-1} B = B^T (I - (s - s_0)M)^{-1} R \\ &= \sum_{j=0}^{\infty} B^T M^j R (s - s_0)^j, \end{aligned} \quad (27)$$

where

$$M := -(s_0 E - A)^{-1} E \quad \text{and} \quad R := (s_0 E - A)^{-1} B. \quad (28)$$

Comparing (26) and (27), it follows that

$$\mu_j = B^T M^j R = ((M^T)^j B)^T R, \quad j = 0, 1, \dots \quad (29)$$

The representation (29) shows that the leading coefficients in the expansion (26) of H can be obtained by computing suitable inner products of the leading columns of the matrices

$$[R \quad MR \quad M^2 R \quad \dots \quad M^{j-1} R \quad \dots] \quad (30)$$

and

$$[B \quad M^T B \quad (M^T)^2 B \quad \dots \quad (M^T)^{j-1} B \quad \dots]. \quad (31)$$

Working directly with the columns of these matrices would result in a procedure that is still sensitive to numerical round-off errors. Instead, Krylov subspace methods generate more suitable basis vectors for the subspaces spanned by the leading columns of these matrices. Computing moment information via inner products of these basis vectors then avoids the numerical issues of moment matching and leads

to algorithms that generate Padé reduced-order models in a numerically stable and reliable way; see [12, 13, 14, 25, 17] for more details.

For later use, we give a formal definition of the Krylov subspaces associated with the matrices (30) and (31). Let N_{\max} ($\leq N$) denote the rank of the matrix (30). Then, for $\hat{n} = 1, 2, \dots, N_{\max}$, the \hat{n} -th (right) *Krylov subspace* (induced by M and R) is defined as the \hat{n} -dimensional subspace of \mathbb{R}^N spanned by the first \hat{n} linearly independent columns of the matrix (30). In the following, $\mathcal{K}_{\hat{n}}(M, R)$ denotes this \hat{n} -th Krylov subspace. Similarly, the \hat{n} -th (left) Krylov subspace, $\mathcal{K}_{\hat{n}}(M^T, B)$, is spanned by the first \hat{n} linearly independent columns of the matrix (31).

An important issue in moment matching is the choice of the expansion point s_0 in (24). Typically, the goal is to construct a reduced-order model (21) such that its transfer function H_n approximates the original transfer function H well enough in some given frequency range of interest of the form

$$s = \mathbf{i}\omega, \quad \omega_{\min} \leq \omega \leq \omega_{\max},$$

where $0 \leq \omega_{\min} < \omega_{\max}$ are given and $\mathbf{i} = \sqrt{-1}$. Moreover, the state-space dimension n of the reduced-order model should be as small as possible. The general convergence theory of Padé approximation suggests to place s_0 close to the frequency range of interest, yet at a safe distance from the poles of the original transfer function H . Recall from Section 2.3 that the poles of H are to the left of or on the imaginary axis. Therefore, the expansion point s_0 is placed in the right half \mathbb{C}_+ of the complex plane. The ideal placement of s_0 would be on a horizontal line through the midpoint of the frequency range of interest, resulting in a non-real s_0 . This in turn would result in complex matrices M and R in (28) and thus the need for complex rather than real arithmetic, which would increase the computational work by roughly a factor of 4. This is the reason for the usual restriction to positive $s_0 > 0$; see, e.g., the discussion in [13]. A typical placement of $s_0 > 0$ relative to the frequency range of interest is shown in Figure 2.

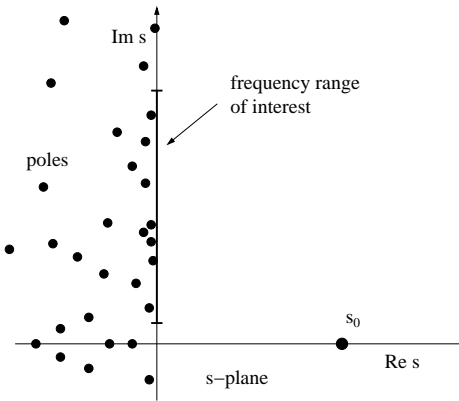


Fig. 2 Typical placement of expansion point s_0 for the moment-matching property (24)

3.4 Passive Models Via Projection

While Padé reduced-order models are optimal in terms of moment matching, they do not preserve the passivity of general RCL networks.

A very basic, general approach to constructing passive reduced-order models is to employ *projection*. Let

$$V_n \in \mathbb{R}^{N \times n} \quad \text{with} \quad \text{rank } V_n = n \quad (32)$$

be given. Then, by simply setting

$$A_n := V_n^T A V_n, \quad E_n := V_n^T E V_n, \quad \text{and} \quad B_n := V_n^T B, \quad (33)$$

one obtains a reduced-order model (21) that can be viewed as a projection of the N -dimensional state space of the original system onto the n -dimensional subspace spanned by the columns of the matrix V_n . In particular, projection employs an ansatz of the form

$$x_n(t) = V_n x(t)$$

for the state-space vector $x_n(t)$ of the reduced-order model (21). Recall from (9) that $x(t)$ denotes the state-space vector of the original descriptor system (8).

Reduced-order models obtained by means of this simple projection approach trivially preserve passivity of the original system; see, e.g., [28, 29, 30, 16]. Indeed, the only additional condition on the matrix (32), V_n , is that the resulting matrix pencil (22) is regular. Recall that A and E satisfy the semidefiniteness properties (12). The definitions of A_n and E_n in (33) readily imply that these matrices satisfy the very same semidefiniteness conditions. Therefore, by Theorem 1, the transfer function (23), H_n is positive real, and thus the corresponding reduced-order model (21) is passive.

3.5 Projection Combined With Krylov Subspaces

The projection approach described in Section 3.4 can be combined with the use of Krylov subspaces to obtain reduced-order models that are passive and at the same time satisfy a moment-matching property, albeit a weaker one than Padé models.

To this end, we choose the projection matrix (32), V_n , such that

$$\mathcal{K}_{\hat{n}}(M, R) \subseteq \text{colspan } V_n. \quad (34)$$

Recall that $\mathcal{K}_{\hat{n}}(M, R)$ is the \hat{n} -th (right) Krylov subspace defined in Section 3.3. Since $\mathcal{K}_{\hat{n}}(M, R)$ has dimension \hat{n} and, by (32), V_n is assumed to have rank n , the condition (34) implies that $\hat{n} \leq n$. Note that $\hat{n} = n$ if, and only if, equality holds true in (34). In this case, the projection matrix (32), $V_{\hat{n}}$, is a *basis matrix* for $\mathcal{K}_{\hat{n}}(M, R)$, i.e., the columns of $V_{\hat{n}}$ form a basis for the linear subspace $\mathcal{K}_{\hat{n}}(M, R)$. We will use

the notation $\hat{V}_{\hat{n}}$, instead of $V_{\hat{n}}$, to indicate that $\hat{V}_{\hat{n}}$ is a basis matrix. We remark that there are standard Krylov subspace techniques, such as the band Arnoldi process [16, Algorithm 1], to construct the columns of $\hat{V}_{\hat{n}}$.

We now return to the general case (34). Let A_n , E_n , and B_n be the matrices defined by the projection approach (33), and let H_n be the transfer function (23) of the corresponding reduced-order model (21). The main result of Krylov subspace-based projection then states that H_n satisfies the moment-matching property

$$H_n(s) = H(s) + \mathcal{O}((s - s_0)^{\hat{q}(\hat{n}, m)}), \quad \text{where} \quad \hat{q}(\hat{n}, m) \geq \left\lfloor \frac{\hat{n}}{m} \right\rfloor. \quad (35)$$

Moreover, in the generic case, $\hat{q}(\hat{n}, m) = \lfloor \hat{n}/m \rfloor$ in (35). Recall from (24) and (25) that Padé reduced-order models match $\tilde{q}(n, m)$ moments, where

$$\tilde{q}(n, m) \geq 2 \left\lfloor \frac{n}{m} \right\rfloor \geq 2 \left\lfloor \frac{\hat{n}}{m} \right\rfloor.$$

Thus, Krylov subspace-based projection results in reduced-order models that, in the generic case, match only half as many moments as Padé models if $\hat{n} = n$ and even less if $\hat{n} < n$. Reduced-order transfer functions H_n that match fewer moments than n -th Padé approximants are called *n-th Padé-type approximants* and the system (21) is a *Padé-type reduced-order model* of (8).

The Padé-type moment matching property (35) of Krylov subspace-based projection is well known. For example, it was established for various special cases in [8, 28, 25]. Proofs for the general case can be found in [16, 19].

4 PRIMA

The simplest way to satisfy condition (34) is to choose the projection matrix (32) as a basis matrix for $\mathcal{K}_{\hat{n}}(M, R)$. In this case $\hat{n} = n$, and the moment-matching property (35) takes on the following form:

$$H_{\hat{n}}(s) = H(s) + \mathcal{O}((s - s_0)^{\hat{q}(\hat{n}, m)}), \quad \text{where} \quad \hat{q}(\hat{n}, m) \geq \left\lfloor \frac{\hat{n}}{m} \right\rfloor. \quad (36)$$

Moreover, in the generic case, $\hat{q}(\hat{n}, m) = \lfloor \hat{n}/m \rfloor$. The corresponding reduced-order models (21) given by the reduced data matrices (33) (where $n = \hat{n}$) are then the ones produced by PRIMA (Passive Reduced-order Interconnect Macromodeling Algorithm) [29, 30].

Next, we present an algorithmic statement of the basic steps of PRIMA.

Algorithm 3 (PRIMA for general RCL networks)

- **Input:** Matrices of the form

$$A = \begin{bmatrix} A_{11} & -A_l & -A_v \\ \mathcal{A}_l^T & 0 & 0 \\ \mathcal{A}_v^T & 0 & 0 \end{bmatrix}, \quad E = \begin{bmatrix} E_{11} & 0 & 0 \\ 0 & L & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} \mathcal{A}_i & 0 \\ 0 & 0 \\ 0 & -I \end{bmatrix},$$

where $A_{11} \preceq 0$, $E_{11} \succeq 0$, and $L \succ 0$.

An expansion point $s_0 \in \mathbb{R}$ such that the matrix $s_0 E - A$ is nonsingular; this last condition is satisfied for any $s_0 > 0$.

- Formally set

$$M = -(s_0 E - A)^{-1} E, \quad R = (s_0 E - A)^{-1} B.$$

- Until \hat{n} is large enough, run your favorite Krylov subspace method (applied to M and R) to construct a matrix

$$\hat{V}_{\hat{n}} = [v_1 \quad v_2 \quad \cdots \quad v_{\hat{n}}]$$

the columns of which span the \hat{n} -th Krylov subspace $\mathcal{K}_{\hat{n}}(M, R)$, i.e.,

$$\text{colspan } \hat{V}_{\hat{n}} = \mathcal{K}_{\hat{n}}(M, R).$$

- Set

$$A_{\hat{n}} = \hat{V}_{\hat{n}}^T A \hat{V}_{\hat{n}}, \quad E_{\hat{n}} = \hat{V}_{\hat{n}}^T E \hat{V}_{\hat{n}}, \quad \text{and} \quad B_{\hat{n}} = \hat{V}_{\hat{n}}^T B.$$

- **Output:** The data matrices $A_{\hat{n}}$, $E_{\hat{n}}$, and $B_{\hat{n}}$ of the PRIMA reduced-order model

$$\begin{aligned} E_{\hat{n}} \frac{d}{dt} x_{\hat{n}}(t) &= A_{\hat{n}} x_{\hat{n}}(t) + B_{\hat{n}} u(t), \\ y_{\hat{n}}(t) &= B_{\hat{n}}^T x_{\hat{n}}(t). \end{aligned} \tag{37}$$

Since the PRIMA reduced-order models (37) are generated via projection onto Krylov subspaces, they satisfy the moment-matching property (36) and they are guaranteed to be passive. In general, however, the PRIMA models do not satisfy reciprocity [34] and thus cannot be synthesized as RCL networks. Furthermore, the reduced-order data matrices $A_{\hat{n}}$, $E_{\hat{n}}$, and $B_{\hat{n}}$ are dense in general and thus do not preserve the block structures of the original data matrices A , E , and B .

5 SPRIM

In this section, we describe the SPRIM (Structure-Preserving Reduced-order Interconnect Macromodeling) algorithm [18, 21], which remedies the lack of structure preservation of PRIMA.

5.1 Preserving Block Structures

As in PRIMA, the main computational step of SPRIM is the generation of a basis matrix $\hat{V}_{\hat{n}}$ for the \hat{n} -th block Krylov subspace $\mathcal{K}_{\hat{n}}(M, R)$. In contrast to PRIMA, however, SPRIM does not use the matrix $\hat{V}_{\hat{n}}$ directly for the projection of the original data matrices. Instead, SPRIM employs a modified version of this matrix that trivially leads to structure preservation. To this end, $\hat{V}_{\hat{n}}$ is first partitioned as follows:

$$\hat{V}_{\hat{n}} = \begin{bmatrix} V^{(1)} \\ V^{(2)} \\ V^{(3)} \end{bmatrix}. \quad (38)$$

Here, the block sizes correspond to the block sizes of the original data matrices A and E in (10). While $\hat{V}_{\hat{n}}$ has full column rank \hat{n} , the same is not necessarily true for the three subblocks $V^{(l)}$, $l = 1, 2, 3$, in (38). In particular, the third block, $V^{(3)}$, is of size $m_v \times \hat{n}$, where m_v denotes the number of voltage sources of the given RCL circuit. Usually, the number m_v is very small and $m_v < \hat{n}$. Therefore, $V^{(3)}$ typically does not have full column rank. In the actual implementation of SPRIM, we run a Gram-Schmidt algorithm on the rows of $V^{(3)}$ to determine a matrix $\tilde{V}^{(3)}$ the columns of which span the same space as the columns of $V^{(3)}$, but which has full column rank. The other two blocks usually have many more rows than columns, and these blocks are unlikely not to have full column rank. In the implementation of SPRIM, there is also the option to check the column ranks of the first two blocks and replace them by matrices $\tilde{V}^{(1)}$ and $\tilde{V}^{(2)}$ of full column rank. Next, we set up the actual projection matrix V_n as follows:

$$V_n := \begin{bmatrix} \tilde{V}^{(1)} & 0 & 0 \\ 0 & \tilde{V}^{(2)} & 0 \\ 0 & 0 & \tilde{V}^{(3)} \end{bmatrix}. \quad (39)$$

By construction, we have

$$\mathcal{K}_{\hat{n}}(M, R) = \text{colspan } \hat{V}_{\hat{n}} \subseteq \text{colspan } V_n. \quad (40)$$

Thus the matrix (39), V_n , satisfies condition (34), which in turn guarantees the moment-matching property (35). Furthermore, in view of the block structure of V_n , the data matrices (33) of the resulting reduced-order model obtained via projection with V_n have the same block structure as the original data matrices A , E , and B .

5.2 The Algorithm

An algorithmic statement of the basic steps of SPRIM is as follows.

Algorithm 4 (SPRIM for general RCL networks)

- **Input:** Matrices of the form

$$A = \begin{bmatrix} A_{11} & -\mathcal{A}_l & -\mathcal{A}_v \\ \mathcal{A}_l^T & 0 & 0 \\ \mathcal{A}_v^T & 0 & 0 \end{bmatrix}, \quad E = \begin{bmatrix} E_{11} & 0 & 0 \\ 0 & L & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} \mathcal{A}_i & 0 \\ 0 & 0 \\ 0 & -I \end{bmatrix},$$

where $A_{11} \preceq 0$, $E_{11} \succeq 0$, and $L \succ 0$.

An expansion point $s_0 \in \mathbb{R}$ such that the matrix $s_0 E - A$ is nonsingular; this last condition is satisfied for any $s_0 > 0$.

- Formally set

$$M = -(s_0 E - A)^{-1} E, \quad R = (s_0 E - A)^{-1} B.$$

- Until \hat{n} is large enough, run your favorite Krylov subspace method (applied to M and R) to construct a matrix

$$V_{\hat{n}} = [v_1 \quad v_2 \quad \cdots \quad v_{\hat{n}}]$$

the columns of which span the \hat{n} -th Krylov subspace $\mathcal{K}_{\hat{n}}(M, R)$, i.e.,

$$\text{colspan } \hat{V}_{\hat{n}} = \mathcal{K}_{\hat{n}}(M, R).$$

- Let

$$\hat{V}_{\hat{n}} = \begin{bmatrix} V^{(1)} \\ V^{(2)} \\ V^{(3)} \end{bmatrix}$$

be the partitioning of $\hat{V}_{\hat{n}}$ corresponding to the block sizes of A and E .

- For $l = 1, 2, 3$ do:

If $r_l := \text{rank } V^{(l)} < \hat{n}$, determine an $N \times r_l$ matrix $\tilde{V}^{(l)}$ with

$$\text{colspan } \tilde{V}^{(l)} = \text{colspan } V^{(l)} \quad \text{and} \quad \text{rank } \tilde{V}^{(l)} = r_l.$$

- Set

$$\tilde{A}_{11} = (\tilde{V}^{(1)})^T A_{11} \tilde{V}^{(1)}, \quad \tilde{\mathcal{A}}_l = (\tilde{V}^{(1)})^T \mathcal{A}_l \tilde{V}^{(2)}, \quad \tilde{\mathcal{A}}_v = (\tilde{V}^{(1)})^T \mathcal{A}_v \tilde{V}^{(3)},$$

$$\tilde{E}_{11} = (\tilde{V}^{(1)})^T E_{11} \tilde{V}^{(1)}, \quad \tilde{L} = (\tilde{V}^{(2)})^T L \tilde{V}^{(2)}, \quad \tilde{\mathcal{A}}_i = (\tilde{V}^{(1)})^T \mathcal{A}_i.$$

- **Output:** The data matrices

$$A_n = \begin{bmatrix} \tilde{A}_{11} & -\tilde{A}_l & -\tilde{A}_v \\ \tilde{A}_l^T & 0 & 0 \\ \tilde{A}_v^T & 0 & 0 \end{bmatrix}, \quad E_n = \begin{bmatrix} \tilde{E}_{11} & 0 & 0 \\ 0 & \tilde{L} & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

$$\text{and } B_n = \begin{bmatrix} \tilde{A}_i & 0 \\ 0 & 0 \\ 0 & -(\tilde{V}^{(3)})^T \end{bmatrix}$$

of the SPRIM reduced-order model

$$\begin{aligned} E_n \frac{d}{dt} x_n(t) &= A_n x_n(t) + B_n u(t), \\ y_n(t) &= B_n^T x_n(t). \end{aligned} \quad (41)$$

5.3 Some Properties

In this subsection, we list some of the properties of the SPRIM Algorithm 4.

Since the SPRIM reduced-order models (41) are generated via projection onto Krylov subspaces, they are guaranteed to be passive and they satisfy the moment-matching property (35). In contrast to the PRIMA models, the data matrices A_n , E_n , and B_n of the SPRIM reduced-order models exhibit the same block structures as the original data matrices A , E , and B . In particular, the conditions of Theorem 2 hold true for the matrices A_n , E_n , and B_n , and consequently, the SPRIM reduced-order models are guaranteed to satisfy reciprocity. While reciprocity is not sufficient to ensure the synthesizability of reduced-order models as RCL networks, it significantly simplifies the synthesis of the models as simple electronic networks; see [1].

Preservation of the block structures of the data matrices also increases the accuracy of the moment-matching property of the SPRIM models. While the theory of projection onto Krylov subspace guarantees (35), the SPRIM reduced-order models satisfy the following stronger moment-matching property:

$$H_n(s) = H(s) + \mathcal{O}((s - s_0)^{2\hat{q}(\hat{n}, m)}), \quad \text{where } \hat{q}(\hat{n}, m) \geq \left\lfloor \frac{\hat{n}}{m} \right\rfloor. \quad (42)$$

The integer $\hat{q}(\hat{n}, m)$ is the same as in (35) and (36), and in the generic case, $\hat{q}(\hat{n}, m) = \lfloor \hat{n}/m \rfloor$ in (42). The property means that at the expansion point s_0 , the transfer function of the SPRIM reduced-order model matches twice as many moments as the theory of general Krylov subspace-based projection methods predicts. The reason for this increased accuracy is a certain J -symmetry, which follows from the block structures of the data matrices of the SPRIM models and results in a dou-

bling of the number of matched moments; for details and proofs, we refer the reader to [19].

5.4 Pros and Cons of PRIMA and SPRIM

The underlying Krylov subspace for both PRIMA and SPRIM is $\mathcal{K}_{\hat{n}}(M, R)$, and generating a basis matrix $\hat{V}_{\hat{n}}$ for this subspace is the dominant computational work for both algorithms. PRIMA uses $\hat{V}_{\hat{n}}$ directly to generate a reduced-order model of state-space dimension \hat{n} that matches $\hat{q}(\hat{n}, m) \geq \lfloor \hat{n}/m \rfloor$ moments and is passive. SPRIM first processes the blocks of $\hat{V}_{\hat{n}}$ to obtain the matrix (39), V_n . This matrix is then used to generate a reduced-order model of state-space dimension n that matches $2\hat{q}(\hat{n}, m) \geq \lfloor \hat{n}/m \rfloor$ moments, is passive, and satisfies reciprocity. Here, n is the number of columns of V_n . Since each of the three diagonal blocks $\hat{V}^{(l)}$, $l = 1, 2, 3$, of V_n has at most \hat{n} columns and $\hat{V}^{(3)}$ has at most m_v columns, n is bounded as follows:

$$n \leq 2\hat{n} + \min\{\hat{n}, m_v\}.$$

Therefore, for the same \hat{n} , PRIMA generates smaller reduced-order models than SPRIM. On the other hand, the SPRIM models satisfy reciprocity, which simplifies their synthesis as electrical networks, and in terms of moment matching, the SPRIM models are twice as accurate as the PRIMA models.

In practice, both PRIMA and SPRIM are run as iterative algorithms with iteration counter \hat{n} . The iterative process is stopped once \hat{n} is large enough so that the reduced-order transfer function has converged to the original transfer function throughout the given frequency range of interest. Due to its higher accuracy, SPRIM will typically converge faster, i.e., for a smaller \hat{n} , than PRIMA. The following example, which is taken from [21], illustrates the difference in convergence behavior between SPRIM and PRIMA. The example (referred to as “package example”) is an RCL network with state-space dimension $N = 1841$. The network has $m_i = 8$ current sources and $m_v = 8$ voltage sources, and thus $m = m_i + m_v = 16$. Its transfer function is 16×16 -matrix valued and has 256 components. The expansion point $s_0 = 2\pi \times 10^{10}$ was used. For this example, $\hat{n} = 128$ was needed for SPRIM to achieve convergence. The corresponding state-space dimension of the SPRIM reduced-order model is $n = 2\hat{n} + m_v = 264$. Figure 5.4 depicts the absolute values of the (8, 1)-component and the (9, 9)-component of the transfer functions. Note that for $\hat{n} = 128$ PRIMA has not converged yet.

6 Thick-Restart Krylov Subspace Techniques

While Krylov subspace-projection algorithms, such as PRIMA and SPRIM, generate reduced-order models that are passive and satisfy a moment-matching property,

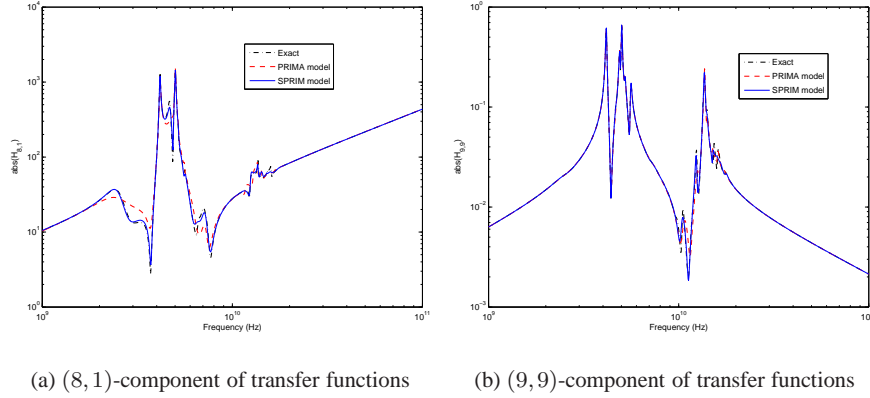


Fig. 3 Absolute values of transfer functions for package example

their use becomes prohibitive for RCL networks with ‘very’ large state-space dimension N . The computational costs of these methods is dominated by the generation of suitable basis vectors for Krylov subspaces $\mathcal{K}_n(M, R)$ of dimension n . For general RCL networks, the work for this task is proportional to n^2N . As a result, for very large N , it is typically not feasible to run the Krylov subspace method for a value of n that is large enough to produce a reduced-order model of sufficient accuracy.

A standard approach in numerical linear algebra to reduce the computational cost from $\mathcal{O}(n^2N)$ to $\mathcal{O}(n_0lN)$ is to restart the Krylov subspace method after each cycle of n_0 steps and run l of these restarted cycles. The most common restarted Krylov subspace techniques are designed for matrix computations that are described by M and for which the starting block R can be chosen freely. In model order reduction, however, R is determined by the given data matrices of the original system, and thus R cannot be chosen freely.

Recently, there has been a lot of interest in so-called *thick-restart* Krylov subspace techniques (see, e.g., [10] and the references therein), which, in contrast to standard restart methods, can be adapted to the problem of model order reduction. The basic idea is that after each cycle of n_0 steps, ‘relevant’ information is extracted from the batch of basis vectors generated in this cycle and the extracted vectors are then used as starting vectors in the next cycle.

For model order reduction, ‘relevant’ information means converged eigenvectors corresponding to poles of the original transfer function (14), H , that are close to the frequency range of interest; cf. Figure 2. Recall from (14) that any pole $\mu \in \mathbb{C}$ of H is a generalized eigenvalue of the matrix pencil $sE - A$; i.e., there exists a generalized eigenvector $v \in \mathbb{C}^N$ such that

$$Av = \mu Ev, \quad v \neq 0. \quad (43)$$

The Krylov subspace method is applied to the matrix $M = -(s_0 E - A)^{-1} E$. It is straightforward to verify that the vector v satisfies the relation (43) if, and only if, v is an eigenvector of M , i.e.,

$$Mv = \lambda v, \quad v \neq 0, \quad (44)$$

where the eigenvalues μ and λ are connected as follows:

$$\mu = s_0 + \frac{1}{\lambda}. \quad (45)$$

It is well known that Krylov subspace methods (applied to M) will generate eigenvectors for the dominant eigenvalues λ of M within a very small number of iterations. Here, ‘dominant’ means ‘largest in absolute value’. In view of (45), the dominant eigenvalues λ of M correspond to the poles μ of H closest to the expansion point s_0 . This means that by placing s_0 close to the frequency range of interest, we can expect to obtain generalized eigenvectors associated with a few converged poles of H close to the frequency range of interest. These generalized eigenvectors are then used for the thick restart in the next cycle of n_0 steps of the Krylov subspace method. The same process is repeated in such a way that in each cycle, one obtains additional converged poles of H close to the frequency range of interest. However, since poles closest to the expansion point converge fastest, this thick-restart process has to allow for changing expansion points, so that in each cycle new additional poles of H can be found. We denote by $s_0^{(j)}$ the expansion point employed in the j -th cycle. Note that, except for the first point $s_0^{(1)}$, the expansion points need to be complex in order to stay close to the frequency range of interest; cf. Figure 4. A typical strategy for choosing the $s_0^{(j)}$ ’s is to move up parallel to the imaginary axis, as indicated in Figure 4 for the case $l = 3$.

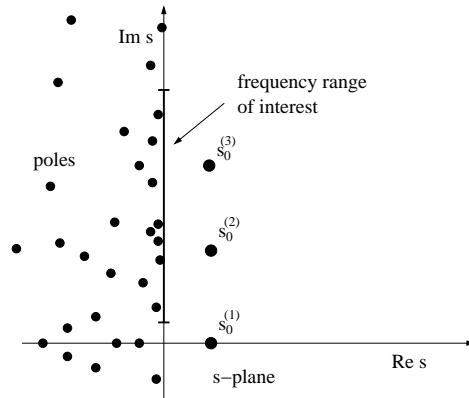


Fig. 4 Placement of multiple expansion points $s_0^{(1)}, s_0^{(2)}, \dots, s_0^{(l)}$

With changing values of $s_0^{(j)}$, the matrix M also changes within each cycle. We denote by

$$M^{(j)} = - \left(s_0^{(j)} E - A \right)^{-1} E$$

the matrix that the Krylov subspace method is applied to in the l -th cycle. It readily follows from the equivalence of (44) and (43) that any eigenvector v of $M^{(j)}$ is also an eigenvector of $M^{(j+1)}$, only the value of the corresponding eigenvalues are transformed according to the relation

$$s_0^{(j)} + \frac{1}{\lambda^{(j)}} = s_0^{(j+1)} + \frac{1}{\lambda^{(j+1)}}.$$

Due to this invariance of the eigenvectors of $M^{(j)}$, the information used in the thick restart remains relevant from cycle to cycle even though the matrix $M^{(j)}$ changes to $M^{(j+1)}$.

In the following algorithm, we outline the basic steps for generating a projection matrix V_n via the thick-restart Krylov subspace approach.

Algorithm 5 (Generation of projection matrix V_n via thick-restart Krylov subspace approach)

- **Input:** Matrices of the form

$$A = \begin{bmatrix} A_{11} & -A_l & -A_v \\ \mathcal{A}_l^T & 0 & 0 \\ \mathcal{A}_v^T & 0 & 0 \end{bmatrix}, \quad E = \begin{bmatrix} E_{11} & 0 & 0 \\ 0 & L & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} \mathcal{A}_i & 0 \\ 0 & 0 \\ 0 & -I \end{bmatrix},$$

where $A_{11} \preceq 0$, $E_{11} \succeq 0$, and $L \succ 0$.

An initial expansion point $s_0^{(1)} \in \mathbb{R}$ such that the matrix $s_0^{(1)} E - A$ is nonsingular; this last condition is satisfied for any $s_0^{(1)} > 0$.

Number $n_0 (> m)$ of iterations in each cycle of the Krylov subspace method.

Maximum number l_{\max} of thick restarts.

- For $l = 1, 2, \dots, l_{\max}$ do:

– Formally set

$$M^{(l)} = - \left(s_0^{(l)} E - A \right)^{-1} E, \quad R^{(l)} = \begin{cases} \left(s_0^{(l)} E - A \right)^{-1} B & \text{if } l = 1, \\ Y^{(l-1)} & \text{if } l > 1. \end{cases}$$

– Run n_0 steps of your favorite Krylov subspace method (applied to $M^{(l)}$ and $R^{(l)}$) to obtain an $n \times n_0$ matrix $V^{(l)}$ the columns of which span the Krylov subspace $\mathcal{K}_{n_0}(M^{(l)}, R^{(l)})$.

– If l is ‘large’ enough: stop.

– Extract ‘relevant’ eigenvector information $Y^{(l)}$ from $V^{(l)}$ to be used in the next thick restart.

– Select suitable next expansion point $s_0^{(l)} \in \mathbb{C}$. It needs to be such that the matrix $s_0^{(l)} E - A$ is nonsingular; this last condition is satisfied if $\operatorname{Re} s_0^{(l)} > 0$.

- **Output:** *Projection matrix*

$$V_n = [V^{(1)} \ V^{(2)} \ \dots \ V^{(l)}], \quad \text{where } n := n_0 l. \quad (46)$$

The matrix V_n produced by Algorithm 5 is then used to obtain a reduced-order model of state-space dimension n via projection by defining the reduced-order data matrices as follows:

$$A_n := V_n^H A V_n, \quad E_n := V_n^H E V_n, \quad \text{and} \quad B_n := V_n^H B. \quad (47)$$

Note that due to complex expansion points, the matrix V_n is complex in general, and thus A_n , E_n , and B_n are complex in general as well. In Section 7 below, we briefly discuss how to obtain real projection matrices even for complex expansion points.

Due to the use of the multiple expansion points $s_0^{(1)}, s_0^{(2)}, \dots, s_0^{(l)}$, the reduced-order model given by the data matrices (47) satisfies a multi-point moment-matching property of the form

$$H_n(s) = H(s) + \mathcal{O}\left((s - s_0^{(j)})^{q_j}\right), \quad j = 1, 2, \dots, l. \quad (48)$$

We remark that reduced-order models characterized by (48) can also be obtained by complex frequency hopping (CFH) [7]. However, CFH was proposed as a modification of AWE to allow changing expansion points, and CFH suffers from the same numerical problems as AWE due to the use of explicit moment computations.

Finally, we present a numerical example that illustrates the benefits of thick-restart Krylov subspace techniques for reduced-order modeling. With a single real expansion point, a reduced-order model of state-space dimension $n = 80$ is needed to achieve satisfactory convergence throughout the frequency range of interest; see Figure 5 (a). With thick restarts, three expansion points (one real, two complex) and $l = 3$ cycles of $n_0 = 14$ Krylov stops are sufficient to obtain a reduced-order model of state-space dimension $n = n_0 l = 42$ of comparable accuracy; see Figure 5 (b).

7 Complex Expansion Points

There are two main reasons why Krylov subspace-based model order reduction usually employs real expansion points s_0 . First, the matrices (28), M and R , to which the Krylov subspace method is applied are real, thus avoiding the use of complex arithmetic, which is 4 times as costly as real arithmetic. Second, when passive reduced-order models are constructed via the projection approach described in Section 3.4, the projection matrix V_n needs to be real in order to obtain real reduced-order data matrices (33). On the other hand, the use of complex expansion points s_0 typically results in a significantly smaller state-space dimension n of the reduced-order models, since s_0 can be placed closer to the frequency range of interest than

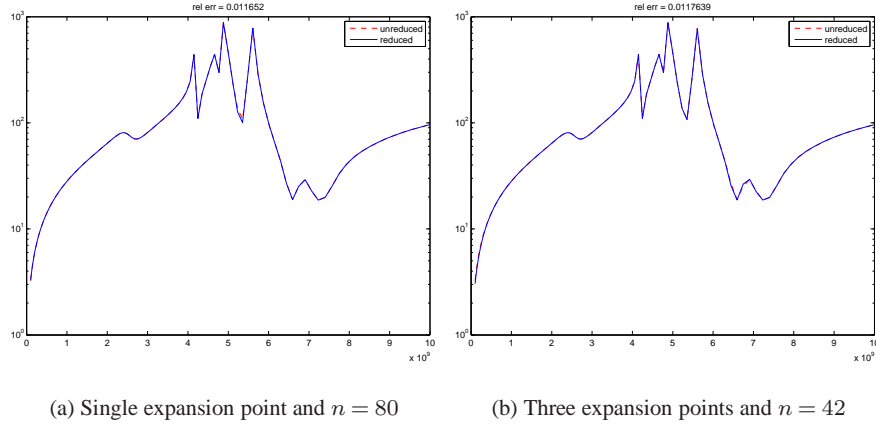


Fig. 5 Single expansion point vs. multiple expansion points and thick restarts

any real s_0 . Finally, as we discussed in Section 6, thick-restart Krylov subspace techniques only make sense when complex expansion points can be used.

We now consider the case that $s_0 \in \mathbb{C} \setminus \mathbb{R}$ is a non-real expansion point and that $V_n \in \mathbb{C}^{N \times n}$ is a basis matrix for the complex n -dimensional Krylov subspace $\mathcal{K}_n(M, R)$, where M and R are the complex matrices given by (28). Suppose we are trying to employ a projection approach similar to the one in Section 3.4 that produces real reduced-order data matrices. One possibility is to replace the complex matrix V_n by the real matrix

$$\begin{bmatrix} \operatorname{Re} V_n & \operatorname{Im} V_n \end{bmatrix}; \quad (49)$$

see [26, 27]. One obvious disadvantage of this approach is that the dimension of the resulting reduced-order model is doubled to $2n$. Furthermore, in general, the matrix (49) is not guaranteed to have full column rank, and so before using (49) as a projection matrix, one would need to check for and possibly delete any linearly dependent columns of (49) by means of some variant of a Gram-Schmidt orthogonalization process. On the other hand, the transfer function of the resulting reduced-order model will satisfy a two-point moment-matching property of the form

$$H_n = H(s) + \mathcal{O}((s - s_0)^{\hat{q}(n,m)}) \quad \text{and} \quad H_n = H(s) + \mathcal{O}((s - \overline{s_0})^{\hat{q}(n,m)}), \quad (50)$$

where $\hat{q}(n, m) \geq \lfloor n/m \rfloor$.

It turns out that the process of generating a real projection matrix by first computing a complex basis matrix V_n for $\mathcal{K}_n(M, R)$ and then orthogonalizing the columns of the matrix (49) is computationally inefficient. First note that the resulting real projection matrix is a basis matrix for the n -th paired complex conjugate Krylov subspace

$$\mathcal{K}_n^{(p)}(M, R) := \text{span} \{ v, \bar{v} \mid v \in \mathcal{K}_n(M, R) \}. \quad (51)$$

In [24], we study the problem of constructing real basis matrices for paired complex conjugate Krylov subspaces $\mathcal{K}_n^{(p)}(M, R)$ and propose an algorithm that is computationally cheaper than the obvious approach outlined above. In particular, employing the algorithm from [24] allows the efficient construction of passive reduced-order models via projection onto Krylov subspaces with non-real expansion points.

Finally, we present a numerical example that illustrates the benefits of using even a single complex expansion point. With a single real expansion point, a reduced-order model of state-space dimension $n = 138$ is needed to achieve satisfactory convergence throughout the frequency range of interest; see Figure 6 (a). With a single complex expansion point, a reduced-order model of state-space dimension $n = 69$ is sufficient to obtain comparable accuracy; see Figure 6 (b). Thus the dimension of the reduced-order model has been halved by employing a complex expansion point.

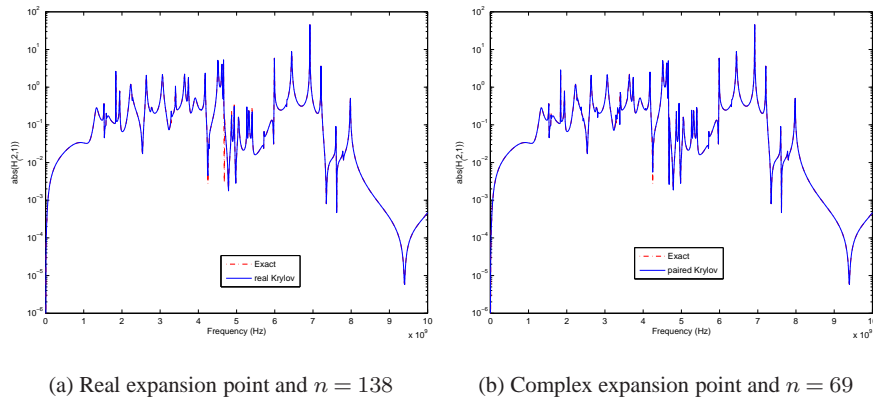


Fig. 6 Single real vs. single complex expansion point

8 Concluding Remarks

Model order reduction is a classical and well-studied subject of control theory. However, in control theory, the systems to be reduced are of relatively small or at most moderate size, and most of the methods developed in this area are not suitable or efficient enough to be applied to large-scale systems. In the early 1990s, the need to efficiently reduce the ever-increasing sizes of the RCL networks used to model the interconnect of VLSI circuits led to a renewed interest in model order reduction especially in the development of methods that are applicable to large-scale systems. Model order reduction techniques based on Krylov subspaces have emerged

as the methods of choice for generating macromodels of large-scale multi-port RCL networks that arise in VLSI circuit simulation. Furthermore, these algorithms have also found applications in other areas, such as structural analysis and computational acoustics.

Despite all the progress in Krylov subspace-based model order reduction of large-scale RCL networks in recent years, there are still many open problems. State-of-the-art structure-preserving methods, such as SPRIM, first generate a basis matrix of the underlying Krylov subspace and then employ explicit projection using some suitable partitioning of the basis matrix to obtain a structure-preserving reduced-order model. In particular, there are two major problems with the use of such explicit projections. First, it requires the storage of the basis matrix, which becomes prohibitive in the case of ‘truly’ large-scale RCL networks. Second, the approximation properties of the resulting structure-preserving reduced-order models are not optimal, and they show that the available degrees of freedom are not fully used in general. It would be highly desirable to have structure-preserving model order reduction methods that do not involve explicit projection and would thus be applicable in the truly large-scale case. Other unresolved issues include the automatic and adaptive choice of the expansion points and robust and reliable stopping criteria and error bounds.

Acknowledgements This work was supported in part by the National Science Foundation through Grant DMS-0613032.

References

1. Anderson, B.D.O., Vongpanitlerd, S.: *Network Analysis and Synthesis*. Prentice-Hall, Englewood Cliffs, New Jersey (1973)
2. Antoulas, A.C.: *Approximation of Large-Scale Dynamical Systems*. SIAM Publications, Philadelphia, Pennsylvania (2005)
3. Antoulas, A.C., Sorensen, D.C.: Approximation of large-scale dynamical systems: An overview. *Int. J. Appl. Math. Comput. Sci.* **11**, 1093–1121 (2001)
4. Bai, Z., Feldmann, P., Freund, R.W.: How to make theoretically passive reduced-order models passive in practice. In: *Proc. IEEE 1998 Custom Integrated Circuits Conference*, pp. 207–210. IEEE, Piscataway, New Jersey (1998)
5. Baker, Jr., G.A., Graves-Morris, P.: *Padé Approximants*, second edn. Cambridge University Press, New York, New York (1996)
6. Celik, M., Pileggi, L., Odabasioglu, A.: *IC Interconnect Analysis*. Kluwer Academic Publishers, Boston/Dordrecht/London (2002)
7. Chiprout, E., Nakhla, M.S.: Analysis of interconnect networks using complex frequency hopping (CFH). *IEEE Trans. Computer-Aided Design* **14**(2), 186–200 (1995)
8. de Villemagne, C., Skelton, R.E.: Model reductions using a projection formulation. *Internat. J. Control* **46**(6), 2141–2169 (1987)
9. Deo, N.: *Graph Theory with Applications to Engineering and Computer Science*. Prentice-Hall, Englewood Cliffs, New Jersey (1974)
10. Eiermann, M., Ernst, O.G.: A restarted Krylov subspace method for the evaluation of matrix functions. *SIAM J. Numer. Anal.* **44**, 2481–2504 (2006)

11. Elmore, W.C.: The transient response of damped linear networks with particular regard to wideband amplifiers. *J. Appl. Phys.* **19**(1), 55–63 (1948)
12. Feldmann, P., Freund, R.W.: Efficient linear circuit analysis by Padé approximation via the Lanczos process. In: Proceedings of EURO-DAC '94 with EURO-VHDL '94, pp. 170–175. IEEE Computer Society Press, Los Alamitos, California (1994)
13. Feldmann, P., Freund, R.W.: Efficient linear circuit analysis by Padé approximation via the Lanczos process. *IEEE Trans. Computer-Aided Design* **14**, 639–649 (1995)
14. Feldmann, P., Freund, R.W.: Reduced-order modeling of large linear subcircuits via a block Lanczos algorithm. In: Proc. 32nd ACM/IEEE Design Automation Conference, pp. 474–479. ACM, New York, New York (1995)
15. Freund, R.W.: Computation of matrix Padé approximations of transfer functions via a Lanczos-type process. In: C. Chui, L. Schumaker (eds.) *Approximation Theory VIII, Vol. 1: Approximation and Interpolation*, pp. 215–222. World Scientific Publishing Co., Inc., Singapore (1995)
16. Freund, R.W.: Krylov-subspace methods for reduced-order modeling in circuit simulation. *J. Comput. Appl. Math.* **123**(1–2), 395–421 (2000)
17. Freund, R.W.: Model reduction methods based on Krylov subspaces. *Acta Numerica* **12**, 267–319 (2003)
18. Freund, R.W.: SPRIM: structure-preserving reduced-order interconnect macromodeling. In: Tech. Dig. 2004 IEEE/ACM International Conference on Computer-Aided Design, pp. 80–87. IEEE Computer Society Press, Los Alamitos, California (2004)
19. Freund, R.W.: On Padé-type model order reduction of J -Hermitian linear dynamical systems. *Linear Algebra Appl.* **429**, 2451–2464 (2008)
20. Freund, R.W.: Structure-preserving model order reduction of RCL circuit equations. In: W. Schilders, H. van der Vorst, J. Rommes (eds.) *Model Order Reduction: Theory, Research Aspects and Applications, Mathematics in Industry*, vol. 13, pp. 49–73. Springer-Verlag, Berlin/Heidelberg, Germany (2008)
21. Freund, R.W.: The SPRIM algorithm for structure-preserving order reduction of general RCL circuits. In: P. Benner, M. Hinze, J. ter Maten (eds.) *Model Reduction for Circuit Simulation, Lecture Notes in Electrical Engineering*, vol. 74, chap. 2. Springer-Verlag, Berlin/Heidelberg, Germany (2011). To appear
22. Freund, R.W., Feldmann, P.: The SyMPVL algorithm and its applications to interconnect simulation. In: Proc. 1997 International Conference on Simulation of Semiconductor Processes and Devices, pp. 113–116. IEEE, Piscataway, New Jersey (1997)
23. Freund, R.W., Feldmann, P.: Reduced-order modeling of large linear passive multi-terminal circuits using matrix-Padé approximation. In: Proc. Design, Automation and Test in Europe Conference 1998, pp. 530–537. IEEE Computer Society Press, Los Alamitos, California (1998)
24. Freund, R.W., Rensi, E.B.: Computing orthonormal bases of paired complex conjugate Krylov subspaces. Technical report, Department of Mathematics, University of California, Davis, California (2011). In preparation
25. Grimme, E.J.: Krylov projection methods for model reduction. Ph.D. thesis, Department of Electrical Engineering, University of Illinois at Urbana-Champaign, Urbana-Champaign, Illinois (1997)
26. Kamon, M., Marques, N.A., Silveira, L.M., White, J.: Automatic generation of accurate circuit models of 3-D interconnect. *IEEE Trans. Compon. Packag. Manuf. Technol—Part B* **21**(3), 225–240 (1998)
27. Lassaux, G., Willcox, K.E.: Model reduction for active control design using multiple-point Arnoldi methods. In: Proceedings of the 41st Aerospace Sciences Meeting & Exhibit. Reno, Nevada (2003). Paper AIAA-2003-0616
28. Odabasioglu, A.: Provably passive RLC circuit reduction. M.S. thesis, Department of Electrical and Computer Engineering, Carnegie Mellon University (1996)
29. Odabasioglu, A., Celik, M., Pileggi, L.T.: PRIMA: passive reduced-order interconnect macromodeling algorithm. In: Tech. Dig. 1997 IEEE/ACM International Conference on Computer-Aided Design, pp. 58–65. IEEE Computer Society Press, Los Alamitos, California (1997)

30. Odabasioglu, A., Celik, M., Pileggi, L.T.: PRIMA: passive reduced-order interconnect macro-modeling algorithm. *IEEE Trans. Computer-Aided Design* **17**(8), 645–654 (1998)
31. Pillage, L.T., Rohrer, R.A.: Asymptotic waveform evaluation for timing analysis. *IEEE Trans. Computer-Aided Design* **9**, 352–366 (1990)
32. Pillage, L.T., Rohrer, R.A., Visweswariah, C.: *Electronic circuit and system simulation methods*. McGraw-Hill, Inc., New York, New York (1995)
33. Ruehli, A.E. (ed.): *Circuit Analysis, Simulation, and Design Part 1: General Aspects of Circuit Analysis and Design*. North-Holland, Amsterdam, The Netherlands (1986)
34. Sheehan, B.N.: ENOR: model order reduction of RLC circuits using nodal equations for efficient factorization. In: *Proc. 36th ACM/IEEE Design Automation Conference*, pp. 17–21. ACM, New York, New York (1999)
35. Vlach, J., Singhal, K.: *Computer Methods for Circuit Analysis and Design*, second edn. Van Nostrand Reinhold, New York, New York (1994)
36. Willems, J.C.: Dissipative dynamical systems, part ii: Linear systems with quadratic supply rates. *Arch. Rational Mech. Anal.* **45**, 352–393 (1972)