Chapter 2
The SPRIM Algorithm for Structure-Preserving Order Reduction of General RCL Circuits

Roland W. Freund

Abstract In recent years, order-reduction techniques based on Krylov subspaces have become the methods of choice for generating macromodels of large-scale multi-port RCL networks that arise in VLSI circuit simulation. A popular method of this type is PRIMA. Its main features are provably passive reduced-order models and a Padé-type approximation property. On the other hand, PRIMA does not preserve other structures inherent to RCL circuits, which makes it harder to synthesize the PRIMA models as actual circuits. For the special case of RCL circuits without voltage sources, SPRIM 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 the formulation of the equations characterizing general RCL circuits as descriptor systems. Second, we describe an extension of SPRIM to the case of general RCL circuits with voltage and current sources. We present some properties of the general SPRIM algorithm and report results of numerical experiments.

Keywords Krylov subspace · Structure preservation · Reduced-order model · Passivity · Electronic circuit · Interconnect

2.1 Introduction

The problem of order reduction of linear dynamical systems has a long history and many powerful methods for generating provably good reduced-order models have been developed. Most of this work was motivated by problems in control theory,

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

P. Benner et al. (eds.), *Model Reduction for Circuit Simulation*,
Lecture Notes in Electrical Engineering, 74, DOI: 10.1007/978-94-007-0089-5_2,

where, typically, the sizes of the dynamical systems to be reduced are relatively small to begin with. In the early 1990s, the continued miniaturization ('Moore's Law') of VLSI circuits lead to the need for efficient order reduction of large-scale linear dynamical systems with ever-increasing state-space dimension. The linear dynamical systems to be reduced in VLSI circuit simulation are RCL models of the VLSI circuit's interconnect system. These RCL models are relatively 'simple', but large, electronic networks with resistors, capacitors, inductors, voltage sources, and current sources as their only elements. Unlike the typical order-reduction problems in control theory, the large state-space dimension of these RCL circuits made the use of most of the more powerful reduction methods prohibitive until recently. Furthermore, RCL circuits are described by systems of differential-algebraic equations, resulting in so-called descriptor systems. Many of the existing order-reduction methods could not be used directly for descriptor systems at that time. Motivated by this 'new' application in VLSI circuit simulation, starting in the early 1990s, there has been tremendous interest in developing techniques for order reduction of large-scale descriptor systems.[1]

Reduced-order modeling techniques based on Padé or Padé-type approximation were quickly recognized to be powerful tools for the problems arising in VLSI circuit simulation. The first such technique was asymptotic waveform evaluation (AWE) [27], which uses explicit moment matching. More recently, the attention has moved to reduced-order models generated by means of Krylov-subspace algorithms, which avoid the typical numerical instabilities of explicit moment matching; see, e.g., the survey papers [12–14].

PVL [8, 9] and its multi-port version MPVL [10] use variants of the Lanczos process [23] to stably compute reduced-order models that represent Padé or matrix-Padé approximations [4] of the circuit transfer function. SyPVL [18] and its multi-port version SyMPVL [11, 19, 20] are versions of PVL and MPVL, respectively, that are tailored to RCL circuits. By exploiting the symmetry of RCL transfer functions, the computational costs of SyPVL and SyMPVL are only half of those of general PVL and MPVL.

Reduced-order modeling techniques based on the Arnoldi process [2], which is another popular Krylov-subspace algorithm, were first proposed in [7, 24–26, 31]. Arnoldi-based reduced-order models are defined by a certain Padé-type approximation property, rather than Padé approximation, and as a result, in general, they are not as accurate as a Padé-based model of the same size. In fact, Arnoldi-based models are known to match only half as many moments as Lanczos-based models; see [13, 24, 25, 31].

In many applications, in particular in VLSI interconnect analysis, the reduced-order model is used as a substitute for the full-blown original model in higher-level simulations. In such applications, it is very important for the reduced-order model to maintain the passivity properties of the original circuit. In [3, 19, 20], it is shown

[1] See Chap. 3 for recent progress in the adaptation of control-theoretic methods to large-scale descriptor systems.

that SyMPVL is passive for RC, RL, and LC circuits. However, the Padé-based reduced-order model that characterizes SyMPVL cannot be guaranteed to be passive for general RCL circuits. On the other hand, in [24–26], it was proved that the Arnoldi-based reduction technique PRIMA produces passive reduced-order models for general RCL circuits. PRIMA employs a block version of the Arnoldi process and then obtains reduced-order models by projecting the matrices defining the RCL transfer function onto the Arnoldi basis vectors. While PRIMA generates provably passive reduced-order models, it does not preserve other structures, such as reciprocity or the block structure of the circuit matrices, inherent to RCL circuits. This has motivated the development of the reduction technique SPRIM [15], which overcomes these disadvantages of PRIMA. In particular, SPRIM generates provably passive and reciprocal macromodels of multi-port RCL circuits. Furthermore, SPRIM models match twice as many moments as the corresponding PRIMA models obtained with identical computational work.

SPRIM was originally proposed in [15] for the special case of RCL circuits without voltage sources. The purpose of this paper is twofold. First, we review the formulation of the equations characterizing general RCL circuits as descriptor systems. Second, we describe an extension of SPRIM to the case of general RCL circuits with voltage and current sources. We present some properties of the general SPRIM algorithm and report results of numerical experiments.

The remainder of this paper is organized as follows. In Sect. 2.2, we review the equations describing general RCL circuits and the formulation of these equations as descriptor systems. In Sect. 2.3, we present some basic facts about generating reduced-order models of descriptor systems by means of Krylov subspace-based projection. In Sect. 2.4, we describe the SPRIM algorithm for the case of general RCL circuits. In Sect. 2.5, we make some comments about how to reduce the number of voltage sources before applying SPRIM to the remaining RCL network. In Sect. 2.6, we report the results of some numerical experiments. Finally, in Sect. 2.7, we mention some open problems and make some concluding remarks.

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 (real or complex) matrices $M = [m_{jk}]$, we denote by $M^T = [m_{kj}]$ the *transpose* of M, and by $M^H := [\overline{m_{kj}}]$ the *Hermitian* (or *complex conjugate*) of M. The identity matrix is denoted by I and the zero matrix by 0; the actual dimensions of I and 0 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*).

2.2 RCL Circuit Equations

We consider general RCL circuits driven by voltage and current sources. In this section, we review the formulation of such RCL circuits as linear time-invariant differential-algebraic equations (DAEs).

2.2.1 The Lumped-Element Approach

The lumped-element approach uses directed graphs to model electronic circuits; see, e.g., [6, 28, 30, 32]. The edges $\mathcal{E}$ of the graph correspond to the electronic elements of the circuit and the nodes $\mathcal{N}$ of the graph correspond to the interconnections of the electronic elements. In this subsection, we review this approach for the case of general RCL circuits driven by voltage and current sources.

The elements of such a general RCL circuit are its resistors, capacitors, inductors, voltage sources, and current sources. With each such element we associate an edge $e \in \mathcal{E}$, written as an ordered pair of nodes:

$$e = (n_1, n_2).$$

Here, $n_1, n_2 \in \mathcal{N}$ are the nodes of the graph representing the interconnections of the element to other circuit elements. We call n_1 the *tail node* of e and n_2 the *head node* of e. Note that 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 e is chosen accordingly. For all other elements, an arbitrary direction of e is assigned. If the computed electric current through such an element is nonnegative, then the current flow is in the direction of e; otherwise, the actual current flow is against the direction of e.

The resulting directed graph $\mathcal{G} = (\mathcal{N}, \mathcal{E})$ can be described by its *incidence matrix* the rows and columns of which correspond to the nodes $n_j \in \mathcal{N}$ and edges $e_k \in \mathcal{E}$, respectively. To this end, we set

$$a_{jk} = \begin{cases} 1 & \text{if edge } e_k \text{ leaves node } n_j, \\ -1 & \text{if edge } e_k \text{ enters node } n_j, \\ 0 & \text{otherwise.} \end{cases}$$

The rows of this matrix add up to the zero row, and thus the matrix is rank deficient. In order to avoid this redundancy, we label one of the nodes as the *ground node* of the circuit and delete the corresponding row. We call the resulting matrix $\mathcal{A}$. It has $|\mathcal{N}| - 1$ rows and $|\mathcal{E}|$ columns, where $|\mathcal{N}|$ and $|\mathcal{E}|$ denote the number of nodes and edges of the graph $\mathcal{G}$, respectively. If $\mathcal{G}$ (viewed as an undirected graph) is connected, then the matrix $\mathcal{A}$ has full row rank:

$$\text{rank } \mathcal{A} = |\mathcal{N}| - 1.$$

Note that $\mathcal{G}$ is indeed connected for any real electronic circuit, and thus, this condition of full row rank of $\mathcal{A}$ is always satisfied.

The matrix $\mathcal{A}$ allows an elegant formulation of the Kirchhoff's laws for the given RCL circuit. To this end, we denote by $i_{\mathcal{E}}$ the vector the entries of which are the currents along the edges $\mathcal{E}$, by $v_{\mathcal{E}}$ the vector the entries of which are the voltages across the edges $\mathcal{E}$, and by v the vector the entries of which are the voltages at the

nodes $\mathcal{N}$, except for the ground node at which the voltage is zero. We remark that $i_\mathcal{E}$ and $v_\mathcal{E}$ are vectors of length $|\mathcal{E}|$ and v is a vector of length $|\mathcal{N}| - 1$. *Kirchhoff's current laws* (KCLs) can then be stated compactly as follows:

$$\mathcal{A} i_\mathcal{E} = 0. \tag{2.1}$$

Kirchhoff's voltage laws (KVLs) have the following compact formulation:

$$\mathcal{A}^T v = v_\mathcal{E}. \tag{2.2}$$

To obtain a complete characterization of the given RCL circuit, the so-called *branch constitutive relations* (BCRs) for the actual circuit elements need to be added to the Kirchhoff's laws (2.1) and (2.2). To formulate the BCRs, it is convenient to assume that the edges $e_k \in \mathcal{E}$ are numbered according to element type in the following order: resistors, capacitors, inductors, voltage sources, and current sources. The matrix $\mathcal{A}$ and the vectors $i_\mathcal{E}$ and $v_\mathcal{E}$ can thus be partitioned as follows:

$$\mathcal{A} = [\,\mathcal{A}_r \quad \mathcal{A}_c \quad \mathcal{A}_l \quad \mathcal{A}_v \quad \mathcal{A}_i\,], \quad i_\mathcal{E} = \begin{bmatrix} i_r \\ i_c \\ i_l \\ i_v \\ i_i \end{bmatrix}, \quad v_\mathcal{E} = \begin{bmatrix} v_r \\ v_c \\ v_l \\ v_v \\ v_i \end{bmatrix}. \tag{2.3}$$

Here, the subscripts r, c, l, v, and i refer to resistors, capacitors, inductors, voltage sources, and current sources, respectively. Using the partitions (2.3), the Kirchhoff's laws (2.1) and (2.2) can be written as follows:

$$\begin{aligned} &\mathcal{A}_r\, i_r + \mathcal{A}_c\, i_c + \mathcal{A}_l\, i_l + \mathcal{A}_v\, i_v + \mathcal{A}_i\, i_i = 0, \\ &\mathcal{A}_r^T v = v_r, \quad \mathcal{A}_c^T v = v_c, \quad \mathcal{A}_l^T v = v_l, \quad \mathcal{A}_v^T v = v_v, \quad \mathcal{A}_i^T v = v_i. \end{aligned} \tag{2.4}$$

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

$$v_r(t) = R i_r(t), \quad i_c(t) = C \frac{d}{dt} v_c(t), \quad v_l(t) = L \frac{d}{dt} i_l(t). \tag{2.5}$$

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 (2.5) always satisfy

$$R \succ 0, \quad C \succ 0, \quad \text{and} \quad L \succ 0. \tag{2.6}$$

2.2.2 RCL Circuit Equations as Integro-DAEs

Equations 2.4 and 2.5, together with suitable initial conditions for some initial time t_0, completely characterize the time behavior of the RCL circuit. Here,

$$v_v(t) \quad \text{and} \quad i_i(t), \quad t \geq t_0, \tag{2.7}$$

are given functions, which describe the voltages and currents provided by the voltage and current sources, respectively. The functions

$$v(t), \quad v_r(t), \quad v_c(t), \quad v_l(t), \quad v_i(t), \quad i_r(t), \quad i_c(t), \quad i_l(t), \quad i_v(t) \tag{2.8}$$

are the unknown solutions of the Eqs. 2.4 and 2.5. These equations can be greatly simplified by using the BCRs (2.5) and the formulae for v_r, v_c, and v_l in (2.4) to eliminate all but the unknown functions $v(t)$ and $i_v(t)$.

To this end, we now assume that $t_0 = 0$ and that

$$i_l(0) = 0. \tag{2.9}$$

The third relation in (2.5) can then be rewritten in the form

$$i_l(t) = L^{-1} \int_0^t v_l(\tau) d\tau. \tag{2.10}$$

Using (2.10), the first two relations of (2.5), and the formulae for v_r, v_c, and v_l in (2.4), it follows that

$$i_r(t) = R^{-1} \mathcal{A}_r^T v(t), \quad i_c(t) = C \mathcal{A}_c^T \frac{d}{dt} v(t), \quad i_l(t) = L^{-1} \mathcal{A}_l^T \int_0^t v(\tau) d\tau. \tag{2.11}$$

Inserting these relations into (2.4), we obtain the coupled system of equations

$$\begin{aligned} E_{11} \frac{d}{dt} v(t) - A_{11} v(t) + \mathcal{A}_l L^{-1} \mathcal{A}_l^T \int_0^t v(\tau) d\tau + \mathcal{A}_v i_v(t) &= -\mathcal{A}_i i_i(t), \\ -\mathcal{A}_v^T v(t) &= -v_v(t) \end{aligned} \tag{2.12}$$

for the unknown functions $v(t)$ and $i_v(t)$. Here, we have set

$$E_{11} := \mathcal{A}_c C \mathcal{A}_c^T \quad \text{and} \quad A_{11} := -\mathcal{A}_r R^{-1} \mathcal{A}_r^T. \tag{2.13}$$

The system (2.12) is completed by adding initial values

$$v(0) \quad \text{and} \quad i_v(0) \tag{2.14}$$

for the solutions $v(t)$ and $i_v(t)$ at initial time $t_0 = 0$.

We remark that the system (2.12) with initial conditions (2.14) represents an *integro-differential-algebraic equation* (integro-DAE). This is the most compact formulation of the equations describing RCL circuits. By solving the integro-DAE (2.12), we obtain the vector $v(t)$ of nodal voltages and the vector $i_v(t)$ of currents through the voltage sources. The remaining circuit quantities, namely

$$v_r(t), \quad v_c(t), \quad v_l(t), \quad v_i(t), \quad i_r(t), \quad i_c(t), \quad \text{and} \quad i_l(t),$$

are then readily computed using the KVLs in (2.4), the first two BCRs in (2.5), and (2.10).

2.2.3 RCL Circuit Equations as Descriptor Systems

While integro-DAEs of the type (2.12) are the most compact formulation of RCL circuit equations, they are not directly amenable to Krylov subspace-based reduction techniques. Instead, we rewrite (2.13) as a descriptor system. This can be done by treating the vector $i_l(t)$ of inductor currents also as an unknown function, along with the functions $v(t)$ and $i_v(t)$. We denote by

$$x(t) := \begin{bmatrix} v(t) \\ i_l(t) \\ i_v(t) \end{bmatrix} \tag{2.15}$$

the resulting new *state-space vector* of unknowns. Furthermore, we define vectors of input and output functions

$$u(t) := \begin{bmatrix} -i_i(t) \\ v_v(t) \end{bmatrix} \quad \text{and} \quad y(t) := \begin{bmatrix} v_i(t) \\ -i_v(t) \end{bmatrix}. \tag{2.16}$$

Recall from (2.12) that the entries of the *input vector* $u(t)$ are all given functions. The *output vector* $y(t)$ is readily obtained from the state-space vector (2.15). Indeed, using the relation $v_i(t) = \mathcal{A}_i^T v(t)$ from (2.4), it follows that

$$y(t) = B^T x(t), \quad \text{where} \quad B := \begin{bmatrix} \mathcal{A}_i & 0 \\ 0 & 0 \\ 0 & -I \end{bmatrix}. \tag{2.17}$$

We now re-introduce $i_l(t)$ into (2.12). To this end, we employ the formula for $i_l(t)$ stated in (2.11) and the differentiated form thereof:

$$L\frac{d}{dt} i_l(t) = \mathcal{A}_l^T v(t).$$

The resulting equivalent version of the system (2.12) can be stated as follows:

$$\begin{aligned} E_{11}\frac{d}{dt}v(t) - A_{11}v(t) + \mathcal{A}_l i_l(t) + \mathcal{A}_v i_v(t) &= -\mathcal{A}_i i_i(t), \\ L\frac{d}{dt}i_l(t) - \mathcal{A}_l^T v(t) &= 0, \\ -\mathcal{A}_v^T v(t) &= -v_v(t). \end{aligned} \tag{2.18}$$

Finally, setting

$$E := \begin{bmatrix} E_{11} & 0 & 0 \\ 0 & L & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad \text{and} \quad A := \begin{bmatrix} A_{11} & -\mathcal{A}_l & -\mathcal{A}_v \\ \mathcal{A}_l^T & 0 & 0 \\ \mathcal{A}_v^T & 0 & 0 \end{bmatrix}, \tag{2.19}$$

and adding the relation (2.17) and (2.18), we obtain the following formulation of the RCL circuit equations as a *descriptor system*:

$$\begin{aligned} E\frac{d}{dt}x(t) &= Ax(t) + Bu(t), \\ y(t) &= B^T x(t). \end{aligned} \tag{2.20}$$

The system (2.20) is completed by the initial conditions from (2.9) and (2.14):

$$x(0) = x_0 := \begin{bmatrix} v(0) \\ 0 \\ i_v(0) \end{bmatrix}.$$

In the following, we denote by N and m the length of the state-space vector $x(t)$ and the length of the input vector $u(t)$ of (2.20), respectively. The number N is called the *state-space dimension* of the descriptor system (2.20). We remark that m is also the length of the vector-valued output vector $y(t)$ of (2.20). In particular, A and E are $N \times N$ matrices, and B is an $N \times m$ matrix. Note that, in view of (2.16), m is the total number of voltage and current sources in the RCL circuit.

The formulation (2.20) is essential for using Krylov subspace techniques for model order reduction of RCL circuits. Furthermore, the matrices A and E in (2.20) need to be such that the *matrix pencil*

$$sE - A, \quad s \in \mathbb{C}, \tag{2.21}$$

is *regular*, i.e., the matrix $sE - A$ is singular only for finitely many values of $s \in \mathbb{C}$.

Regularity of the matrix pencil (2.21) is equivalent to certain rank conditions involving the subblocks of the matrix $\mathcal{A}$ in (2.3). Indeed, the matrix pencil (2.21) is regular if, and only if, the matrix

$$\mathcal{A}_v \quad \text{has full column rank} \tag{2.22}$$

and the matrix

$$[\,\mathcal{A}_r \quad \mathcal{A}_c \quad \mathcal{A}_l \quad \mathcal{A}_v\,] \quad \text{has full row rank.} \tag{2.23}$$

For an elementary proof of this characterization of regularity, we refer the reader to [17, Theorem 1].

The rank conditions (2.22) and (2.23) have simple interpretations in terms of the RCL circuit described by the descriptor system (2.20). Condition (2.22) means that the subcircuit consisting of only the voltage sources of the given RCL circuit has no closed (undirected) loops. Condition (2.23) means that the (undirected) graph corresponding to the subcircuit obtained from the given RCL circuit by deleting all current sources is still connected. Both these conditions are satisfied for any practically relevant RCL circuit, and thus from now on, we always assume that the matrix pencil $sE - A$ associated with the descriptor system (2.20) is regular. Furthermore, we stress that the occurrence of a singular matrix pencil is a strong indication that some error was made in the design or the modeling of the RCL circuit described by (2.20).

2.2.4 Passivity

Any RCL circuit is *passive*, i.e., it consumes energy (provided by the voltage and current sources), but does not generate energy.

One of the possible mathematical characterizations of passivity uses the concept of the transfer function of a descriptor system. Recall that the matrix pencil (2.21) is assumed to be regular. Therefore, we can define an $m \times m$-matrix-valued function by setting

$$H(s) := B^T(sE - A)^{-1}B, \quad s \in \mathbb{C}. \tag{2.24}$$

Note that H is a rational function, with possible poles at those finitely many values of $s \in \mathbb{C}$ for which the matrix $sE - A$ is singular. The function (2.24) is called the *transfer function* of the descriptor system (2.20).

The function (2.24), H, is said to be *positive real* if it has no poles in the right-half

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

of the complex plane and

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

It is well known that a system described by (2.20) is passive if, and only if, the transfer function of (2.20) is positive real, see, e.g., [1].

Since RCL circuits are passive, it thus follows that the transfer functions of the circuit equations stated as descriptor systems (2.20) are guaranteed to be positive real. An alternative proof of this fact uses the following simple theorem, which can be found as Theorem 13 in [13].

Theorem 1 *Let $A, E \in \mathbb{R}^{N\times m}$ and $B \in \mathbb{R}^{N\times m}$. Assume that*

$$E = E^T \succeq 0, \quad A + A^T \preceq 0, \tag{2.25}$$

and that the matrix pencil $sE - A$ is regular. Then, the function (2.24), H, is positive real.

In view of this theorem, we only need to verify that the matrices A and E defined in (2.19) satisfy the conditions (2.25). Note that by (2.6) and (2.13), we have

$$E_{11} = E_{11}^T \succeq 0, \quad L = L^T \succ 0, \quad \text{and} \quad A_{11} = A_{11}^T \preceq 0.$$

Together with (2.19) it follows that

$$E = E^T \succeq 0 \quad \text{and} \quad A + A^T = \begin{bmatrix} 2A_{11} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \preceq 0.$$

Hence the transfer functions of RCL circuit equations stated as descriptor systems (2.20) are indeed positive real.

2.3 Projection-Based Order Reduction

In this section, we review some basic facts about generating reduced-order models of descriptor systems (2.20) by means of projection. From now on, we always assume that the system (2.20) describes a given RCL circuit. In particular, the matrices A and E are of the form (2.19), and the matrix B is of the form (2.17).

2.3.1 Reduced-Order Models

A general *reduced-order model* of (2.20) is a descriptor system of the same form as (2.20), but with a reduced state-space dimension $n < N$, instead of N. Thus a reduced-order model is of the form

$$\begin{aligned} E_n \frac{d}{dt}\tilde{x}(t) &= A_n\tilde{x}(t) + B_n u(t), \\ \tilde{y}(t) &= B_n^T\tilde{x}(t), \end{aligned} \tag{2.26}$$

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 $u(t)$ is the same as in (2.20). In particular, the number m is unchanged from (2.20). The output vector $\tilde{y}(t)$ of (2.26) is only an approximation of the original output vector $y(t)$ of (2.20). In fact, the problem of 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 (2.26) is a 'sufficiently good' approximation of the output vector of the original system (2.20).

Provided that the matrix pencil

$$sE_n - A_n, \quad s \in \mathbb{C}, \tag{2.27}$$

associated with (2.26) is regular, we can define a transfer function as before:

$$H_n(s) := B_n^T (sE_n - A_n)^{-1} B_n, \quad s \in \mathbb{C}. \tag{2.28}$$

In terms of transfer functions, the problem of 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 (2.28), H_n, of the reduced-order model is a 'sufficiently good' approximation of the transfer function (2.24), H of the original system:

$$H_n(s) \approx H(s) \quad \text{in 'some sense'}. \tag{2.29}$$

2.3.2 Order Reduction Via Projection

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

$$V_n \in \mathbb{R}^{N \times n} \quad \text{with} \quad \operatorname{rank} V_n = n \tag{2.30}$$

be given. Then, by simply setting

$$A_n := V_n^T A V_n, \quad E_n := V_n^T E V_n, \quad \text{and} \quad B_n := V_n^T B, \tag{2.31}$$

one obtains a reduced-order model (2.26) 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

$$\tilde{x}(t) = V_n x(t)$$

for the state-space vector $\tilde{x}(t)$ of the reduced-order model (2.26). Recall that $x(t)$ denotes the state-space vector of the original system (2.20).

There are two appealing aspects of the projection approach:

1. Reduced-order models obtained by means of projection trivially preserve passivity of the original system; see, e.g., [24–26]. Indeed, the only additional condition on the matrix (2.30), V_n, is that the resulting matrix pencil (2.27) is regular. Recall that A and E satisfy the semidefiniteness properties (2.25). The definitions of A_n and E_n in (2.31) readily imply that these matrices also satisfy (2.25). Therefore, by Theorem 1, the transfer function (2.28), H_n is positive real, and thus the corresponding reduced-order model (2.26) is passive.

2. By choosing the matrix V_n such that the subspace spanned by its columns contains a certain Krylov subspace, one obtains reduced-order models for which (2.28), H_n, is a Padé-type approximation of the original transfer function (2.24), H; see Sect. 2.3.4 below.

2.3.3 Block Krylov Subspaces

Suppose we want to evaluate the transfer function (2.24), H, of the descriptor system (2.20) at some point $s_0 \in \mathbb{C}$ at which the matrix $s_0E - A$ is nonsingular. The most efficient way of obtaining $H(s_0)$ is to first solve the system of linear equations

$$(s_0E - A)X(s_0) = B \tag{2.32}$$

for $X(s_0)$ and then compute

$$H(s_0) = B^T X(s_0).$$

The coefficient matrix $s_0E - A$ of (2.32) is large, but sparse, and as it is the case for all practical circuit matrices, a sparse LU factorization of this matrix can be computed with typically little fill-in; see, e.g., [30, 32].

The basic idea behind Krylov subspace-based order reduction for electronic circuits is to reuse the LU factorization, which was computed to obtain $H(s_0)$, to generate the information contained in the leading Taylor coefficients of H expanded about s_0. To this end, we rewrite the transfer function (2.24) as follows:

$$H(s) = B^T(s_0E - A + (s - s_0)E)^{-1}B = B^T(I + (s - s_0)M)^{-1}R, \tag{2.33}$$

where

$$M := (s_0E - A)^{-1}E \quad \text{and} \quad R := (s_0E - A)^{-1}B. \tag{2.34}$$

In view of (2.33), the Taylor expansion of H about s_0 is given by

$$H(s) = \sum_{j=0}^{\infty}(-1)^j B^T M^j R \quad (s - s_0)^j. \tag{2.35}$$

The leading Taylor coefficients of H can thus be obtained by computing inner products of the columns of the matrix B and the leading columns of the *block Krylov matrix*

$$\begin{bmatrix} R & MR & M^2R & \cdots & M^{j-1}R & \cdots \end{bmatrix}. \tag{2.36}$$

Let $N_{\max}(\leq N)$ denote the rank of this matrix. Then for $\hat{n} = 1, 2, \ldots, N_{\max}$, the $\hat{n}$th *block Krylov subspace* (induced by M and R) is defined as the $\hat{n}$-dimensional

subspace of $\mathbb{C}^N$ spanned by the first $\hat{n}$ linearly independent columns of the block Krylov matrix (2.36). In the following, $\mathcal{K}_{\hat{n}}(M,R)$ denotes this $\hat{n}$th block Krylov subspace. Note that, by construction, $\mathcal{K}_{\hat{n}}(M,R)$ contains the necessary information to generate at least the first

$$\left\lfloor \frac{\hat{n}}{m} \right\rfloor \tag{2.37}$$

Taylor coefficients of the expansion (2.35) of H about s_0. In the generic case, the integer (2.37) is the exact number of Taylor coefficients that can be obtained from $\mathcal{K}_{\hat{n}}(M,R)$. However, in certain degenerate cases, one can obtain even more coefficients; we refer the reader to [16, 17] for a complete characterization of the exact number of coefficients.

We remark that for actual numerical computations, this definition of $\mathcal{K}_{\hat{n}}(M,R)$ is useless. Instead, one employs a suitable Krylov subspace algorithm to generate a numerically well-behaved basis of $\mathcal{K}_{\hat{n}}(M,R)$. One such algorithm is the band Arnoldi process described in [14]. It produces orthonormal basis vectors for the subspaces $\mathcal{K}_{\hat{n}}(M,R)$. We remark that Krylov subspace algorithms involve the matrix M only in the form of matrix-vector products Mv. For the computation of these, the matrix M never needs to be formed explicitly. Instead, in view of the definition of M in (2.34), each matrix-vector product Mv can be obtained via one sparse multiplication with E and two sparse triangular solves with the LU factors of the matrix $s_0E - A$.

Finally, we stress that for general $s_0 \in \mathbb{C}$, $\mathcal{K}_{\hat{n}}(M,R)$ is a subspace of $\mathbb{C}^N$. If we restrict s_0 to be real, then the matrices M and R are real and $\mathcal{K}_{\hat{n}}(M,R)$ is a subspace of $\mathbb{R}^N$.

2.3.4 Krylov Subspace-Based Projection

We now employ the block Krylov subspaces defined in Sect. 2.3.3 to choose suitable projection matrices V_n.

Recall from (2.30) that the matrices V_n are assumed to be real. For the sake of generating reduced-order models, this assumption is not essential, and in fact, complex $N \times n$ matrices can be used as well. However, in order to obtain passive models, the matrices V_n need to be real. To guarantee this condition, from now on we assume that

$$s_0 \in \mathbb{R}, \tag{2.38}$$

so that $\mathcal{K}_{\hat{n}}(M,R)$ is a subspace of $\mathbb{R}^N$. At the end of this subsection, we include some remarks about how to proceed in the case of truly complex $s_0 \in \mathbb{C} \setminus \mathbb{R}$.

For Krylov subspace-based projection, we choose the matrix (2.30), V_n, such that

$$\mathcal{K}_{\hat{n}}(M,R) \subseteq \text{colspan } V_n. \tag{2.39}$$

Since, by construction, $\mathcal{K}_{\hat{n}}(M,R)$ has dimension $\hat{n}$ and, by (2.30), V_n is assumed to have rank n, the condition (2.39) implies that

$$\hat{n} \leq n. \tag{2.40}$$

An obvious choice for V_n is a matrix whose columns form a basis of $\mathcal{K}_{\hat{n}}(M,R)$, such as the vectors generated by the band Arnoldi process. In this case, we have $\hat{n} = n$ in (2.40).

Let A_n, E_n, and B_n be the matrices (2.31), and let H_n be the transfer function (2.28) of the corresponding reduced-order model (2.26). The main result of Krylov subspace-based projection then states that H_n is a *Padé-type approximation* of the original transfer function (2.24), H, in the following sense:

$$H_n(s) = H(s) + \mathcal{O}\Big((s-s_0)^{j(\hat{n})}\Big), \quad \text{where} \quad j(\hat{n}) \geq \left\lfloor \frac{\hat{n}}{m} \right\rfloor. \tag{2.41}$$

Moreover, in the generic case, $j(\hat{n}) = \lfloor \hat{n}/m \rfloor$ in (2.41). The property (2.41) is well known. For example, it was established for various special cases in [5, 21, 24]. Proofs for the general case be found in [13, 16]. We remark that the matrix-valued coefficients of the Taylor expansion of the transfer function (2.24), H, about the expansion point s_0 are often called *moments* and that the Padé-type approximation (2.41) is also referred to as *moment matching*.

If instead of (2.38), the expansion point s_0 is chosen to be truly complex, then $\mathcal{K}_{\hat{n}}(M,R)$ is a complex subspace and the projection matrix V_n in (2.39) will be complex as well in general. One possibility of keeping the projection matrix real is to replace the complex matrix V_n satisfying (2.39) by the real matrix

$$[\,\text{Re } V_n \quad \text{Im } V_n\,]. \tag{2.42}$$

The obvious disadvantage of this approach is that the dimension of the reduced-order model is doubled. Furthermore, in general, the matrix (2.42) is not guaranteed to have full column rank, and so before using (2.42) as a projection matrix, one would need to check for and possibly delete any linearly dependent columns of (2.42). On the other hand, the transfer function of the resulting reduced-order model will satisfy a Padé-type property of the form (2.41) for both s_0 and the complex conjugate expansion point $\overline{s_0}$; see, e.g., [22].

2.3.5 Structure Preservation

Recall from (2.19) and (2.17) that the matrices A, E, and B exhibit certain block structures reflecting the fact that the descriptor system (2.20) describes an RCL

circuit. As long as the expansion point s_0 is chosen to be real, the reduced-order model generated via projection preserves the passivity of the original RCL circuit, but not these block structures of the data matrices. In fact, in general V_n will be a dense matrix, and then the data matrices (2.31) of the reduced-order model will be dense matrices as well.

For example, consider the 'minimal' choice of the matrix V_n in (2.39), i.e., $\hat{n} = n$ and V_n is chosen as any matrix whose columns span $\mathcal{K}_{\hat{n}}(M, R)$. The resulting order reduction method is mathematically equivalent to PRIMA [24–26]. However, in general the data matrices of the PRIMA reduced-order model are dense and do not preserve the block structures of the original matrices A, E, and B. In the next section, we describe how these structures can indeed be preserved by taking advantage of the fact that in (2.39) we can choose a matrix V_n with $n > \hat{n}$.

The reader may ask why preservation of the block structures of the matrices A, E, and B is important. There are two reasons:

1. In practice, one would like to synthesize the system described by the reduced-order model (2.26) as an actual physical electronic circuit. It is well known that passivity of the reduced-order model is sufficient to guarantee the existence of such a synthesized circuit; see, e.g., [1]. However, passivity alone is not enough to guarantee synthesis as an RCL circuit, and in general, some other circuit elements are needed. Additional properties, such as *reciprocity*, of the reduced-order model are necessary to ensure synthesis as an RCL circuit. While preserving the block structures of the data matrices of the original descriptor system alone is not enough to guarantee synthesis as an RCL circuit, it ensures reciprocity and significantly increases the probability that the reduced-order model can be synthesized as an actual RCL circuit.
2. Preserving the block structure also doubles the number of Taylor coefficients that are matched in the Padé-type approximation property of the reduced-order model. More precisely, the structure-preserving SPRIM algorithm matches twice as many coefficients as the non-structure-preserving PRIMA algorithm; see Eq. 2.47 in Sect. 2.4.3 below.

2.4 The SPRIM Algorithm

The SPRIM (*Structure-Preserving Reduced-order Interconnect Macromodeling*) algorithm was originally introduced in [15] for the special case of RCL circuits with only current sources, but no voltage sources. In this case, the third block row and column of A and E in (2.19) and the third block row of E in (2.17) are non-existent. This significantly simplifies both the implementation of SPRIM and the derivation of certain theoretical properties of SPRIM. In this section, we describe the SPRIM algorithm for the case of general RCL circuits with both voltage and current sources.

2.4.1 Preserving the Block Structures

The main computational step of SPRIM is the generation of a suitable basis for the $\hat{n}$th block Krylov subspace $\mathcal{K}_{\hat{n}}(M, R)$. This step is identical to what is done in PRIMA. Let $\hat{V}_{\hat{n}}$ be the resulting matrix, the columns of which form a basis of $\mathcal{K}_{\hat{n}}(M, R)$. PRIMA employs this matrix as the projection matrix to obtain the reduced-order data matrices (2.31). As pointed out before, in general, these PRIMA data matrices are dense and thus do not preserve the block structures of the original matrices A, E, and B.

Instead of using the matrix $\hat{V}_{\hat{n}}$ directly for the projection, 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:

$$\hat{V}_{\hat{n}} = \begin{bmatrix} V^{(1)} \\ V^{(2)} \\ V^{(3)} \end{bmatrix}. \tag{2.43}$$

Here, the block sizes correspond to the block sizes of A and E in (2.19). 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 (2.43). In particular, the third block, $V^{(3)}$ is of size $n_v \times \hat{n}$, where n_v denotes the number of voltage sources of the given RCL circuit. The number n_v is very small and usually $n_v < \hat{n}$. 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 (instead of spans) the same space as the columns of $V^{(3)}$, but 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 actual implementation of SPRIM, there is 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 columns rank. Next, we set up the actual projection matrix V_n as follows:

$$V_n := \begin{bmatrix} \tilde{V}^{(1)} & 0 & 0 \\ 0 & \tilde{V}^{(2)} & 0 \\ 0 & 0 & \tilde{V}^{(3)} \end{bmatrix}. \tag{2.44}$$

By construction, we have

$$\mathcal{K}_{\hat{n}}(M, R) = \text{colspan}\hat{V}_{\hat{n}} \subseteq \text{colspan } V_n. \tag{2.45}$$

Thus the matrix (2.44), V_n, satisfies the condition (2.39), which in turn guarantees the Padé-type approximation property (2.41). Furthermore, in view of the block structure of V_n, the data matrices (2.31) 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.

2.4.2 The Algorithm

The order reduction procedure outlined in the previous subsection is the SPRIM algorithm for general RCL circuits. SPRIM can be formulated as an actual algorithm in the following form.

Algorithm 2 (SPRIM for general RCL circuits)

- 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 $s_0 \in \mathbb{R}$.

- 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 block Krylov subspace method (applied to M and R) to construct the columns of the basis matrix

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

  of the $\hat{n}$th block 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 $V_{\hat{n}}$ 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

$$\tilde{A}_{11} = \left(\tilde{V}^{(1)}\right)^T A_{11} \tilde{V}^{(1)}, \quad \tilde{\mathcal{A}}_l = \left(\tilde{V}^{(1)}\right)^T \mathcal{A}_l \tilde{V}^{(2)}, \quad \tilde{\mathcal{A}}_v = \left(\tilde{V}^{(1)}\right)^T \mathcal{A}_v \tilde{V}^{(3)},$$
$$\tilde{E}_{11} = \left(\tilde{V}^{(1)}\right)^T E_{11} \tilde{V}^{(1)}, \quad \tilde{L} = \left(\tilde{V}^{(2)}\right)^T L \tilde{V}^{(2)}, \quad \tilde{\mathcal{A}}_i = \left(\tilde{V}^{(1)}\right)^T \mathcal{A}_i.$$

- Output: the data matrices

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

$$B_n = \begin{bmatrix} \tilde{\mathcal{A}}_i & 0 \\ 0 & 0 \\ 0 & -\left(\tilde{V}^{(3)}\right)^T \end{bmatrix},$$

of the SPRIM reduced-order model

$$\begin{aligned} E_n \frac{d}{dt}\tilde{x}(t) &= A_n \tilde{x}(t) + B_n u(t), \\ \tilde{y}(t) &= B_n^T \tilde{x}(t). \end{aligned} \tag{2.46}$$

2.4.3 The Padé-Type Approximation Property of SPRIM

Recall that SPRIM is a Krylov subspace-based projection method. In particular, the matrix V_n used in the projection satisfies the condition (2.39), which implies the Padé-type approximation property (2.41). It turns out that SPRIM actually has a stronger Padé-type approximation property. More precisely, the transfer function H_n of the SPRIM reduced-order model (2.46) satisfies

$$H_n(s) = H(s) + \mathcal{O}\Big((s - s_0)^{2j(\hat{n})}\Big), \quad \text{where} \quad j(\hat{n}) \geq \left\lfloor \frac{\hat{n}}{m} \right\rfloor, \tag{2.47}$$

instead of (2.41). The integer $j(\hat{n})$ is the same as in (2.41), and in the generic case, $j(\hat{n}) = \lfloor \hat{n}/m \rfloor$ in (2.47). The property means that at the expansion point s_0, the transfer function of the SPRIM reduced-order model matches twice as many leading Taylor coefficients as the theory of general Krylov subspace-based projection methods predicts.

This higher accuracy of SPRIM is a consequence of structure preservation. Projection methods that do not preserve the structures of the original data matrices, such as PRIMA, do not satisfy the more accurate Padé-type property (2.47).

The proof of (2.47) is relatively straightforward for the special case of RCL circuits without voltage sources; see [15]. In this case, the third block row and column of the matrices A and E and the third block row of the matrix B are empty, and one can easily make both matrices A and E symmetric by changing the sign of the second block row, without changing B. This symmetry, along with the corresponding symmetry of the data matrices of the SPRIM reduced-order models, can be used to establish (2.47). In the case of general RCL circuits, it is no longer possible to make A and E symmetric without changing B, and a different approach

to proving (2.47) is required. The key observation here is that the matrices A and E are *J-symmetric*, i.e., they satisfy

$$JA = A^T J \quad \text{and} \quad JE = E^T J,$$

with respect to the indefinite matrix

$$J = \begin{bmatrix} I & 0 & 0 \\ 0 & -I & 0 \\ 0 & 0 & -I \end{bmatrix},$$

where the I's denote identity matrices of the same size as the diagonal blocks of A and E. Since the matrices A_n and E_n of the SPRIM reduced-order model have the same block structure as A and E, respectively, the matrices A_n and E_n are J_n-symmetric with respect to a 'reduced' version J_n of J. Finally, the projection matrix (2.44), V_n, employed in SPRIM is *compatible* with J and J_n in the sense that

$$JV_n = V_n J_n. \tag{2.48}$$

In [16], we developed a general theory of Krylov subspace-based projection of J-symmetric descriptor systems. More precisely, we showed that the stronger Padé-type approximation property (2.47) holds true, provided that the data matrices of the reduced-order models are J_n-symmetric and the compatibility condition (2.48) is satisfied. By applying this more general result [16, Theorem 9] to SPRIM, we obtain its stronger Padé-type approximation property (2.47).

2.5 Treatment of Voltage Sources

Recall from (2.19) that the third block row and column of the matrices A and E arise due to the presence of voltage sources in the given RCL circuit. Note that the size of the third block rows is $n_v \times N$ and the size of the third block columns is $N \times n_v$. Here, n_v denotes the number of voltage sources, which is usually very small. In SPRIM, the corresponding third block $V^{(3)}$ of the matrix (2.43), $\hat{V}_{\hat{n}}$ is usually rank deficient and thus needs to be replaced by a block $\tilde{V}^{(3)}$ of full column rank, see Algorithm 2.

In many cases, it is actually possible to treat all or at least some of the voltage sources separately and to apply the model reduction algorithm itself to a slightly smaller RCL circuit. The reason is that voltage sources are usually connected to the ground node of the circuit and they are connected to the remaining RCL circuit by a resistor. Such a case is illustrated in Fig. 2.1, which shows an RCL circuit with 4 such voltage sources connected to the ground node and by a resistor to the remaining RCL network. Recall from (2.16) that the given voltages of the voltage sources are part of the input vector $u(t)$ of the descriptor system (2.20) and from (2.16) that the unknown currents through the voltage sources and the voltages at

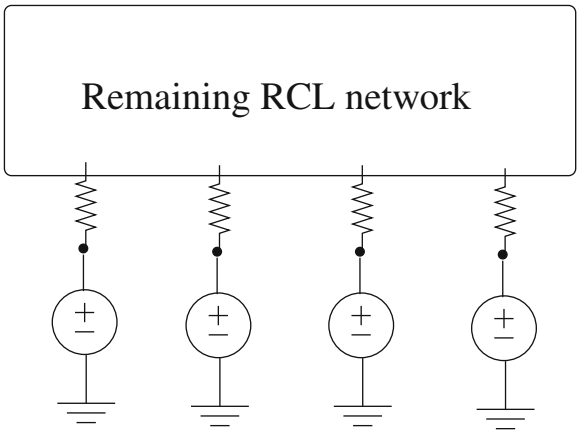

Fig. 2.1 An RCL circuit for which 4 voltage sources can be eliminated before model reduction is applied

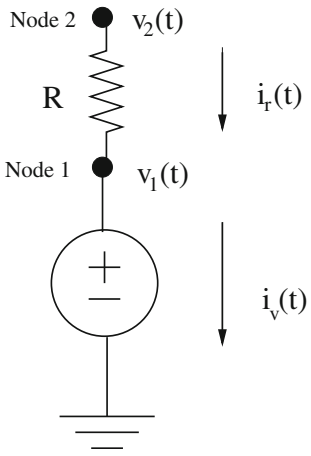

Fig. 2.2 A typical voltage source connected to the ground node and by a single resistor to the remaining RCL network

the circuit nodes are part of the unknown state-vector $x(t)$ of (2.20). For voltage sources connected to the remaining RCL network as in Fig. 2.1, there are some trivial relations between some of these circuit quantities, which can be used to reduce the state-space dimension of (2.20). More precisely, consider a typical voltage source connected to ground and to a resistor, as shown in Fig. 2.2. In this case, the voltage $v_1(t)$ at node 1 is equal to the voltage $v_v(t)$ provided by the voltage source:

$$v_1(t) = v_v(t). \tag{2.49}$$

By Kirchhoff's current law, the unknown current $i_v(t)$ through the voltage source is equal to the unknown current $i_r(t)$ through the resistor. Together with Ohm's law for resistors, it follows that

$$i_v(t) = i_r(t) = \frac{1}{R}(v_2(t) - v_1(t)), \tag{2.50}$$

where $v_2(t)$ denotes the unknown voltage at node 2 and R is the given resistance of the resistor. Using the two relations (2.49) and (2.50), we can eliminate the unknowns $v_1(t)$ and $i_v(t)$ from the circuit equation, thus reducing the state-space dimension by 2.

The above procedure can be carried out for all voltage sources that are connected to the ground node and by a single resistor to the remaining RCL network. If $n_v^{(e)}$ denotes the number of such voltage sources in the given RCL circuit, then the state-space dimension of the resulting descriptor system is $N^{(r)} := N - 2n_v^{(e)}$. Here, N is the state-space dimension of the original descriptor system (2.20).

It is relatively straightforward to carry out this elimination on the matrices of the original system (2.20). Here, we omit the full details and just state the final result. The given RCL circuit is again described by a descriptor system that now has the following form:

$$\begin{aligned} E^{(r)}\frac{d}{dt}x^{(r)}(t) &= A^{(r)}x^{(r)}(t) + B^{(r)}u(t), \\ y(t) &= Du(t) + \left(C^{(r)}\right)^T x^{(r)}(t). \end{aligned} \tag{2.51}$$

Here, $A^{(r)}$ and $E^{(r)}$ are $N^{(r)} \times N^{(r)}$ matrices corresponding to the remaining RCL network. These matrices have the same block structure as the matrices in (2.20), but with the sizes of each first and third block row and column reduced by $n_v^{(e)}$. In particular, in the case that all voltage sources have been eliminated, then $A^{(r)}$ and $E^{(r)}$ have no third block rows and columns at all. The state-vector $x^{(r)}(t)$ of (2.51) is obtained from the state-vector $x(t)$ of (2.20) by deleting the voltages at the nodes between the eliminated voltage sources and the directly connected resistors and the currents through the eliminated voltage sources. The input and output vectors of (2.51) are the same as in (2.16). We now partition these vectors as follows:

$$u(t) = \begin{bmatrix} -i_i(t) \\ v_v^{(e)}(t) \\ v_v^{(r)}(t) \end{bmatrix} \quad \text{and} \quad y(t) = \begin{bmatrix} v_i(t) \\ -i_v^{(e)}(t) \\ -i_v^{(r)}(t) \end{bmatrix}. \tag{2.52}$$

Here, the superscripts "(e)" and "(r)" refer to eliminated and remaining voltages sources, respectively. Finally, in (2.51), $D \in \mathbb{R}^{m\times m}$ and $B^{(r)}, C^{(r)} \in \mathbb{R}^{N^{(r)}\times m}$ are matrices of the following form:

$$B^{(r)} = \begin{bmatrix} \mathcal{A}_i & B_{12} & 0 \\ 0 & 0 & 0 \\ 0 & 0 & -I \end{bmatrix}, \quad C^{(r)} = \begin{bmatrix} \mathcal{A}_i & -B_{12} & 0 \\ 0 & 0 & 0 \\ 0 & 0 & -I \end{bmatrix}, \quad \text{and}$$
$$D = \begin{bmatrix} 0 & 0 & 0 \\ 0 & R_1^{-1} & 0 \\ 0 & 0 & 0 \end{bmatrix}. \tag{2.53}$$

Here, R_1 denotes the diagonal matrix the entries of which are the resistances of the resistors connected directly to the eliminated voltage sources, and the partitions in (2.53) are conforming with the partitions of the input and output vectors (2.52).

To obtain a reduced-order model of the descriptor system (2.51), we can again employ SPRIM. The data matrices of the reduced system are obtained analogous to (2.31), together with the additional relations

$$C_n := V_n^T C^{(r)} \quad \text{and} \quad D_n := D.$$

It is easy to see that SPRIM applied to the descriptor system (2.51) preserves the structures of all the data matrices.

2.6 Numerical Examples

In this section, we present some results for two classes of circuit examples. In all cases, we ran the SPRIM algorithm for increasing values of the dimension $\hat{n}$ of the underlying block Krylov subspaces and stopped when the transfer function of the SPRIM reduced-order model had converged to the transfer function of the unreduced original descriptor system. Here, convergence is monitored over a relevant frequency range of interest of the form

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

where $\mathtt{i} = \sqrt{-1}$. For the value of $\hat{n}$ at convergence of SPRIM, we also generated the corresponding PRIMA reduced-order model produced from the same $\hat{n}$-dimensional block Krylov subspace. This is a fair comparison since generating basis vectors for this subspace is the dominating computational cost for both SPRIM and PRIMA. In all cases, we plot the absolute values of $H(s)$ and $H_n(s)$ (for both SPRIM and PRIMA) over the frequency range of interest.

The first example is a variant of the so-called PEEC circuit [29]. It only has state-space dimension $N = 308$, but due to its many poles and zeros close to the frequency range of interest, its transfer function has many features. This variant of the PEEC circuit has two current sources and one voltage source, and thus $m = 3$. The expansion point $s_0 = \pi \times 10^{10}$ was used. In this case, the Krylov dimension $\hat{n} = 90$ is needed to achieve convergence. Figure 2.3 depicts the absolute values of the (1,1)-component of the 3×3-matrix-valued transfer functions. Clearly, for $\hat{n} = 90$ PRIMA has not converged yet. Figure 2.4 shows a close-up of the subrange where the PRIMA and SPRIM reduced-order models differ the most. Figures 2.5 and 2.6 display the corresponding plots for the (1,3)-component of the 3×3-matrix-valued transfer functions.

The second example (referred to as "package example") is a larger RCL circuit with state-space dimension $N = 1841$. This circuit has 8 current sources and 8 voltage sources, and thus $m = 16$. Its transfer function is 16×16-matrix-valued

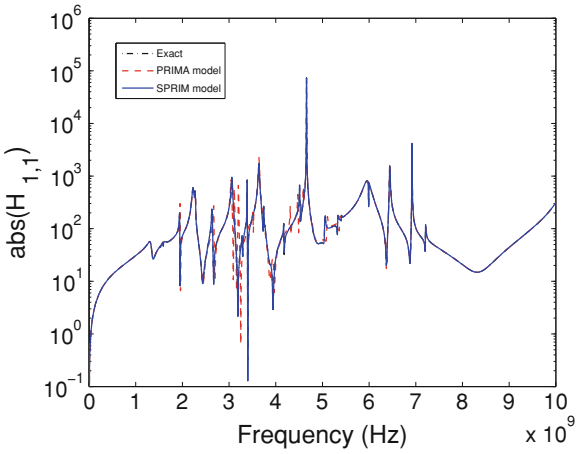

Fig. 2.3 PEEC example, (1,1)-component of transfer functions

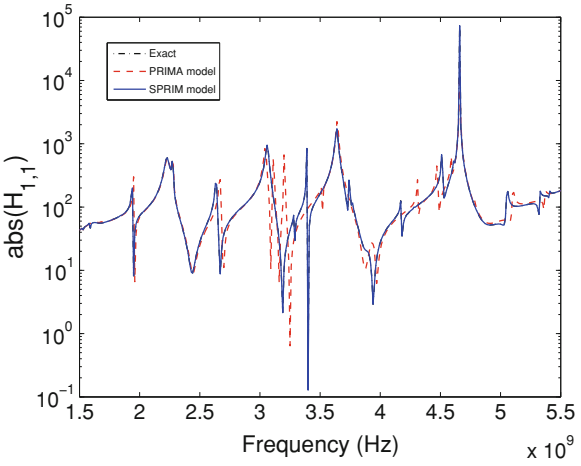

Fig. 2.4 PEEC example, close-up of (1,1)-component of transfer functions

and has 256 components. The expansion point $s_0 = 2\pi \times 10^{10}$ was used. For this example, the Krylov dimension $\hat{n} = 128$ is needed to achieve convergence. Figures 2.7 and 2.8 depict 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.

The 8 voltage sources of the package example are all of the type shown in Fig. 2.2, and so all voltage sources can be eliminated using the approach outlined in Sect. 2.5. We have applied SPRIM and PRIMA to the resulting descriptor system (2.51) of state-space dimension $N^{(r)} = 1841-16 = 1825$. As before, the Krylov dimension $\hat{n} = 128$ is needed to achieve convergence. Figure 2.9 shows the absolute values of the (16,9)-component of the transfer functions.

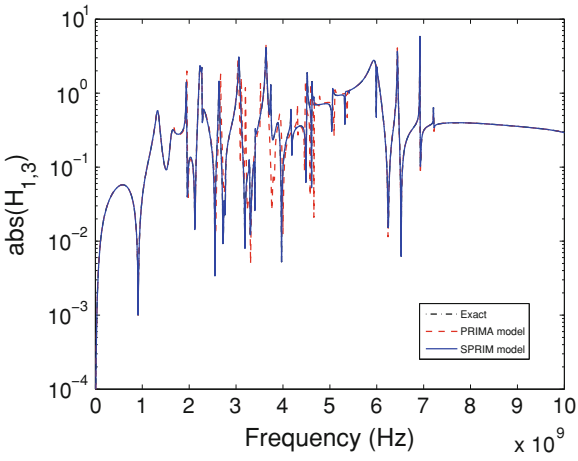

Fig. 2.5 PEEC example, (1,3)-component of transfer functions

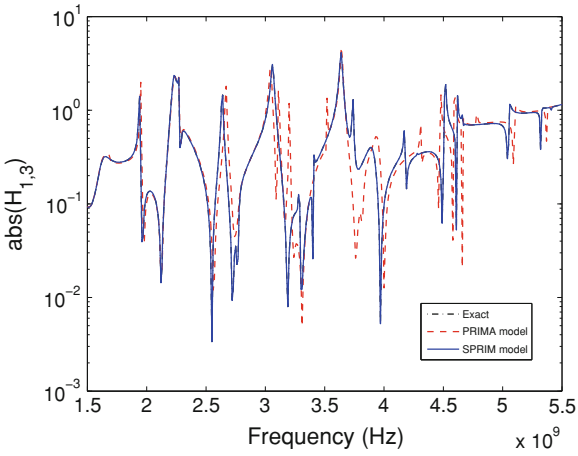

Fig. 2.6 PEEC example, close-up of the (1,3)-component of the transfer functions

2.7 Concluding Remarks

In this paper, we reviewed the formulation of general RCL circuits as descriptor systems and described the SPRIM reduction algorithm for general RCL circuits.

While there has been a lot of progress in Krylov subspace-based order reduction of large-scale RCL circuits in recent years, there are still many open problems. All 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

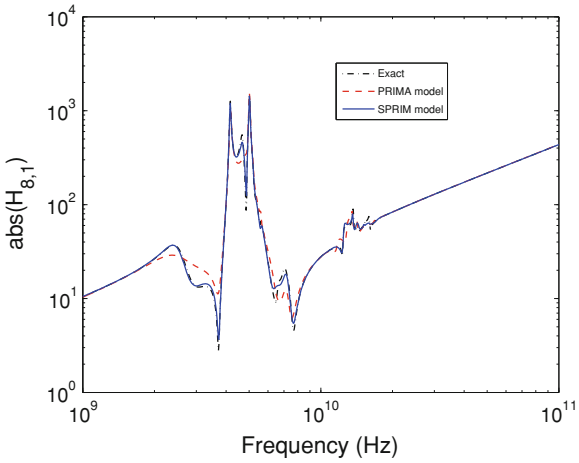

Fig. 2.7 Package example, (8,1)-component of transfer functions

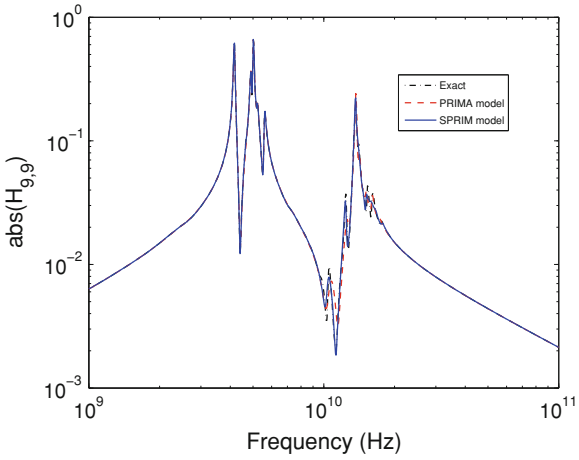

Fig. 2.8 Package example, (9,9)-component of transfer functions

such explicit projections. First, it requires the storage of the basis matrix, which becomes prohibitive in the case of truly large-scale linear dynamical systems. 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 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 suitable expansion points s_0 and robust and reliable stopping criteria and error bounds.

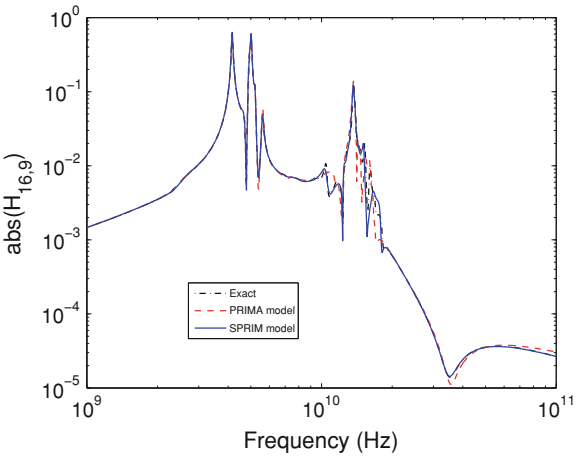

Fig. 2.9 Package example with voltage sources eliminated, (16,9)-component of transfer functions

Acknowledgements This work was supported in part by the National Science Foundation through Grant DMS-0613032. The author is grateful to the two anonymous referees for their careful reading of the original manuscript and their constructive comments that helped to improve the presentation of the paper.

References

1. Anderson, B.D.O., Vongpanitlerd, S.: Network Analysis and Synthesis. Prentice-Hall, Englewood Cliffs, New Jersey (1973)
2. Arnoldi, W.E.: The principle of minimized iterations in the solution of the matrix eigenvalue problem. Quart. Appl. Math. **9**, 17–29 (1951)
3. Bai, Z., Feldmann, P., Freund, R.W.: How to make theoretically passive reduced-order models passive in practice. In: Proceedings of IEEE 1998 Custom Integrated Circuits Conference, pp. 207–210. IEEE, Piscataway (1998)
4. Baker, Jr., G.A., Graves-Morris, P.: Padé Approximants, 2nd edn. Cambridge University Press, New York (1996)
5. de Villemagne, C., Skelton, R.E.: Model reductions using a projection formulation. Int. J. Control **46**(6), 2141–2169 (1987)
6. Deo, N.: Graph Theory with Applications to Engineering and Computer Science. Prentice-Hall, Englewood Cliffs (1974)
7. Elfadel, I.M., Ling, D.D.: Zeros and passivity of Arnoldi-reduced-order models for interconnect networks. In: Proceedings of 34nd ACM/IEEE Design Automation Conference, pp. 28–33. ACM, New York (1997)
8. 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 (1994)
9. Feldmann, P., Freund, R.W.: Efficient linear circuit analysis by Padé approximation via the Lanczos process. IEEE Trans. Comput. Aided Des. **14**, 639–649 (1995)
10. Feldmann, P., Freund, R.W.: Reduced-order modeling of large linear subcircuits via a block Lanczos algorithm. In: Proceedings 32nd ACM/IEEE Design Automation Conference, pp. 474–479. ACM, New York (1995)

11. Feldmann, P., Freund, R.W.: Interconnect-delay computation and signal-integrity verification using the SyMPVL algorithm. In: Proceedings of 1997 European Conference on Circuit Theory and Design, pp. 132–138. IEEE Computer Society Press, Los Alamitos (1997)
12. Freund, R.W.: Reduced-order modeling techniques based on Krylov subspaces and their use in circuit simulation. In: Datta, B.N. (ed.) Applied and Computational Control, Signals, and Circuits, vol. 1, pp. 435–498. Birkhäuser, Boston (1999)
13. Freund, R.W.: Krylov-subspace methods for reduced-order modeling in circuit simulation. J. Comput. Appl. Math. **123**(1–2), 395–421 (2000)
14. Freund, R.W.: Model reduction methods based on Krylov subspaces. Acta Numerica **12**, 267–319 (2003)
15. Freund, R.W.: SPRIM: structure-preserving reduced-order interconnect macromodeling. In: Technical Digest 2004 IEEE/ACM International Conference on Computer-Aided Design, pp. 80–87. IEEE Computer Society Press, Los Alamitos (2004)
16. Freund, R.W.: On Padé-type model order reduction of *J*-Hermitian linear dynamical systems. Linear Algebra Appl. **429**, 2451–2464 (2008)
17. Freund, R.W.: Structure-preserving model order reduction of RCL circuit equations. In: Schilders, W., van der Vorst, H., Rommes, J. (eds.) Model Order Reduction: Theory, Research Aspects and Applications, Mathematics in Industry, vol. 13, pp. 49–73. Springer-Verlag, Berlin (2008)
18. Freund, R.W., Feldmann, P.: Reduced-order modeling of large passive linear circuits by means of the SyPVL algorithm. In: Technical Digest 1996 IEEE/ACM International Conference on Computer-Aided Design, pp. 280–287. IEEE Computer Society Press, Los Alamitos (1996)
19. Freund, R.W., Feldmann, P.: The SyMPVL algorithm and its applications to interconnect simulation. In: Proceedings of 1997 International Conference on Simulation of Semiconductor Processes and Devices, pp. 113–116. IEEE, Piscataway (1997)
20. Freund, R.W., Feldmann, P.: Reduced-order modeling of large linear passive multi-terminal circuits using matrix-Padé approximation. In: Proceedings of Design, Automation and Test in Europe Conference 1998, pp. 530–537. IEEE Computer Society Press, Los Alamitos (1998)
21. Grimme, E.J.: Krylov projection methods for model reduction. Ph.D. thesis, Department of Electrical Engineering, University of Illinois at Urbana-Champaign, Urbana-Champaign (1997)
22. 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)
23. Lanczos, C.: An iteration method for the solution of the eigenvalue problem of linear differential and integral operators. J. Res. Nat. Bur. Standards **45**, 255–282 (1950)
24. Odabasioglu, A.: Provably passive RLC circuit reduction. M.S. thesis, Department of Electrical and Computer Engineering, Carnegie Mellon University (1996)
25. Odabasioglu, A., Celik, M., Pileggi, L.T.: PRIMA: passive reduced-order interconnect macromodeling algorithm. In: Technical Digest 1997 IEEE/ACM International Conference on Computer-Aided Design, pp. 58–65. IEEE Computer Society Press, Los Alamitos (1997)
26. Odabasioglu, A., Celik, M., Pileggi, L.T.: PRIMA: passive reduced-order interconnect macromodeling algorithm. IEEE Trans. Comput. Aided Des. **17**(8), 645–654 (1998)
27. Pillage, L.T., Rohrer, R.A.: Asymptotic waveform evaluation for timing analysis. IEEE Trans. Comput. Aided Des. **9**, 352–366 (1990)
28. Pillage, L.T., Rohrer, R.A., Visweswariah, C.: Electronic circuit and system simulation methods. McGraw-Hill, Inc., New York (1995)
29. Ruehli, A.E.: Equivalent circuit models for three-dimensional multiconductor systems. IEEE Trans. Microw. Theory Tech. **22**, 216–221 (1974)
30. Ruehli, A.E. (ed.): Circuit Analysis, Simulation, and Design Part 1: General Aspects of Circuit Analysis and Design. North-Holland, Amsterdam (1986)

31. Silveira, L.M., Kamon, M., Elfadel, I., White, J.: A coordinate-transformed Arnoldi algorithm for generating guaranteed stable reduced-order models of RLC circuits. In: Technical Digest 1996 IEEE/ACM International Conference on Computer-Aided Design, pp. 288–294. IEEE Computer Society Press, Los Alamitos (1996)
32. Vlach, J., Singhal, K.: Computer Methods for Circuit Analysis and Design, 2nd edn. Van Nostrand Reinhold, New York (1994)