Model order reduction for multi-terminals systems : with applications to circuit simulation

Model Order Reduction for Multi-terminal Systems

with

Applications to Circuit Simulation

This work has has been carried out under a dual doctorate agreement between Jacobs University - Bremen, Germany and Technische Universiteit Eindhoven - Eindhoven, The Netherlands, in collaboration with NXP Semiconductors - Eindhoven, The Netherlands.

A catalogue record is available from the Eindhoven University of Technology Library
ISBN 978-90-386-2580-5

Model Order Reduction for Multi-terminals Systems with Applications to Circuit Simulation

PROEFSCHRIFT

ter verkrijging van de graad van doctor aan de
Technische Universiteit Eindhoven, op gezag van de
rector magnificus, prof.dr.ir. C.J. van Duijn, voor een
commissie aangewezen door het College
voor Promoties in het openbaar te verdedigen
op maandag 26 september 2011 om 16.00 uur

door

Roxana Ionuţiu

geboren te Oradea, Roemenië

Dit proefschrift is goedgekeurd door de promotoren:

prof.dr. W.H.A. Schilders
en
prof.dr. A.C. Antoulas

Copromotor:
dr. J. Rommes

Contents

Chapter 1

Introduction

1.1 Motivation

Over the past decades, developments in microelectronics have followed the path predicted by the american scientist and Intel co-founder, Gordon E. Moore, who, already back in the early days of the integrated circuit, extrapolated that the number of transistors that can be packed onto a chip of silicon would double approximately every two years. What became known as *Moore's Law* has since dictated the speed with which the complexity of integrated circuits increases and with that, the rate at which the price of electronics goes down. From a circuit design perspective however, more transistors per silicon area means that components are more densely packed together, and that the behaviors of different parts of the chip are no longer independent. The design of current integrated circuits is therefore hampered by parasitic electromagnetic effects that strongly influence the behavior of the device. During the physical verification of circuit designs, it is thus vital to take these parasitic effects into account. This requires simulation of large scale electrical networks, with numbers of nodes and electronic circuit components (resistors, capacitors, inductors) exceeding hundreds of thousands. Standard circuit simulation tools are often insufficient for this task, as they may be unable to compute solutions to the *differential algebraic systems* of very high order underlying these circuits. To provide a solution to this challenging industrial problem, a new methodology is developed in this thesis, which combines techniques from electrical engineering, numerical linear algebra, and graph theory.

Model order reduction (MOR) aims at constructing a model of lower dimension than an original system while well approximating its behavior. For instance, if the original system is an electrical circuit, a reduced circuit is sought which is to replace the original one within the desired simulation setup. When this is achieved, more time and memory efficient simulations can be performed with the available computing resources. Model re-

duction however involves a mathematical procedure, through which usually the physical interpretation of the original system is lost. That is, if the original system is, say, an electrical circuit, the reduction returns a smaller mathematical model, rather than an actual circuit. The *synthesis* problem then arises, that of mapping the reduced mathematical model back into an equivalent electrical circuit. Also, as different parts of a circuit design often need to be analyzed separately, another important question is whether these can be reduced independently and re-assembled together in a simple way. This seems to be a natural feature, however is not directly satisfied with traditional reduction approaches, because the physical meaning of interconnection points between different parts (also called *terminals*) is lost. A distinguishing feature of the methods developed in this thesis is their ability to convert reduced models into circuit representations which are easily re-coupled to one another and re-used within the original simulation setup.

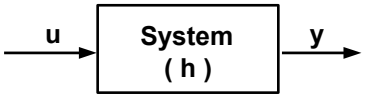
Yet another critical challenge arises when the the original systems are so large, that even the process of reduction itself is hampered by limited CPU and memory resources. The emerging question is then: how does one reduce a system that is too big to "fit" within the available reduction methodology? This scenario is especially relevant when reducing circuits with a very large number of *terminal* nodes, for which *preserving sparsity* is a critical requirement. Here, these problems are overcome with the help of *graph partitioning* and *fill-reducing node reorderings*. Very large, multi-terminal networks are reduced in a divide-and-conquer manner, while the behavior of the reduced circuit remains close to that of the original. The reduced circuits thus obtained contain much fewer nodes and elements than otherwise obtained from conventional reduction techniques and, as industrial examples show, allow up to 50 times faster simulations at little loss of accuracy. In addition, the proposed multi-terminal model reduction methods make circuit simulations possible for designs which could not be handled in their original dimension by SPICE-like tools.

While the applications in this thesis come from the electronics industry, many of the problems addressed here are fundamental, and the solutions proposed could resolve similar issues arising in other disciplines. Whether they are models of electrical circuits, mechanical systems, neuronal networks in the brain, or links between web-pages, one can always benefit from numerical algorithms that can approximate their functionality efficiently, either by parts or in whole, enabling a simpler and faster understanding of their behavior.

The following sections review some general facts and established results concerning linear dynamical systems and their reduction. The material is based mostly on the references [72], [4], [78], with some adaptations pertaining to the context of this thesis. Then, an overview of the remaining chapters is provided.

1.2 Preliminaries from linear system theory

Dynamical systems arise in various disciplines: chemical processing, biomedical engineering, acoustics or circuit design and can describe different physical processes. They share however some common features. To quote [72], a system "can be viewed as a process in which input signals are transformed by the system or cause the system to respond in a certain way, resulting in other signals as outputs". For instance, an electrical circuit is a system which produces some voltages or currents, in response to applied currents or voltages. Fig. 1.1 gives a simple representation of a linear system with driving inputs $\mathbf{u}(t)$ and observed outputs $\mathbf{y}(t)$. The word *dynamic* refers to the fact that



$$\mathbf{y}(t) = \mathbf{h} * \mathbf{u}(t) = \int_{-\infty}^{t} \mathbf{h}(t-\tau)\mathbf{u}(\tau)d\tau, \quad (1.1)$$

Figure 1.1: *System with input* $\mathbf{u}$ *and output* $\mathbf{y}$*. The output is the convolution of the input with the impulse response* $\mathbf{h}(t)$*.*

the system *has memory*, i.e., the system's future behavior depends on its past evolution. An electrical network with capacitors or inductors is a dynamical system, where the memory has a physical interpretation related to the storage of energy; for example, the capacitor stores energy by accumulating electrical charge. The voltage across a capacitor is thus an integral of the current, namely:

$$y(t) = \frac{1}{C}\int_{-\infty}^{t} u(\tau)d\tau. \quad (1.2)$$

Whichever the underlying application, the systems which fit in the reduction framework of this thesis are linear, dynamic, continuous, time-invariant[1], and share a common mathematical description, presented next.

1.2.1 External description

The *external description* characterizes a system by means of the input variables $\mathbf{u}$ and the outputs $\mathbf{y}$, related through the convolution operation (1.1), where $\mathbf{h}(t)$ is the system's *impulse response* (the response of the system to the Dirac delta input, denoted here informally as the unit impulse function $\delta(t) = \begin{cases} 1, t = 0 \\ 0, t \neq 0 \end{cases}$ [2]).

[1] Continuous refers to the fact that continuous-time input signals result in continuous-time signals at the outputs, while time-invariance refers to the system's behavior being fixed over time (for an electrical circuit, this holds if the resistances/capacitors/inductors are constant over time).

[2] See [21,83] for formal definitions of the Dirac delta distribution.

1.2.2 Internal description

In addition to the inputs and outputs one can use the *internal variables* to characterize a dynamical system. The internal variable is by definition the least amount of information required at time $t = t_0$ so that, together with the excitation for $t > t_0$, one can compute the future behavior of the system. This **input/internal-variable/output** representation is called the *internal description* of the system and is governed by a set of differential-algebraic equations:

$$\boldsymbol{\Sigma} \begin{cases} \mathbf{E}\dot{\mathbf{x}}(t) &= \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \\ \mathbf{y}(t) &= \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t) \end{cases}, \tag{1.3}$$

where $\mathbf{u} \in \mathbb{R}^p$ is the input of the system, $\mathbf{x} \in \mathbb{R}^n$ are the internal variables, $\mathbf{y} \in \mathbb{R}^m$ is the output (the variables observed), and $\mathbf{E} \in \mathbb{R}^{n\times n}$, $\mathbf{A} \in \mathbb{R}^{n\times n}$, $\mathbf{B} \in \mathbb{R}^{n\times p}$, $\mathbf{C} \in \mathbb{R}^{m\times n}$, $\mathbf{D} \in \mathbb{R}^{m\times p}$ are the system matrices. The first equation from (1.3) is the *state equation* which describes the system's dynamics, and the second is the *output equation* which describes the observation. The number of internal variables n is the *dimension* of $\boldsymbol{\Sigma}$. If $m, p > 1$, the system is called multiple-input multiple-output (MIMO), and if $m = p = 1$ it is called single-input single-output (SISO). In the most general case which also covers the scenario of a singular $\mathbf{E}$, system $\boldsymbol{\Sigma}$ is a *descriptor system* [68,76], while for the particular case of $\mathbf{E}$ invertible, it is a *state space system*, and the internal variables $\mathbf{x}$ are called *states* [4]. Systems describing electrical circuits often have a special structure (see Sect. 1.3.3), in particular $\mathbf{C} = \mathbf{B}^T$ and $\mathbf{D} = \mathbf{0}$.

It turns out that the external and internal descriptions are intimately related. In particular, using the Laplace transform [72], (1.3) is expressed in the frequency domain, where the inputs, outputs, and internal variables as a function of frequency s are: $\mathbf{U}(s)$, $\mathbf{Y}(s)$, and $\mathbf{X}(s)$ respectively. More precisely, the Laplace transform of $\mathbf{x}(t)$ is $\mathbf{X}(s) = (s\mathbf{E} - \mathbf{A})^{-1}\mathbf{x}(0^-) + (s\mathbf{E} - \mathbf{A})^{-1}\mathbf{B}\mathbf{U}(s)$ and it is assumed that the initial condition is $\mathbf{x}(0^-) = 0$ [4, Chapter 4]. Eliminating now the states $\mathbf{X}(s)$ from the first equation of (1.3) and replacing them in the second, one obtains the system's *transfer function*:

Definition 1.2.1 *The transfer function of the dynamical system (1.3) is defined as:*

$$\mathbf{H}(s) = \mathbf{C}(s\mathbf{E} - \mathbf{A})^{-1}\mathbf{B} + \mathbf{D} \quad \in \mathbb{C}^{m\times p}. \tag{1.4}$$

$\mathbf{H}(s)$ describes the input/output behavior of the system in the frequency domain, and is precisely the Laplace transform of the impulse response $\mathbf{h}(t)$, introduced previously in the external description. For the transfer function to be well defined, the pencil $(\mathbf{A}, \mathbf{E})$ must be regular[3] [a matrix pair $(\mathbf{A}, \mathbf{E})$ is regular if there exists at least one $\lambda \in \mathbb{C}$ such that $det(\mathbf{A} - \lambda\mathbf{E}) \neq 0$]. Most of the existing model reduction methods rest on this assumption however, the methods developed in chapters 3, 4, 5, and 6 also apply to a

[3] For electrical networks, the pencil is regular if the network is made consistent by grounding one of the nodes.

particular type of singular pencils, mainly those describing un-grounded electrical networks.

The *poles* of a system lie at the heart of its behavior, in particular they determine its *stability* (see Sect. 1.2.3). They are defined as follows:

Definition 1.2.2 *The poles of the transfer function* $\mathbf{H}(s)$ *are the points* $\lambda \in \mathbb{C}$ *for which* $\lim_{s\to\lambda} \|\mathbf{H}(s)\|_s = \infty$, *i.e., the generalized eigenvalues of the pair* $(\mathbf{A}, \mathbf{E})$[4].

1.2.3 Stability and passivity

Original systems $\mathbf{\Sigma}$ describing electrical circuits are stable and passive; it is hence desirable for the reduced $\widetilde{\mathbf{\Sigma}}$ to be also stable and passive. These are among the most important system properties that should be preserved ideally by *any* reduction method.

Stability

In the external description, the system $\mathbf{\Sigma}$ characterized by the convolution (1.1) is *bounded -input, bounded-output stable* if any bounded input $\mathbf{u}(t)$ results in a bounded output $\mathbf{y}(t)$. This has an equivalent interpretation via the internal description, namely that a system is stable if and only if all poles lie in the closed left half of the complex plane (all poles have non-positive real parts and all pure imaginary eigenvalues have multiplicity one) [4, Theorem 5.10].

Passivity

Passive systems are those which do not generate energy. According to [17], the system (1.3) is passive if there exists a non-negative-valued function $\Theta \in \mathbb{R}^n \to \mathbb{R}_+$, such that:

$$\Theta(\mathbf{x}(t_1)) - \Theta(\mathbf{x}(t_0)) \leq \left[\int_{t_0}^{t_1} \mathbf{u}^T(\tau)\mathbf{y}(\tau)d\tau\right], \tag{1.5}$$

for $\forall\ t_0, t_1 \in \mathbb{R}$, $t_1 \geq t_0$, and all trajectories $(\mathbf{u}, \mathbf{x}, \mathbf{y})$ which satisfy the system equations (1.3). If it exists, Θ is called a storage function. The interpretation of equation (1.5) is that, the change in internal storage $\Theta(\mathbf{x}(t_1)) - \Theta(\mathbf{x}(t_0))$ can never exceed what is supplied to the system [4, Chapter 5.9]. Electrical networks with passive components (resistors, capacitors, inductors and ideal transformers) are passive systems; they absorb supply (energy) in the form of electrical power, which is the sum of the product of the voltage and current at the external ports: $\mathbf{u}^T(t)\mathbf{y}(t) = \Sigma_k V_k I_k$.

[4] For $\mathbf{A} \in \mathbb{C}^{n\times n}$, $\mathbf{E} \in \mathbb{C}^{n\times n}$, λ is a generalized eigenvalue of $(\mathbf{A}, \mathbf{E})$, if $\exists \mathbf{x} \in \mathbb{C}^n, \mathbf{x} \neq \mathbf{0}$ such that $\mathbf{A}\mathbf{x} = \lambda\mathbf{E}\mathbf{x}$.

An established result which characterizes passivity in a unifying manner is that a descriptor system $\mathbf{\Sigma}$ with $m = p$ is passivie if and only if its transfer function $\mathbf{H}(s)$ is positive real [3,76]. By definition [17]:

Definition 1.2.3 *A rational function $\mathbf{H}(s) \in \mathbb{C}^{m\times m}$ is positive real if and only if it satisfies both of the following conditions:*

1. $\mathbf{H}(s)$ *is analytic in* $\mathbb{C}_+$
2. $\mathbf{H}(s) + \mathbf{H}^H(s) \geq 0$, *for all* $s \in \mathbb{C}_+$[5].

[4, Theorem 5.22] gives a complete characterization of positive realness in terms of the external representation of $\mathbf{\Sigma}$.

By ensuring that the positive realness condition remains satisfied after reduction, several passivity preserving model reduction methods have been developed. Among those which make no assumptions about the structure of the underlying system matrices are [77] based on balancing and [44] based on moment matching. Other model-reduction methods [27,71,89] exploit the special structure and matrix properties of descriptor systems describing electrial circuits to preserve passivity. In short, starting from an original system in a *passive form* [57] the reduced models retain the passive form. To this type of methods belong also the ones developed in this thesis. Passive systems are also stable [4, Theorem 5.22], [57], hence all reduction methods which preserve passivity are implicitly stability preserving.

1.3 Model order reduction

At the heart of model reduction lies the desire to approximate the behavior of a large dynamical system in an efficient manner, so that the resulting approximation error is small. Other requirements are: the preservation of important system properties, of its physical interpretation, and an efficient implementation. In short, starting from an n-dimensional system $\mathbf{\Sigma}$, a *reduced* k-dimensional system $\widetilde{\mathbf{\Sigma}}$ is sought:

$$\widetilde{\mathbf{\Sigma}} \left\{ \begin{array}{rcl} \widetilde{\mathbf{E}}\dot{\widetilde{\mathbf{x}}}(t) & = & \widetilde{\mathbf{A}}\widetilde{\mathbf{x}}(t) + \widetilde{\mathbf{B}}\mathbf{u}(t) \\ \widetilde{\mathbf{y}}(t) & = & \widetilde{\mathbf{C}}\widetilde{\mathbf{x}}(t) + \mathbf{D}\mathbf{u}(t) \end{array} \right. , \tag{1.6}$$

where $k \ll n$, so that the output approximation error $\|\mathbf{y}(t) - \tilde{\mathbf{y}}(t)\|$ (in appropriate norm) is small. Through reduction, the number of inputs and outputs is the same as in the reduced model, however the internal variables and the system matrices are reduced in dimension: $\widetilde{\mathbf{x}} \in \mathbb{R}^k$, $\widetilde{\mathbf{E}} \in \mathbb{R}^{k\times k}$, $\widetilde{\mathbf{A}} \in \mathbb{R}^{k\times k}$, $\widetilde{\mathbf{B}} \in \mathbb{R}^{k\times p}$, $\widetilde{\mathbf{C}} \in \mathbb{R}^{m\times k}$. The inputs of the

[5] For $s = \sigma + j\omega$, and $\mathbf{H}(s)$ defined as in (1.4), the Hermitian complex conjugate reads: $(\mathbf{H}(s))^H = \mathbf{B}^H((\sigma - j\omega)\mathbf{E}^H - \mathbf{A}^H)^{-1}\mathbf{C}^H + \mathbf{D}^H$.

reduced system are the same as for the original. The transfer function of $\widetilde{\mathbf{\Sigma}}$ is:

$$\widetilde{\mathbf{H}}(s) = \widetilde{\mathbf{C}}(s\widetilde{\mathbf{E}} - \widetilde{\mathbf{A}})^{-1}\widetilde{\mathbf{B}} + \mathbf{D} \quad \in \mathbb{C}^{m\times p}. \tag{1.7}$$

The unifying approach for obtaining a reduced model from an original system is via a Petrov-Galerkin[6] projection [6]:

$$\widetilde{\Sigma}(\widetilde{\mathbf{E}}, \widetilde{\mathbf{A}}, \widetilde{\mathbf{B}}, \widetilde{\mathbf{C}}, \widetilde{\mathbf{D}}) \equiv (\mathbf{W}^*\mathbf{E}\mathbf{V}, \mathbf{W}^*\mathbf{A}\mathbf{V}, \mathbf{W}^*\mathbf{B}, \mathbf{C}\mathbf{V}, \widetilde{\mathbf{D}}), \tag{1.8}$$

where $\mathbf{V}, \mathbf{W} \in \mathbb{R}^{n\times k}$ are matrices whose $k \ll n$ columns form bases for the relevant subspaces pertaining to the reduction method chosen. In this projection framework it is common to set $\widetilde{\mathbf{D}} = \mathbf{D}$, but other scenarios are possible, as described in [6].

1.3.1 A classification

The governing principle behind all reduction methods is that, after a suitable decomposition is found, the non-dominant[7] internal variables are eliminated from the system. Reduction methods differ in the way the decomposition is performed. This in turn dictates how the projection matrices $\mathbf{V}$ and $\mathbf{W}$ are constructed. Roughly speaking, reduction methods are classified into (a) spectral-, and (b) Krylov-based. Among the volumes which give a comprehensive coverage of the various methods are [4, 12, 82]. A comparison of spectral and Krylov-based methods is available in [39]. By no means attempting to give an exhaustive literature review, a few well-know methods are mentioned here, with a stronger emphasis on Krylov-based methods. The latter are the foundation for the multi-terminal reduction methodology in this thesis.

Spectral methods

Among the *spectral methods*, one distinguishes between those based on the SVD (singular value decomposition) or the EVD (eigenvalue decomposition) of relevant system quantities. From the SVD methods, *balanced truncation* [11] and *positive real balanced truncation* [74, 77] use SVD-based projections to reduce a *balanced* representation of the system, i.e. one where the observability and controllability Gramians are equal and diagonal [4, Chapter 7]. Compared to balancing, which uses both the observability and controllabilty gramians as the basis for decomposition, *Poor man's TBR* [75] decomposes only *one* of the system gramians. *Proper orthogonal decomposition (POD)* [4, Chapter 9.1] constructs the reducing projection based on the SVD of a matrix of *time snapshots* (samples of the state computed at given time instants). Among the EVD methods, *modal approximation* [78] constructs a reduced model which interpolates *dominant poles* of the original system, based on the generalized eigenvalue decomposition of the pair $(\mathbf{A}, \mathbf{E})$.

[6] When $\mathbf{W} = \mathbf{V}$ the projection is of Galerkin type.

[7] Non-dominant here refers to internal variables which contribute the least to the system's response.

Krylov-based methods

Krylov-based reduction methods exploit the use of Krylov subspace iterations to achieve system approximation by *moment matching* (explained in Sect. 1.3.2). Among these are PRIMA [71], the structure preserving version SPRIM [27], the second order variants SOAR [9] and SAPOR [66,89], the spectral zero method (SZM) [5,44,45,86] and optimal $\mathcal{H}_2$ [32]. Next, the concept of reduction by moment matching is explained.

1.3.2 Model reduction by moment matching

The starting point for reduction by moment matching is the desire to approximate a transfer function $\mathbf{H}(s)$ by a rational function of lower degree, $\widetilde{\mathbf{H}}(s)$. The questions are then: (a) what are the coefficients of a reduced $\widetilde{\mathbf{H}}(s)$ which accurately approximates $\mathbf{H}(s)$ and, once these are identified, (b) how are the projections $\mathbf{V}$, $\mathbf{W}$ constructed, so that the reduced model $\widetilde{\mathbf{\Sigma}}$ (2.1) is charcterized precisely by the $\widetilde{\mathbf{H}}(s)$ with the desired coefficients?

The answer to (a) is found by constructing $\widetilde{\mathbf{H}}(s)$ to match some terms of the Laurent series expansion of $\mathbf{H}(s)$ at various points of the complex plane. In particular, the k'th *moment* of $\mathbf{H}(s)$ at $s_0 \in \mathbb{C}$ is the k'th derivative of $\mathbf{H}(s)$ evaluated at $s = s_0$:

$$\eta_k(s_0) = (-1)^k \frac{d^k}{ds^k}\mathbf{H}(s)\Big|_{s=s_0} \in \mathbb{R}^{m\times p},\ k \geq 0, \tag{1.9}$$

so the Laurent expansion of $\mathbf{H}(s)$ around $s_0 \in \mathbb{C}$ is:

$$\mathbf{H}(s) = \eta_0(s_0) + \eta_1(s_0)\frac{(s-s_0)}{1!} + \eta_2(s_0)\frac{(s-s_0)^2}{2!} + \ldots + \eta_k(s_0)\frac{(s-s_0)^k}{k!} + \ldots \tag{1.10}$$

Model reduction by moment matching thus amounts to finding, given the original $\mathbf{\Sigma}$, a reduced model $\widetilde{\mathbf{\Sigma}}$ with transfer function $\widetilde{\mathbf{H}}(s)$, such that $\widetilde{\mathbf{H}}(s)$ has the same moments as $\mathbf{H}(s)$ up to a desired number k. More precisely:

$$\begin{aligned} \widetilde{\mathbf{H}}(s) &= \widetilde{\eta}_0(s_0) + \widetilde{\eta}_1(s_0)\frac{(s-s_0)}{1!} + \widetilde{\eta}_2(s_0)\frac{(s-s_0)^2}{2!} + \ldots + \widetilde{\eta}_k(s_0)\frac{(s-s_0)^k}{k!} + \ldots, \\ \text{and } \widetilde{\eta}_i &= \eta_i,\ i = 1\ldots k. \end{aligned} \tag{1.11}$$

From (1.9), the moments can be expressed directly in terms of the system matrices, as derivatives of (1.4) for the original system [and (1.7) for the reduced system]. In particular, introducing the following notation:

$$\mathcal{A} = -(s_0\mathbf{E} - \mathbf{A})^{-1}\mathbf{E}, \quad \mathcal{R} = (s_0\mathbf{E} - \mathbf{A})^{-1}\mathbf{B}, \tag{1.12}$$

$$\widetilde{\mathcal{A}} = -(s_0\widetilde{\mathbf{E}} - \widetilde{\mathbf{A}})^{-1}\widetilde{\mathbf{E}}, \quad \widetilde{\mathcal{R}} = (s_0\widetilde{\mathbf{E}} - \widetilde{\mathbf{A}})^{-1}\widetilde{\mathbf{B}} \tag{1.13}$$

the moments (1.9) for the original and reduced systems respectively are:

$$\eta_0(s_0) = \mathbf{C}\mathcal{R} + \mathbf{D}, \quad \eta_i(s_0) = [i!(-1)^i]\mathbf{C}\mathcal{A}^i\mathcal{R}, \ i \geq 1, \quad \text{and} \tag{1.14}$$

$$\widetilde{\eta}_0(s_0) = \widetilde{\mathbf{C}}\widetilde{\mathcal{R}} + \mathbf{D}, \quad \widetilde{\eta}_i(s_0) = [i!(-1)^i]\widetilde{\mathbf{C}}\widetilde{\mathcal{A}}^i\widetilde{\mathcal{R}}, \ i \geq 1. \tag{1.15}$$

The general case of matching moments around arbitrary points $s_0 \in \mathbb{C}$ is called *rational interpolation*. Two special cases are when the expansion is around $s = 0$ (*Padé approximation*) and $s = \infty$ (*partial realization*, where the moments are the *Markov parameters* of the system). The Laurent expansion and moments at $s = 0$ are easily derived by setting $s_0 = 0$ in (1.10)-(1.11) and (1.12)-(1.13) respectively, and requiring $\mathbf{A}$ and $\widetilde{\mathbf{A}}$ to be invertible. For matching around $s = \infty$, the Laurent expansion and moments (Markov parameters)[8] are:

$$\mathbf{H}(s) = \eta_0(\infty) + \eta_1(\infty)s^{-1} + \eta_2(\infty)s^{-2} + \ldots + \eta_k(\infty)s^{-k} + \ldots \tag{1.16}$$

$$\eta_0(\infty) = \mathbf{D}, \quad \eta_i(\infty) = \mathbf{C}(\mathbf{E}^{-1}\mathbf{A})^{i-1}(\mathbf{E}^{-1}\mathbf{B}), \ i \geq 1, \tag{1.17}$$

similarly for the reduced transfer function, requiring $\mathbf{E}$, $\widetilde{\mathbf{E}}$ to be invertible.

An important result in reduction by moment matching is that reducing projections $\mathbf{W}$ and $\mathbf{V}$ can be constructed to ensure that a desired number of moments of $\mathbf{\Sigma}$ are matched by $\widetilde{\mathbf{\Sigma}}$. Following [4, Sect. 11.3] and [55, Theorem 2.1] (to which we refer for proofs) this result is repeated here, with some adaptations in notation to make the theory in this section uniform:

Theorem 1.3.1 *Let $\mathcal{A} \in \mathbb{R}^{n\times n}$, $\mathcal{R} \in \mathbb{R}^{n\times p}$ be the matrices from (1.12), $\mathbf{V} \in \mathbb{R}^{n\times k}$, $\mathbf{W} \in \mathbb{R}^{n\times k}$, $\mathbf{W}^T\mathbf{V} = \mathbf{I}$, $k < n$, $m \leq n$, $p \leq n$. If:*

$$\text{span}\left(\mathcal{A}^i\mathcal{R}\right) \subseteq \text{span}\left(\mathbf{V}\right), \ i \in (0, 1, \ldots, q_1 - 1), \ \text{and} \tag{1.18}$$

$$\text{span}\left(\left(\mathcal{A}^T\right)^i\mathbf{C}^T\right) \subseteq \text{span}\left(\mathbf{W}\right), \ i \in (0, 1, \ldots q_2 - 1), \tag{1.19}$$

then

$$\mathbf{C}\mathcal{A}^i\mathcal{R} = \widetilde{\mathbf{C}}\widetilde{\mathcal{A}}^i\widetilde{\mathcal{R}}, \ \text{for } i \in (0, 1, \ldots q_1 + q_2 - 1), \tag{1.20}$$

where :

$$\widetilde{\mathbf{C}} = \mathbf{C}\mathbf{V}, \quad \widetilde{\mathcal{A}} = \mathbf{W}^T\mathcal{A}\mathbf{V}, \ \widetilde{\mathcal{R}} = \mathbf{W}^T\mathcal{R}.$$

More precisely, $\widetilde{\mathbf{\Sigma}}$ obtained from the projection (2.1) matches $q_1 + q_2$ moments of the original system $\mathbf{\Sigma}$ at a chosen expansion point s_0.

In practice, $\mathbf{W}$ and $\mathbf{V}$ are not formed explicitly, due to the fact that computing the moments is numerically problematic [4]. Rather, exploiting the analogy between the $\mathbf{V}$ which satisfies (1.18) and the *Krylov subspace* $\mathbf{K}_q = \text{span}\left[\mathcal{R}, \mathcal{A}\mathcal{R}, \ldots, \mathcal{A}^{q_1-1}\mathcal{R}\right]$ (known

[8] The following notation is used: $\eta_k(\infty) = \lim_{s\to\infty}\eta_k(s)$.

as the reachability subspace in system theory, while **W** is associated with the observability subspace), **V** and **W** are constructed iteratively based on the Lanczos [58] or Arnoldi [7] algorithms. [4, Chapter 11.2] gives a detailed analogy between Lanczos/Arnoldi and moment matching, and discusses numerical issues such as the loss of orthogonality or break-down and the means to overcome them. One of the disadvantages of Lanczos/Arnoldi-based moment matching algorithms is that they do not automatically preserve the stability or the passivity of the original system (some variants do preserve passivity based on either assumptions on the structure of the original system, or special ways to pick the interpolation points). Devising robust implementations for Krylov methods and their use in model recution has received the attention of numerous works, among which [30], [31], [26], [36].

A related problem is that of matching moments at different points $s_1, \dots s_k$, rather than more moments at one point s_0. The solution to this problem is generally known as *rational Krylov* [4, Section 11.3], [81], and brings in three important additional questions: (a) what is the appropriate selection of points $s_1, \dots s_k$ to ensure passivity preservation (b) how can the associated projection matrices **W**, **V** be computed efficiently and (c) is there a suitable selection of $s_1, \dots s_k$ which ensures good approximation and if so, is there an efficient procedure to compute the corresponding interpolating projection? The answers are provided by the *spectral zero interpolation* approach, on which a brief overview follows. Antoulas [5] showed that if the expansion points are chosen among the roots of $\mathbf{H}(s) + \mathbf{H}^*(-s) = 0$ (the so called *spectral zeros*), then the reduced $\widetilde{\mathbf{\Sigma}}$ which matches one moment at each selected spectral zero is passive. At the same time, Sorensen [86] shows the analogy between spectral zeros and the eigenvalues of an associated *Hamiltonian pair*, demonstrating that the projection matrices **W**, **V** which interpolate at the spectral zeros can be computed from the eigenvectors of the Hamiltonian eigenvalue problem. Later, Ionuţiu et al. [44] show that interpolating the spectral zeros which are *dominant* in an appropriate residue norm ensures approximation quality and they propose to compute the corresponding **W**, **V** with a specialized iterative eigenvalue solver.

1.3.3 Exploiting the structure of electrical circuits

While the above literature overview applies to a general type of descriptor systems, some reduction methods exploit the special properties and structure of systems arising in circuit simulation. The most encountered representation which describes the dynamics of electrical circuits is obtained using *modified nodal analysis (MNA)* [37]. From Kirchhoff's current, voltage laws, and the branch constitutive equations, the MNA representation of an *RLC* circuit leads to the following descriptor system:

$$\underbrace{\begin{bmatrix} \mathcal{C} & \mathbf{0} \\ \mathbf{0} & \mathcal{L} \end{bmatrix}}_{\mathbf{E}} \underbrace{\frac{d}{dt}\begin{bmatrix} \mathbf{v}(t) \\ \mathbf{i}_L(t) \end{bmatrix}}_{\dot{\mathbf{x}}} + \underbrace{\begin{bmatrix} \mathcal{G} & \mathcal{E} \\ -\mathcal{E}^T & \mathbf{0} \end{bmatrix}}_{-\mathbf{A}} \underbrace{\begin{bmatrix} \mathbf{v}(t) \\ \mathbf{i}_L(t) \end{bmatrix}}_{\mathbf{x}} = \underbrace{\begin{bmatrix} \mathcal{B} \\ \mathbf{0} \end{bmatrix}}_{\mathbf{B}} \mathbf{u}(t), \tag{1.21}$$

where without loss of generality it is assumed that the inputs are current sources applied at the port nodes and that the outputs are voltages measured at the ports. An alternative representation with input voltages and output currents is treated in Chapter 2, from which the representation (1.21) can be obtained, as shown therein. The MNA system (1.21) has a special structure. The internal variables $\mathbf{x}$ are are split into node voltages, $\mathbf{v}$ and currents through inductors, $\mathbf{i}_L$. $\mathcal{G}$ is the conductance matrix, $\mathcal{C}$ is the capacitance matrix, $\mathcal{L}$ is the inductance matrix (a diagonal with the inductor values if there are no mutual inductances, otherwise the mutuals appear as off-diagonal entries). $\mathcal{E}$ is the incidence matrix which determines the topological connections for the inductors. $\mathcal{B}$ is the incidence matrix of current injections (the i^{th} column of $\mathcal{B}$ is the i^{th} unit vector for each input i). For demonstrative purposes, two simple *RLC* examples in MNA form are given in Sect. 2.5.1 and Sect. 5.3.1 respectively, and an *RC* example in Sect. 4.2.2. The blocks of the MNA matrices have important numerical properties, which are often exploited by reduction methods especially to preserve passivity: $\mathcal{G}$, $\mathcal{C}$, L are symmetric nonnegative definite (they have non-negative eigenvalues), and $\mathbf{C} = \mathbf{B}^T$ (the outputs are measured at the nodes to which the inputs are applied).

One of the most popular reduction methods for electrical circuits is PRIMA [71], which matches moments of the original system at $s_0 = 0$, and is based on a block-Arnoldi implementation. The most important advantages offered by PRIMA are: the applicability to MIMO systems and passivity preservation. With PRIMA, passivity is preserved by means of a congruence transformation (i.e. the left and right projection matrices are equal $\mathbf{W} = \mathbf{V}$) applied on the MNA matrices (1.21); the passivity proof is given in [71], and holds for all reduction methods based on congruence transforms applied to systems in the MNA form (1.21). One of the limitations of PRIMA is that the MNA structure of the original $\mathbf{\Sigma}$ is lost during the projection (2.1), and as a consequence finding a netlist representation for $\widetilde{\mathbf{\Sigma}}$ is not straightforward. Later, the structure preserving SPRIM [27] and the second order SAPOR [66, 89] variants have been introduced which, through structure preservation, allow a simpler realization of the reduced model as a circuit, while maintaining the desired moment matching and passivity preserving properties. In this thesis, Chapter 2 addresses the realization problem from different angles, including that of structure preservation via SPRIM. Chapter 6 brings an additional contribution in demonstrating how even models reduced with non-structure preserving methods such as PRIMA can be easily realized as netlists and re-used in simulations.

1.3.4 Special methods for multi-terminal systems

Despite the recent advances in model reduction, mostly aimed at improving the accuracy and efficiency of state-of-the-art methods, there remains a class of systems to which the applicability of existing approaches is limited. These are systems with a large number of inputs and outputs, in short, *multi-terminal systems*, which receive special attention in this thesis. Examples of large, multi-terminal systems are electrical networks resulting from parasitic extraction which can have terminal numbers exceeding thousands. The terminals can be either the user defined input-output nodes, or interconnec-

tion nodes between the linear, parasitic network and other non-linear circuit elements such as diodes or transistors. When reducing a multi-terminal system via a traditional approach, be it spectral- or Krylov-based, usually a reduced model results which is much denser than the original (even though its dimension is smaller). As explained and demonstrated in this thesis (see especially Chapter 4), re-using a dense reduced model in simulation requires the same, or even more computational effort than the original system, so the entire reduction effort becomes useless. Aside from the multi-terminal challenge, there is yet another limitation for traditional reduction approaches: forming the reducing projection is often too expensive (computational- and memory-wise) or even unfeasible for systems with a large number of internal variables (e.g. circuits with node numbers exceeding tens of thousands). It is thus vital to develop new methods which can efficiently handle such challenging systems arising in industrial problems.

Recently, the *ReduceR* [80] method has provided promising results in reducing very large, multi-terminal resistor networks. At the heart of ReduceR are tools from graph-theory and fill-reducing node reorderings, which ensure that only those variables are eliminated from the network, that do no generate a lot of fill-in. Using graph partitioning and a special hierarchical ordering of the system matrices new, efficient methods for reducing large, multi-terminal *RC* networks are developed in this thesis (Chapters 3 and 4), with extensions to *RLC* networks in Chapter 5. The foundation for the multi-terminal framework of these chapters is the congruence-based reduction methodology PACT [56, 57] which, compared to the PRIMA/SPRIM approach, exploits a different splitting of the system variables and matrices to construct related Krylov subspaces (the PACT methods are described in the afore-mentioned chapters).

Modeling multi-port systems from measured data

While all of the above methods construct a reduced model starting from an original descriptor system (1.3), a different problem arises when such a representation is not available. Rather, for a multi-port device, a reduced order model must be constructed *directly from frequency response measurements*. A recent solution to this problem is provided by the Loewner-based methodology [60–64] (an earlier approach is by Vector Fitting [20, 35]). Although the Loewner approach may not directly apply to networks with a very large number of terminals (due to the sparsity considerations above described), [60–64] show that it provides promising results for those with a moderate number of terminals. For such systems, Chapter 6 shows that low order macromodels obtained with the Loewner framework can also be synthesized and re-inserted in a circuit simulation flow.

1.4 Thesis outline

Throughout this thesis, model reduction and synthesis are closely linked. Therefore, the research is focused both on the approximation quality and efficiency of reduction, and

on whether/how the resulting reduced model can be cast into a circuit and reconnected with other sub-circuits, sources, or non-linear devices in the actual simulation phase. We call this the "use" (or "re-use") of the reduced model. Hence, almost all experiments in this thesis involve reduction, synthesis, and the simulation of the reduced circuit. Except for the self-created demonstrative examples, all circuits are provided by NXP Semiconductors. Unless otherwise stated, all circuit simulations are performed with Spectre [16].

This thesis is organized as a collection of articles, hence each chapter can be read individually. The material nevertheless builds up from chapter to chapter, and the appropriate connections among the different topics are made throughout the text. An outline of the thesis follows next.

With Chapter 1, the reader should become familiar with the problem background and with basic concepts related to system theory and model order reduction. The tie between reduction and synthesis is given in Chapter 2 which proposes a new framework for dealing with multi-terminal systems. The framework allows the decoupling of all sources or non-linear devices from the linear circuit which must be reduced, and their re-insertion after reduction and synthesis. Although this is intuitive from a physical perspective, an appropriate set-up for the original system equations is necessary, if this procedure is to succeed also numerically. The mathematical formulation which ensures this is proposed in Chapter 2. The reduction framework can be thought of as independent from the types of inputs/outputs chosen for the circuit. The synthesis problem is also analyzed and shows that unstamping (also denoted as RLCSYN [93]) is the most suitable in the derived multi-terminal framework. The material in this chapter has been published as [40, 41, 43] and has been reorganized here for presentation clarity. The proposed framework for multi-terminal reduction and synthesis gives the theoretical foundation for the approaches taken in later chapters.

The reduction of RC circuits is addressed in Chapters 3 and 4. Chapter 3 shows that the same projection which underlies the reduction of R-networks [80] extends immediately to RC networks and brings two main additional contributions. First, it proves that the governing projection for multi-terminal R/RC reduction ensures that resistors in the reduced network are positive. Then, a partition-based reduction for RC networks is derived, shown mathematically equivalent to the unpartitioned approach. This result guarantees that reduction accuracy remains satisfied also when the network is reduced by parts. The proposed partition-based scheme also gives a simple solution for computing path resistances between the terminals of a network, a problem which often occurs in circuit simulation [80]. The properties identified in Chapter 3 also hold for the more advanced method in Chapter 4.

In Chapter 4, the attention is turned to reducing very large RC networks with many terminals, for which a new method is developed, SparseRC. As the name suggests, the main feature of SparseRC is that it preserves sparsity during reduction. Retaining sparsity is crucial when reducing circuits with terminal numbers exceeding thousands, as otherwise the fill-in would render the reduced models useless during simulation.

SparseRC preserves sparsity with the help of graph partitioning and the identification of nodes responsible for fill-in. Another important feature of SparseRC is that it matches moments at *DC* for each subnet resulting from partitioning as well as for the recombined network. In addition, it can reduce efficiently very large networks for which existing techniques are inappropriate. Extensive experiments demonstrate that SparseRC achieves significant reduction rates and simulation speed-ups with little computation effort and at almost no accuracy loss. This chapter is published as [48]. Other publications related to this work are [42, 46, 49].

Chapter 5 addresses multi-terminal *RLC* reduction. Two approaches are compared, based on the first and second order formulation of system equations. A partition-based implementation is derived based on the second order form. An extensive comparison between the first and second order form is drawn, through which the advantages and limitations of each approach are identified. Among the most important contributions of Chapter 5 is a rank revealing decomposition of the reduced inductive susceptance matrix, which ensures that the synthesized model successfully simulates. Other problems, such as recovering circuit behavior at DC, are also analyzed.

Chapter 6 takes a new turn and resolves two important problems which usually limit the applicability of traditional reduction methods to multi-terminal circuits: (1) the reduction of ungrounded systems and (2) the synthesis without controlled sources. For un-grounded systems, the underlying matrix pencil $(\mathbf{A}, \mathbf{E})$ is singular. An example would be a sub-circuit which is extracted from a larger network. While such circuits can be reduced with the methods in Chapters 3, 4 and 5, other approaches such as PRIMA [71] would be immediately dismissed, as they generally assume that the pair $(\mathbf{A}, \mathbf{E})$ is regular. Chapter 6 proposes a terminal removal and recovery action, which allows un-grounded multi-terminal models to be reduced with standard methods as well. The second problem also finds a solution in Chapter 6 using transformations based on the input/output matrices of the reduced model.

As the key to preserving sparsity in multi-terminal MOR is graph partitioning, Chapter 7 gives a closing analysis on this topic. The standard partitioning objectives of state-of-the art software are revised and new objectives aimed at explicitly minimizing fill-in are derived. Although implementing these objectives exceeds the scope of this research, a new partitioning problem in the context of multi-terminal model reduction is proposed. Selections from this material appear in [47].

Chapter 2

Reduction and synthesis framework for multi-terminal circuits

A framework is presented for the reduction and synthesis of multi-terminal systems arising in circuit simulation. Two main problems are addressed: (1) setting up the appropriate circuit equations, so that reduction can be used without any constraints on the types of inputs applied to the circuit. This feature becomes especially useful when reducing sub-circuits of bigger systems individually, or the linear part of circuits containing non-linear devices, and (2) ensuring that reduced models recover their physical interpretation and that they can be re-inserted naturally in the original simulation flow via the relevant interconnection nodes.

2.1 Introduction

Although many model order reduction methods have been developed and have evolved since the 1990s (see for instance [4] for an overview), it is usually less clear how to use these methods efficiently in industrial practice, e.g., in a circuit simulator. One reason can be that the reduced order model does not satisfy certain physical properties, for instance, it may not be stable or passive while the original system is. Failing to preserve these properties is typically inherent to the reduced order method used (or its implementation). Passivity (and stability implicitly) can nowadays be preserved via several methods [5, 27, 44, 71, 74, 77, 86], but none addresses explicitly the practical aspect of (re)using the reduced order models with circuit simulation software (e.g., SPICE [92]).

Difficulties can occur at several levels, when inserting a reduced model in place of the original circuit in a simulation:

1. If the reduced model (2.1) was obtained based on certain restrictive assumptions about the types of inputs applied $\mathbf{u}$ and measured outputs $\mathbf{y}$, the simulation is usually also constrained to the same input/output types. Hence, if the model must be re-used in different simulation runs which would require other input and output types, a new reduced model would have to be generated for each scenario. A more elegant solution would be to have a single reduced circuit which can be re-coupled via its terminal nodes to *any* kind of driving inputs: voltage sources, current sources, non-linear devices, or other circuit blocks. In this chapter, the appropriate reduction setup is derived which ensures that reduced models can indeed accommodate any types of driving sources in simulations.

2. The linear circuit to be reduced is represented by a *netlist*, which is a description of the circuit element values (R,L,C) and their connections to the circuit nodes. Such an example is the simple netlist in Fig. 2.1. However, reduced order models are usually represented in terms of *system matrices* or of the *input-output transfer function*. Typically, the default input format for circuit simulators are netlists with nodes and circuit elements, and would require additional software to directly handle mathematical representations. Thus, a reduced model in *netlist* representation is more easily coupled to other circuit blocks, simulated, or used by circuit designers for further analysis. This chapter also addresses *synthesis* methods which convert reduced models into netlists. Here, two methods are presented: *Foster synthesis* for single input single output systems (SISO), and *synthesis by un-stamping* for multi input multi output (MIMO) systems. The latter especially suits the multi-terminal reduction setup mentioned at item [1.]. With the proposed multi-terminal reduction and synthesis framework reduced circuits are obtained which contain no controlled sources or transformers, and which are easily re-inserted in any SPICE-like circuit simulation environment.

The material is organized as follows. Sect. 2.2 formulates the two main problems of this chapter. The first is treated in Sect. 2.3, and provides the setup for multi-terminal model reduction. The second problem, namely synthesis, is addressed in Sect. 2.4, which describes two procedures: RLCSYN unstamping, and Foster synthesis. Experiments in Sect. 2.5 demonstrate the functionality of the proposed reduction and synthesis framework. Sect. 2.6 concludes.

2.2 Problem formulation

Following Sect. 1: given an original system $\Sigma(\mathbf{E}, \mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D})$ of dimension n in the form (1.3), a reduced model $\widetilde{\Sigma}(\widetilde{\mathbf{E}}, \widetilde{\mathbf{A}}, \widetilde{\mathbf{B}}, \widetilde{\mathbf{C}}, \mathbf{D})$ of dimension $k \ll n$ is sought. In particular:

$$\widetilde{\Sigma}(\widetilde{\mathbf{E}}, \widetilde{\mathbf{A}}, \widetilde{\mathbf{B}}, \widetilde{\mathbf{C}}, \mathbf{D}) \equiv (\mathbf{W}^T\mathbf{E}\mathbf{V}, \mathbf{W}^T\mathbf{A}\mathbf{V}, \mathbf{W}^T\mathbf{B}, \mathbf{C}\mathbf{V}, \mathbf{D}), \tag{2.1}$$

where $\mathbf{V}, \mathbf{W} \in \mathbb{R}^{n \times k}$ are matrices whose $k \ll n$ columns form bases for relevant subspaces of the space in which the original internal variables lie. Rather than addressing explicitly the construction of $\mathbf{V}$, $\mathbf{W}$ here, the aim of this chapter is to derive a reduction and synthesis framework which overcomes the two *global* problems identified in Sect. 2.1. Chapters 3, 4 and 5 are based on this framework and are concerned explicitly with forming the projection matrices $\mathbf{V}$, $\mathbf{W}$ appropriate to the context therein.

For clarity, the two problems identified in Sect. 2.1 are re-stated:

1. What is the appropriate modeling of equations (1.3) governing the original system $\Sigma(\mathbf{E}, \mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D})$, which ensures that the reduced $\widetilde{\Sigma}(\widetilde{\mathbf{E}}, \widetilde{\mathbf{A}}, \widetilde{\mathbf{B}}, \widetilde{\mathbf{C}}, \mathbf{D})$ accommodates any input/output types in later simulation stages?
2. How can $\widetilde{\Sigma}(\widetilde{\mathbf{E}}, \widetilde{\mathbf{A}}, \widetilde{\mathbf{B}}, \widetilde{\mathbf{C}}, \mathbf{D})$ recover its physical interpretation? For instance, if the original $\Sigma(\mathbf{E}, \mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D})$ describes a circuit with R, L, C elements, which methods are suitable to convert $\widetilde{\Sigma}(\widetilde{\mathbf{E}}, \widetilde{\mathbf{A}}, \widetilde{\mathbf{B}}, \widetilde{\mathbf{C}}, \mathbf{D})$ into a circuit representation with R, L, C elements only, without introducing new elements such as controlled sources or transformers?

Sect. 2.3 provides the setup for multi-terminal model reduction which underlies all reduction methods in this thesis. In this chapter in particular, the reduction itself is performed with SPRIM/IOPOR [27, 93] but for demonstration purposes only. Then, Sect. 2.4 discusses two synthesis methods, from which the *unstamping* method is chosen as the most suitable for many-terminal systems. The proposed framework offers flexibility in choosing the input/output types of the original or reduced system during the simulation stage, demonstrates that synthesis is possible with R, L, C elements only, and that the reduced circuits can be re-inserted easily via its terminals nodes to the desired simulation environment.

Further on, the following naming conventions are used. $\Sigma(\mathbf{E}, \mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D})$ will denote a system which "hides" any further information about the structure of the underlying matrices. It will be clear from the context when the special structure of system matrices is exploited, such as in (1.21). Also, if the inputs $\mathbf{u}$ are currents and the outputs $\mathbf{y}$ are voltages, then $\mathbf{H}(s)$ is an impedance function (each i, j entry of $\mathbf{H}(s)$, $\mathbf{H}_{(i,j)}(s) = \frac{\mathbf{Y}_i(s)}{\mathbf{U}_j(s)}$, represents a voltage deviled by a current). We denote such a current driven circuit to be in *impedance form*. If on the other hand the inputs are voltages and the outputs are currents, then $\mathbf{H}(s)$ is an admittance function, and we denote the circuit to be in *admittance form*. We will also often refer to reduction methods that are *input-output structure preserving*. These ensure that the reduced input-output matrices $\widetilde{\mathbf{B}}$, $\widetilde{\mathbf{C}}$ retain the special structure of the original input-output matrices $\mathbf{B}$, $\mathbf{C}$ respectively (simply put, $\widetilde{\mathbf{B}}$, $\widetilde{\mathbf{C}}$ are sub-blocks of $\mathbf{B}$, $\mathbf{C}$). Preserving input-output structure during reduction is an important feature which later enables synthesis without controlled sources or transformers. Among the input-output structure preserving methods are SPRIM/IOPOR [27, 93], PACT [56] for RC networks, the RLC version [57] and the methods developed in Chapters 3, 4 and 5. Chapter 6 presents another reduction approach which enables synthesis

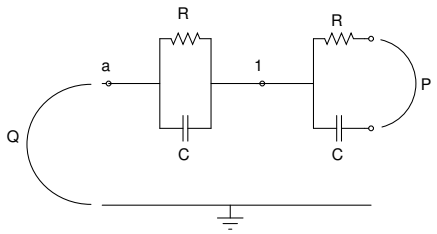


Figure 2.1: *Circuit with terminal a, internal node 1, port P, and port Q(a,0).*

without controlled sources even for methods that are not input-output structure preserving by default. The approach taken in Chapter 6 is nevertheless also based on the reduction setup presented next.

2.3 Reduction setup for multi-terminal systems

The terms *internal nodes*, *terminals (or external nodes)*, and *ports* are often found in electronic engineering related papers. In short, external nodes are those related to the system inputs **u** and outputs **y**, while all the rest are internal nodes. Thus, a *terminal (external node)* is a node that is visible on the outside of a circuit, i.e., a node in which currents can be injected (cf. node a in Fig. 2.1). An *internal node* is one which is not visible on the outside of a circuit, i.e., no currents can be injected in an internal node (cf. node 1 in Figure 2.1). A port consists of two terminals that can be connected, for instance, by a source, a non-linear circuit element, or another (sub)circuit (cf. port P in Fig. 2.1). Sometimes terminals are referred to as ports and vice versa: from the context it should then be clear which terminal(s) complete the ports; usually it is implicitly assumed that the ground node completes the ports. In Fig. 2.1, for instance, terminal "a" can be seen as a port (Q) by including the ground node.

Most of the multi-terminal systems in this thesis arise from full device-parasitic simulations. The linear part i.e. the *parasitic network* is first decoupled from other non-linear elements such as transistors or diodes, and then reduced. Through the de-coupling, the interconnection points between the linear subcircuit and non-linear elements become *terminals* of the linear part to be reduced. A similar scenario occurs for instance when reducing parts of a network individually, for instance, after *partitioning* a large circuit, as is the case in Chapters 3, 4 and 5. The reduction setup proposed in this chapter ensures that the removal of non-linear elements or other circuit blocks before reduction and their re-insertion after reduction has a theoretical foundation. Without loss of generality, in the following section this is exemplified through the removal/re-insertion of voltage sources from a circuit. The result holds for the other afore-mentioned scenarios as well. By reducing a current driven model, the reduced netlist can be easily coupled to other circuitry in place of the original netlist, and (re)using the reduced model in simulation becomes straightforward.

In the following section, it is shown that the *impedance form* (defined in Sect. 2.2) is the one which gives the desired freedom in connecting the circuit in practice to other types of elements than current sources. First, we motivate reduction in impedance form, and show how it also applies for systems that are originally in admittance form. The impedance-based reduction setup is demonstrated via the input-output structure preserving method SPRIM/IOPOR [27,93]. Finally, a note on numerical aspects concerning the SPRIM/IOPOR projection is given.

2.3.1 A simple admittance to impedance conversion

The strength of input-output structure preserving methods is that the input/output connectivity can be synthesized after reduction without controlled sources, provided that the system is in *impedance form*: the inputs are currents injected into the circuit terminals, and the outputs are voltages measured at the terminals. The emerging question is: how to ensure synthesis without controlled sources, if the original model is in admittance form (i.e., it is voltage driven)? We show that reduction and synthesis via the input impedance transfer function can be performed, also when the original circuit is initially voltage driven. The same principle of an impedance-based reduction serves as the basis for reducing the linear part of circuits with non-linear elements, as is done in Chapters 3, 4, 5 and 6 or other methods such as [80, 88]. The admittance to impedance conversion proposed herein (also published as [43]) enables the reduction, synthesis and most importantly re-use of complex linear system which contain couplings to other circuit blocks (e.g. non-linear devices, sources, etc.).

Consider the modified nodal analysis (MNA) description of an input *admittance*[1] type *RLC* circuit, driven by n_s voltage sources:

$$\underbrace{\begin{bmatrix} \mathcal{C} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathcal{L} \end{bmatrix}}_{\mathbf{E}_Y} \underbrace{\frac{d}{dt}\begin{bmatrix} \mathbf{v}(t) \\ \mathbf{i}_S(t) \\ \mathbf{i}_L(t) \end{bmatrix}}_{\dot{\mathbf{x}}_Y} + \underbrace{\begin{bmatrix} \mathcal{G} & \mathcal{E}_{\mathbf{v}} & \mathcal{E}_{\mathbf{L}} \\ -\mathcal{E}_{\mathbf{v}}{}^T & \mathbf{0} & \mathbf{0} \\ -\mathcal{E}_{\mathbf{L}}{}^T & \mathbf{0} & \mathbf{0} \end{bmatrix}}_{-\mathbf{A}_Y} \underbrace{\begin{bmatrix} \mathbf{v}(t) \\ \mathbf{i}_S(t) \\ \mathbf{i}_L(t) \end{bmatrix}}_{\mathbf{x}_Y} = \underbrace{\begin{bmatrix} \mathbf{0} \\ \mathcal{B} \\ \mathbf{0} \end{bmatrix}}_{\mathbf{B}_Y} \mathbf{u}(t), \quad (2.2)$$

where $\mathbf{u}(t) \in \mathbb{R}^{n_s}$ are input voltages and $\mathbf{y}(t) = \mathbf{C}_Y \mathbf{x}(t) \in \mathbb{R}^{n_s}$ are output currents, $\mathbf{C}_Y = \mathbf{B}_Y^T$. The states are $\mathbf{x}_Y(t)^T = [\mathbf{v}(t),\ \mathbf{i}_S(t),\ \mathbf{i}_L(t)]^T$, with $\mathbf{v}(t) \in \mathbb{R}^{n_v}$ the node voltages, $\mathbf{i}_S(t) \in \mathbb{R}^{n_s}$ the currents through the voltage sources, and $\mathbf{i}_L(t) \in \mathbb{R}^{n_L}$ the currents through the inductors, $n_v + n_s + n_L = n$. The $n_v = n_1 + n_2$ node voltages are formed by the n_1 external nodes/terminals[2] and the n_2 internal nodes (assuming that the voltage sources may be ungrounded, n_1 satisfies: $n_s < n_1 \leq 2n_s + 1$). The dimensions of the underlying matrices are: $\mathcal{C} \in \mathbb{R}^{n_v \times n_v}$, $\mathcal{G} \in \mathbb{R}^{n_v \times n_v}$, $\mathcal{E}_v \in \mathbb{R}^{n_v \times n_s}$, $\mathcal{L} \in$

[1] The subscript Y refers to quantities associated with a system in admittance form.

[2] The MNA form (2.2) corresponds to the ungrounded circuit (i.e., the reference node is counted within the n_1 external nodes), resulting in a defective matrix pencil $(\mathbf{A}_Y, \mathbf{E}_Y)$. For subsequent computations such as the construction of a Krylov subspace, the pencil $(\mathbf{A}_Y, \mathbf{E}_Y)$ must be regular. Thus in (2.2), one node must be chosen as a ground (reference) node by removing the row/column corresponding to that node; this ensures that the regularity conditions (i) and (ii) from [76, Assumption 4] are satisfied. The positive definiteness of $\mathcal{C}, \mathcal{L}, \mathcal{G}$ is also a necessary condition to ensure the circuit's passivity.

$\mathbb{R}^{n_L \times n_L}$, $\mathcal{E}_L \in \mathbb{R}^{n_v \times n_L}$, $\mathcal{B} \in \mathbb{R}^{n_1 \times n_s}$. Assuming without loss of generality that (2.2) is permuted such that the *first* n_1 nodes are the external nodes, the input voltages are determined by a linear combination of $\mathbf{v}_{1:n_1}(t)$ only. Thus the following holds:

$$\mathcal{E}_{\mathbf{v}} = \begin{bmatrix} \mathcal{B}_v \\ \mathbf{0}_{n_2 \times n_s} \end{bmatrix} \in \mathbb{R}^{n_v \times n_s}, \ \mathcal{B}_v \in \mathbb{R}^{n_1 \times n_s}, \ \mathcal{B} = -\mathcal{B}_v. \tag{2.3}$$

We derive the first order impedance-type system associated with (2.2). Note that by definition, $\mathbf{i}_S(t)$ flows **out of** the circuit terminals into the voltage source (i.e., from the $+$ to the $-$ terminal of the voltage source, see also the example in Sect. 2.5.1). We can define new input currents as the currents flowing **into** the circuit terminals: $\mathbf{i}_{in}(t) = -\mathbf{i}_S(t)$. Since $\mathbf{v}_{1:n_1}(t)$ are the terminal voltages, they describe the new output equations, and it is straightforward to rewrite (2.2) in the impedance form:

$$\begin{cases} \underbrace{\begin{bmatrix} \mathcal{C} & \mathbf{0} \\ \mathbf{0} & \mathcal{L} \end{bmatrix}}_{\mathbf{E}} \underbrace{\frac{d}{dt}\begin{bmatrix} \mathbf{v}(t) \\ \mathbf{i}_L(t) \end{bmatrix}}_{\dot{\mathbf{x}}} + \underbrace{\begin{bmatrix} \mathcal{G} & \mathcal{E}_{\mathbf{L}} \\ -\mathcal{E}_{\mathbf{L}}{}^T & \mathbf{0} \end{bmatrix}}_{-\mathbf{A}} \underbrace{\begin{bmatrix} \mathbf{v}(t) \\ \mathbf{i}_L(t) \end{bmatrix}}_{\mathbf{x}} = \underbrace{\begin{bmatrix} \mathcal{E}_v \\ \mathbf{0} \end{bmatrix}}_{\mathbf{B}} \mathbf{i}_{in}(t) \\ \underbrace{\begin{bmatrix} \mathcal{E}_v^T & \mathbf{0} \end{bmatrix}}_{\mathbf{C}} \underbrace{\begin{bmatrix} \mathbf{v}(t) \\ \mathbf{i}_L(t) \end{bmatrix}}_{\mathbf{x}} = \mathbf{y}(t) = \mathcal{B}_v \mathbf{v}_{1:n_1}(t), \quad \mathcal{E}_v^T = \begin{bmatrix} \mathcal{B}_v^T & \mathbf{0}_{n_s \times n_2} \end{bmatrix} \end{cases} \tag{2.4}$$

where $\mathbf{B}$ describes the new *input incidence matrix* corresponding the input currents, $\mathbf{i}_{in}$. The new *output incidence matrix* is $\mathbf{C}$, corresponding to the voltage drops over the circuit terminals. We emphasize that (2.4) has fewer unknowns than (2.2), since the currents $\mathbf{i}_S$ have been eliminated. The transfer function associated to (2.4) is an input impedance: $\mathbf{H}_{(i,j)}(s) = \frac{\mathbf{Y}_i(s)}{\mathbf{I}_{in_j}(s)}$, where $\mathbf{Y}(s)$ and $\mathbf{I}_{in}(s)$ are the Laplace transforms of $\mathbf{y}(t)$ and $\mathbf{i}_{in}(t)$ respectively. In Sect. 2.3.2 we explain how to obtain an impedance type reduced order model in the input/output structure preserved form:

$$\begin{cases} \underbrace{\begin{bmatrix} \widetilde{\mathcal{C}} & \mathbf{0} \\ \mathbf{0} & \widetilde{\mathcal{L}} \end{bmatrix}}_{\widetilde{\mathbf{E}}} \underbrace{\frac{d}{dt}\begin{bmatrix} \widetilde{\mathbf{v}}(t) \\ \widetilde{\mathbf{i}}_L(t) \end{bmatrix}}_{\dot{\widetilde{\mathbf{x}}}} + \underbrace{\begin{bmatrix} \widetilde{\mathcal{G}} & \widetilde{\mathcal{E}}_L \\ -\widetilde{\mathcal{E}}_L^T & \mathbf{0} \end{bmatrix}}_{-\widetilde{\mathbf{A}}} \underbrace{\begin{bmatrix} \widetilde{\mathbf{v}}(t) \\ \widetilde{\mathbf{i}}_L(t) \end{bmatrix}}_{\widetilde{\mathbf{x}}} = \underbrace{\begin{bmatrix} \widetilde{\mathcal{E}}_v \\ \mathbf{0} \end{bmatrix}}_{\widetilde{\mathbf{B}}} \mathbf{i}_{in}(t) \\ \underbrace{\begin{bmatrix} \widetilde{\mathcal{E}}_v^T & \mathbf{0} \end{bmatrix}}_{\widetilde{\mathbf{C}}} \underbrace{\begin{bmatrix} \widetilde{\mathbf{v}}(t) \\ \widetilde{\mathbf{i}}_L(t) \end{bmatrix}}_{\widetilde{\mathbf{x}}} = \mathbf{y}(t) = \mathcal{B}_v \mathbf{v}_{1:n_1}(t), \ \widetilde{\mathcal{E}}_v^T = \begin{bmatrix} \mathcal{B}_v^T & \mathbf{0}_{n_s \times k_2} \end{bmatrix} \end{cases} \tag{2.5}$$

where $\widetilde{\mathcal{C}}$, $\widetilde{\mathcal{L}}$, $\widetilde{\mathcal{G}}$, $\widetilde{\mathcal{E}}_v$ are the reduced MNA matrices, and the reduced input impedance transfer function is: $\widetilde{\mathbf{H}}(s) = \widetilde{\mathbf{C}}(s\widetilde{\mathbf{E}} - \widetilde{\mathbf{A}})^{-1}\widetilde{\mathbf{B}}$. Due to the input/output preservation, the circuit terminals are easily preserved in the reduced model (2.5).

It turns out that after reduction and synthesis, the reduced model (2.5) can still be used as a voltage driven admittance block in simulation. This is shown next. We can rewrite the second equation in (2.5) as: $\begin{bmatrix} -\widetilde{\mathcal{E}}_v^T & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \widetilde{\mathbf{v}}(t)^T & \widetilde{\mathbf{i}}_S(t)^T & \widetilde{\mathbf{i}}_L(t)^T \end{bmatrix}^T = \mathcal{B}\mathbf{u}(t)$. This

result together with $\mathbf{i}_{in}(t) = -\mathbf{i}_S(t)$, reveals that (2.5) can be rewritten as:

$$\underbrace{\begin{bmatrix} \widetilde{\mathcal{C}} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \widetilde{\mathcal{L}} \end{bmatrix}}_{\widetilde{\mathbf{E}}_Y} \underbrace{\frac{d}{dt}\begin{bmatrix} \widetilde{\mathbf{v}}(t) \\ \mathbf{i}_S(t) \\ \widetilde{\mathbf{i}}_L(t) \end{bmatrix}}_{\dot{\widetilde{\mathbf{x}}}_Y} + \underbrace{\begin{bmatrix} \widetilde{\mathcal{G}} & \widetilde{\mathcal{E}}_v & \widetilde{\mathcal{E}}_L \\ -\widetilde{\mathcal{E}}_v^T & \mathbf{0} & \mathbf{0} \\ -\widetilde{\mathcal{E}}_L^T & \mathbf{0} & \mathbf{0} \end{bmatrix}}_{-\widetilde{\mathbf{A}}_Y} \underbrace{\begin{bmatrix} \widetilde{\mathbf{v}}(t) \\ \mathbf{i}_S(t) \\ \widetilde{\mathbf{i}}_L(t) \end{bmatrix}}_{\widetilde{\mathbf{x}}_Y} = \underbrace{\begin{bmatrix} \mathbf{0} \\ \mathcal{B} \\ \mathbf{0} \end{bmatrix}}_{\widetilde{\mathbf{B}}_Y} \mathbf{u}(t), \quad (2.6)$$

which has the same structure as the original admittance model (2.2). Conceptually one could have reduced system (2.2) directly via the input admittance. In that case, synthesis would have required controlled sources [36], due to the fact that the structure of the input/output matrix would not be preserved. As shown above, this is avoided by: applying the simple admittance-to-impedance conversion (2.2) to (2.4), reducing (2.4) to (2.5), and finally reinserting voltage sources as in (2.6). In other words, the input-output structure preserved reduced admittance (2.6) tells that, after synthesizing the reduced impedance model (2.5) into a reduced netlist with all terminal nodes preserved, the voltage source elements are safely reconnected at the terminals.

2.3.2 Input-output structure preservation with SPRIM/IOPOR

A reduced model of the impedance form (2.5) can be obtained for instance via the input-output structure preserving SPRIM/IOPOR projection [27, 93] as follows. Let $\mathbf{V}$ be the projection matrix obtained with PRIMA [71], split according to the block dimensions of (2.4):

$$\mathbf{V} = [\mathbf{V}_1, \mathbf{V}_2, \mathbf{V}_3] \in \mathbb{R}^{((n_1+n_2+n_L)\times k)},$$

where $n_1 + n_2 = n_v$, $\mathbf{V}_1 \in \mathbb{R}^{(n_1 \times k)}$, $\mathbf{V}_2 \in \mathbb{R}^{(n_2 \times k)}$, $\mathbf{V}_3 \in \mathbb{R}^{(n_L \times k)}$, $k \geq n_1$, $i = 1 \ldots 3$. After appropriate orthonormalization (e.g., via Modified Gram-Schmidt [78, Chapter 1]), we obtain: $\widetilde{\mathbf{V}}_i = \text{orth}(\mathbf{V}_i) \in \mathbb{R}^{n_i \times k_i}$, $k_i \leq k$. The SPRIM [27] block structure preserving projection is:

$$\widetilde{\mathbf{V}} = \begin{bmatrix} \widetilde{\mathbf{V}}_1 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \widetilde{\mathbf{V}}_2 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \widetilde{\mathbf{V}}_3 \end{bmatrix} \in \mathbb{R}^{n \times (k_1+k_2+k_3)}$$

which preserves the block structure of (2.4) but does not yet preserve the structure of the input and output matrices.

The input-output structure preserving projection is obtained via SPRIM/IOPOR as proposed in [93]:

$$\overline{\mathbf{V}} = \left[\begin{array}{cc|c} \mathbf{I}_{n_1 \times n_1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \widetilde{\mathbf{V}}_2 & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} & \widetilde{\mathbf{V}}_3 \end{array}\right] \in \mathbb{R}^{n \times (n_1+k_2+k_3)},$$

where the top block is:

$$\mathbf{W} = \begin{bmatrix} \mathbf{I}_{n_1 \times n_1} & \mathbf{0} \\ \mathbf{0} & \widetilde{\mathbf{V}}_2 \end{bmatrix} \in \mathbb{R}^{(n_1+n_2)\times(n_1+k_2)}. \tag{2.7}$$

Recalling (2.3), we obtain the reduced system matrices in (2.5):

$$\widetilde{\mathcal{C}} = \mathbf{W}^T \mathcal{C} \mathbf{W},\ \widetilde{\mathcal{G}} = \mathbf{W}^T \mathcal{G} \mathbf{W},\ \widetilde{\mathcal{L}} = \widetilde{\mathbf{V}}_3^T \mathcal{L} \widetilde{\mathbf{V}}_3,\ \widetilde{\mathcal{E}}_L = \mathbf{W}^T \mathcal{E}_L \widetilde{\mathbf{V}}_3 \tag{2.8}$$

$$\widetilde{\mathcal{E}}_v = \mathbf{W}^T \mathcal{E}_{\mathbf{v}} = \begin{bmatrix} \mathcal{B}_v^T & \mathbf{0}_{n_1 \times k_2} \end{bmatrix}^T, \tag{2.9}$$

which compared to (2.3) clearly preserves input-output structure. Therefore a netlist representation for the reduced impedance-type model can be obtained, that is driven by injected currents just as the original circuit. To this end, we use the Laplace transform and convert (2.5) to the second order form:

$$\begin{cases} [s\widetilde{\mathcal{C}} + \widetilde{\mathcal{G}} + \frac{1}{s}\widetilde{\Gamma}]\widetilde{\mathbf{v}}(s) = \widetilde{\mathcal{E}}_v \mathbf{i}_{in}(s) \\ \widetilde{\mathbf{y}}(s) = \widetilde{\mathcal{E}}_v^T \widetilde{\mathbf{v}}(s), \end{cases} \tag{2.10}$$

where $\widetilde{\mathbf{i}}_L(s) = \frac{1}{s}\widetilde{\mathcal{L}}^{-1}\left(\widetilde{\mathcal{E}_{\mathbf{L}}}^T\right)\widetilde{\mathbf{v}}(s)$ are the eliminated current variables and $\widetilde{\Gamma} = \widetilde{\mathcal{E}}_L \widetilde{\mathcal{L}}^{-1} \widetilde{\mathcal{E}}_L^T$. The reduced model (2.10) is synthesized via RLCSYN unstamping according to [93] (this is revised in Sect. 2.4.1).

On SPRIM/IOPOR and rank loss of $\widetilde{\mathbf{A}}$

In some cases it was observed (and shown by results in Sect. 2.5.3) that models reduced with SPRIM or SPRIM/IOPOR exhibit poles and zeros at 0. This section explains when this happens and supports theoretically the interpretation of the results in Sect. 2.5.3.

Proposition 2.3.1 *Let* $\mathbf{W}$ *and* $\widetilde{\mathbf{V}}_3$ *be the SPRIM/IOPOR projection matrices (2.7), with full column rank. If* $\widetilde{\mathcal{E}}_L = \mathbf{W}^T \mathcal{E}_L \widetilde{\mathbf{V}}_3$ *in (2.8) has deficient column rank, then the SPRIM/IOPOR reduced model (2.5) has at least a pole-zero pair at* 0*.*

Proof 2.3.1 *Recalling that* $\widetilde{\mathbf{V}}_2$ *has full column rank, it is clear from (2.7) that* $\mathbf{W}$ *also has full column rank. Nonetheless* $\mathbf{W}^T \in \mathbb{R}^{(n_1+k_2)\times(n_1+n_2)}$ *has more columns than rows (usually* $k_2 \ll n_2$*), thus* $\mathbf{W}^T$ *has deficient column rank. Hence* $\widetilde{\mathcal{E}}_L = \mathbf{W}^T \mathcal{E}_L \widetilde{\mathbf{V}}_3 \in \mathbb{R}^{(n_1+k_2)\times k_3}$ *may also lose column rank (even if* $\mathcal{E}_L$ *and* $\widetilde{\mathbf{V}}_3$ *have full column rank). If* $\widetilde{\mathcal{E}}_L$ *has deficient column rank, then in (2.5) we have: rank(*$\widetilde{\mathbf{A}}$*) < #cols(*$\widetilde{\mathbf{A}}$*). Thus* $\widetilde{\mathbf{A}}$ *has deficient column rank* $\Rightarrow$ $\widetilde{\mathbf{A}}$ *has at least an eigenvalue at* $0 \Rightarrow 0$ *is also an eigenvalue of the pencil (*$\widetilde{\mathbf{A}}, \widetilde{\mathbf{E}}$*) and this is a pole at* 0*. The zeros of the reduced system (2.5) are determined as eigenvalues of the extended matrix pair* $(\widetilde{\mathbf{A}}_z, \widetilde{\mathbf{E}}_z)$*, where* $\widetilde{\mathbf{A}}_z$ *is the Rosenbrock matrix [78, Chapter 5]:* $\widetilde{\mathbf{A}}_z = \begin{bmatrix} \widetilde{\mathbf{A}} & \widetilde{\mathbf{B}} \\ -\widetilde{\mathbf{C}} & \mathbf{0} \end{bmatrix} = \begin{bmatrix} -\widetilde{\mathcal{G}} & -\widetilde{\mathcal{E}}_L & \widetilde{\mathcal{E}}_v \\ \widetilde{\mathcal{E}}_L^T & \mathbf{0} & \mathbf{0} \\ -\widetilde{\mathcal{E}}_v^T & \mathbf{0} & \mathbf{0} \end{bmatrix}$,

and $\widetilde{\mathbf{E}}_z = \begin{bmatrix} \widetilde{\mathbf{E}} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}$. *From the structure of* $\widetilde{\mathbf{A}}_z$ *it is seen that if* $\widetilde{\mathcal{E}}_L$ *has deficient column rank, then* $\widetilde{\mathbf{A}}_z$ *also loses rank and will have a* 0 *eigenvalue. Consequently,* $(\widetilde{\mathbf{A}}_z, \widetilde{\mathbf{E}}_z)$ *will also have an eigenvalue at* 0*, which means that the SPRIM/IOPOR reduced system (2.5) has a zero at* 0. ■

Analytically, pole-zero pairs at 0 are harmless since they theoretically cancel. Numerically this may not be the case, altering the approximation for low frequencies (as seen for instance in the result in Sect. 2.5.3). As also reported in Chapter 5, the deficiency in the column rank of $\widetilde{\mathcal{E}}_L$ [and implicitly of $(\widetilde{\mathbf{A}})$] may prejudice the computation of the DC solution when the reduced, synthesized *RLC* model is re-simulated. A rank revealing procedure for $\widetilde{\Gamma}$ which avoids these numerical limitations is proposed in Chapter 5.

2.4 Synthesis

Synthesis is the realization step needed to map the reduced order model from the mathematical representation (in terms of system matrices or transfer function) into a netlist consisting of electrical circuit components [34]. In [15] it was shown that passive systems (with positive real transfer functions) can be synthesized with positive *R*,*L*,*C* elements and transformers (see also [76]). Later developments [14] propose a method to circumvent the introduction of transformers, however the resulting realization is non-minimal (i.e., the number of electrical components generated during synthesis is too large). Allowing for possibly negative R, L, C values, other methods have been proposed via e.g. direct stamping [57, 71] or full realization [36, 73]. These mostly model the input/output connections of the reduced model with controlled sources. The term "synthesis" is usually tied to circuit realizations which contain only positive circuit elements, but which may include transformers or controlled sources. The main reason for requiring that circuit elements in the reduced model are positive, is to guarantee passivity. This is however not a necessary requirement. A reduced circuit with negative elements can also be passive as long as its underlying transfer function is positive real. This holds for all reduced models obtained in this thesis. For this reason, we dispose of the positiveness condition on circuit elements when using the term "synthesis". Furthermore, in this work synthesis methods are considered which do not introduce transformers or controlled sources. Two such synthesis methods are: (1) *RLCSYN synthesis by unstamping* [93] and (2) *Foster synthesis* [34], discussed next.

2.4.1 Synthesis by unstamping: RLCSYN

This section focuses on synthesis via the RLCSYN [93] method, which is based on *unstamping* the reduced matrix data directly into the netlist representation. It is suitable for synthesizing models which were reduced via methods that preserve the MNA structure

and the input-output connectivity at the circuit terminals. All reduction methods in this thesis satisfy these properties, hence RLCSYN unstamping is the synthesis method of choice in the remaining chapters.

The presentation of RLCSYN follows [93, Sect. 4], [40] and is briefly described here. In circuit simulation, the process of forming the $\mathcal{C}, \mathcal{G}, \mathcal{L}$ system matrices from the individual branch element values is called "stamping". The reverse operation of "unstamping" involves decomposing entry-wise the values of the reduced system matrices in (2.10) into the corresponding R, L, and C values. The resulting Rs, Ls and Cs are connected in the reduced netlist according to the MNA topology. The reduced input/output matrices of (2.10) directly reveal the input connections in the reduced model via injected currents, without any controlling elements. The unstamping procedure is best understood via an example, which is provided in Sect. 2.5.1. The general framework reduction followed by RLCSYN unstamping is as follows:

1. The system to be reduced is in MNA impedance form (2.4). If the system is of admittance type (2.2), apply the admittance-to-impedance conversion from Sect. 2.3.1. If the circuit contains non-linear elements, they are removed in a similar manner so that the linear part in impedance form (2.4) remains.

2. System (2.4) is reduced with an input-output structure preserving method and converted to second order form (2.10). The alternative is to obtain the second order form of the original system first, and reduce it directly in second order form [8,93].

3. According to [93], to obtain a reduced RLC netlist which successfully simulates, the reduced $\widetilde{\Gamma}$ from (2.10) must be diagonalized and regularized as proposed therein. Diagonalization ensures that all inductors in the synthesized model are connected to ground (i.e., there are no inductor loops). Regularization eliminates spurious over-large inductors. These steps however are not needed for purely RC circuits.

4. Originally [93] impose that in (2.4), no Ls are directly connected to the input terminals so that, after reduction, diagonalization and regularization preserve the input/output structure. In Chapter 5 it is shown that this restrictive assumption is unnecessary, since the rank revealing decomposition of $\widetilde{\Gamma}$ can be performed, without affecting the input-output structure, irrespective of whether there are Ls connected to terminals or not. In fact, the reduced model can be cast into the form (2.5) and synthesized as an RC equivalent netlist; this procedure allows multi-terminal circuits with Ls connected to ports to be reduced, synthesized and re-simulated.

5. Finally, the reduced netlist is inserted via its terminal nodes in the original simulation setup. Voltage sources or non-linear elements that were removed from the network are simply reconnected at the terminal nodes.

With unstamping, roughly speaking every non-zero entry in the upper triangle of the reduced matrices maps to a circuit element in the reduced netlist. Hence, the main drawback of unstamping would be that, when the reduced system matrices are dense

and the number of terminals is large [p is of $O(10^3)$], the RLCSYN procedure yields dense netlists. For a dense reduced network with p terminals and k internal nodes, the RLCSYN synthesized netlist will have $O[(p+k)^2]$ circuit elements. This situation is remedied however when the reduction itself ensures that the reduced matrices are sparse also when p is large. Then, the $O[(p+k)^2]$ factor becomes only a loose upper bound on the actual number of elements resulting from unstamping, and in practice much fewer elements are generated. Hence, RLCSYN becomes especially suitable for synthesizing multi-terminal reduced models obtained by sparsity preserving reduction methods. Devising such methods is a very challenging task, a solution for which is the SparseRC method, developed in Chapter 4. Other sparsity-related contributions are given in Chapters 5 and 7.

2.4.2 Foster synthesis of rational transfer functions

This section describes the Foster synthesis method, which was developed in the 1930s by Foster and Cauer [34] and involves realization based on the system's *transfer function* rather than on the systems matrices. Hence, the Foster approach can be used to realize any reduced order model that is computed by standard projection based model order reduction techniques (where "standard" means that no structure preserving requirements are necessary).

Realization of impedance functions

Realizations in this section are described in terms of SISO impedances (Z-parameters): the circuit input is a current injection, and the output is a voltage. Given the reduced system (2.1) and assuming that the matrix pencil $(\widetilde{\mathbf{A}}, \widetilde{\mathbf{E}})$ is non-defective[3], consider the partial fraction expansion [50] of its transfer function:

$$\widetilde{\mathbf{H}}(s) = \sum_{i=1}^{k} \frac{\tilde{r}_i}{s - \tilde{p}_i} + \mathbf{D} + s r_j, \tag{2.11}$$

The residues are $\tilde{r}_i = \frac{(\widetilde{\mathbf{C}}\tilde{\mathbf{x}}_i)(\tilde{\mathbf{y}}_i^* \widetilde{\mathbf{B}})}{\tilde{\mathbf{y}}_i^* \widetilde{\mathbf{E}} \tilde{\mathbf{x}}_i}$, the poles are $\tilde{p}_i$ and, if non-zero, $\mathbf{D}$ and r_j give additional contributions from poles at ∞. An eigentriplet $(\tilde{p}_i, \tilde{\mathbf{x}}_i, \tilde{\mathbf{y}}_i)$ is composed of an eigenvalue $\tilde{p}_i$ of $(\widetilde{\mathbf{A}}, \widetilde{\mathbf{E}})$ and the corresponding right and left eigenvectors $\tilde{\mathbf{x}}_i, \tilde{\mathbf{y}}_i \in \mathbb{C}^k$. The expansion (2.11) consists of basic summands of the form:

$$Z(s) = r_1 + \frac{r_2}{s - p_2} + \frac{r_3}{s} + \left(\frac{r_4}{s - p_4} + \frac{\bar{r}_4}{s - \bar{p}_4} \right) + s r_6 + \left(\frac{r_7}{s - p_7} + \frac{r_7}{s - \bar{p}_7} \right), \tag{2.12}$$

where for completeness we can assume that any kind of poles may appear, i.e., either purely real, purely imaginary, in complex conjugate pairs, at ∞ or at 0 (see also Table 2.1). The Foster realization converts each term in (2.12) into the corresponding circuit

[3] For every eigenvalue of the pencil $(\widetilde{\mathbf{A}}, \widetilde{\mathbf{E}})$, its algebraic multiplicity is equal to its geometric multiplicity.

block with R, L, C components, and connects these blocks in series in the final netlist. This is shown in Fig. 2.2, where each summand from (2.12) is represented in order as a circuit block between two nodes. Note that any reordering of the circuit blocks in the realization of (2.12) in Fig. 2.2 still is a realization of (2.12). The values for the circuit components in Fig. 2.2 are determined according to Table 2.1 (for a full derivation see [40]).

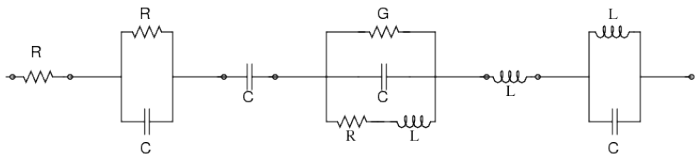


Figure 2.2: *Realization of a general impedance transfer function as a series RLC circuit.*

Table 2.1: *Circuit element values for Fig. 2.2 for the Foster impedance realization of (2.12)*

pole	residue	R(Ohm)	C(F)	L(H)	G(Ohm^{-1})
$p_1 = \infty$	$r_1 \in \mathbb{R}$	r_1			
$p_2 \in \mathbb{R}$	$r_2 \in \mathbb{R}$	$-\frac{r_2}{p_2}$	$\frac{1}{r_2}$		
$p_3 = 0$	$r_3 \in \mathbb{R}$		$\frac{1}{r_3}$		
$p_4 = \sigma + i\omega \in \mathbb{C}$ $p_5 \equiv \bar{p}_4$	$r_4 = \alpha + i\beta \in \mathbb{C}$ $r_5 \equiv \bar{r}_4$	$\frac{a_0}{a_1} L$	$\frac{1}{a_1}$	$\frac{a_1^3}{a_1^2 b_0 - a_0(a_1 b_1 - a_0)}$	$\frac{a_1 b_1 - a_0}{a_1^2}$
$a_0 = -2(\alpha\sigma + \beta\omega), \quad a_1 = 2\alpha, \quad b_0 = \sigma^2 + \omega^2, \quad b_1 = -2\sigma$					
$p_6 = \infty$	$r_6 \in \mathbb{R}$			r_6	
$p_7 \in i\mathbb{R}$ $p_8 \equiv \bar{p}_7$	$r_7 \in \mathbb{R}$ $r_8 \equiv \bar{r}_7$		$\frac{1}{r_7}$	$\frac{2r_7}{p_7 \bar{p}_7}$	

Realization of admittance functions

Realizations in this section will be described according to [65] in terms of SISO admittances (Y-parameters): the inputs are voltage sources and the outputs are currents. Consider the admittance function:

$$Y(s) = r_1 + \frac{r_2}{s - p_2} + \left(\frac{r_3}{s - p_3} + \frac{\bar{r}_3}{s - \bar{p}_3} \right) + s r_5,$$

where $r_1, r_2, p_1, r_6 \in \mathbb{R}$ and $r_3 = \nu + i\mu, p_3 = \alpha + i\beta \in \mathbb{C}$. This admittance function can be realized by the *RLCG* network in Fig. 2.3, with the elements defined by:

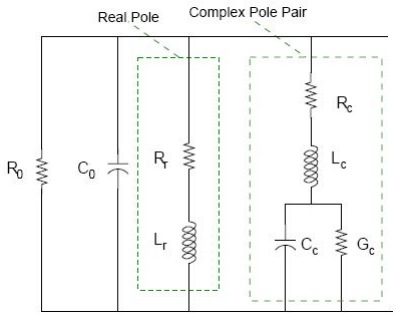


Figure 2.3: *Realization of a general admittance transfer function as a RLCG parallel circuit [65].*

pole	residue	R	C	L	G
$p_1 = \infty$	$r_1 \in \mathbb{R}$	$\frac{1}{r_1}$			
$p_2 \in \mathbb{R}$	$r_2 \in \mathbb{R}$	$-\frac{p_2}{r_2}$		$\frac{1}{r_2}$	
$p_3 \in \mathbb{C}$ $p_4 \equiv \bar{p}_3$	$r_3 \in \mathbb{C}$ $r_4 \equiv \bar{r}_3$	$2L(L(\nu\alpha + \mu\beta) - \alpha)$	$\frac{1}{L(\alpha^2+\beta^2 2R(\nu\alpha+\mu\beta))}$	$\frac{1}{2\nu}$	$-2LC(\nu\alpha + \mu\beta)$
$p_5 = 0$	$r_5 \in \mathbb{R}$		r_5		

Multi-port transfer functions

If the transfer function is an input admittance [$\mathbf{H}(s) = \mathbf{Y}(s)$], then a MIMO Foster realization is determined as described in [90]. To the author's best knowledge, MIMO Foster realizations for general impedance transfer functions $\mathbf{Z}(s)$ are only known to be feasible for 2-terminal systems. Even in that case, an admittance Foster network is determined first [90], which is converted to an impedance network using a $\Pi - Y$ transformation. Some developments for multiport realizations of *RC* impedance transfer functions can be found in [87].

The realization in netlist form can be implemented in any language such as SPICE [92], so that it can be reused and combined with other circuits as well. The advantages of Foster synthesis are: (1) its straightforward implementation for single-input-single-output (SISO) transfer functions, via either the impedance or the admittance transfer function, (2) after reducing purely *RC* or *RL* circuits via modal approximation [78], the reduced netlists obtained from Foster synthesis are guaranteed to have positive *RC* or *RL* values respectively (see [40] for a proof). Note however that Foster synthesis does not guarantee positive circuit elements in general (e.g., when used to synthesize reduced models originating from *RLC* circuits, or reduced models of *RC* and *RL* circuits that were obtained with methods different than modal approximation). The main disadvantage is that for systems with many terminals, the MIMO Foster realization (see for instance [90]) would give dense reduced netlists, since the transfer function between every pair of ex-

ternal nodes has to be realized. So, for a k-dimensional reduced system with p terminals, the Foster synthesis will yield a total of p^2k circuit elements. In contrast, synthesis by unstamping is more straightforward to use for MIMO systems and, when applied after structure and sparsity preserving reductions, can yield reduced netlists with much fewer circuit elements (as seen in Chapter 4).

To summarize the synthesis section, a comparison between Foster and unstamping is drawn in Table 2.2.

Table 2.2: *Summary and comparison of synthesis methods*

To summarize	**Unstructured projection and Foster realization of $\hat{\mathbf{H}}(s)$**	**I/O preserving reduction and RLCSYN unstamping**
Properties	Passivity guaranteed by positive realness of $\widehat{\mathbf{H}}(s)$ from appropriate $\mathbf{V}$, $\mathbf{W}$	Passivity ensured via congruence transformation $\widetilde{\mathbf{W}}$, $\widetilde{\mathbf{W}}^T\widetilde{\mathbf{W}} = \mathbf{I}$ on Σ in MNA form
Advantages	Σ need not be in MNA form RC, RL reduced with modal approximation $\Rightarrow$ positive elements	Preserved I/O and MNA structure Unstamping easy for MIMO via impedance $\widehat{\Sigma}$ Benefits from sparsity preserving reduction $\Rightarrow$ sparse MIMO netlists
Main hurdles	MIMO realization for > 2 terminals only for **admittance** $\mathbf{H}(s)$ and $\widehat{\mathbf{H}}(s)$ Dense MIMO netlists: for p ports, k poles $O(p^2k)$ circuit elements Positive R, L, C not guaranteed	$\widetilde{\mathcal{G}}$ may loose rank with SPRIM/IOPOR When reduced $\widetilde{\mathcal{G}}$, $\widetilde{\mathcal{C}}$, $\widetilde{\Gamma}$ are dense, for p terminals, k internal nodes $\Rightarrow O((p+k)^2)$ circuit elements Positive R, L, C not guaranteed

2.5 Numerical examples

Three circuits are chosen to demonstrate the applicability of the reduction and synthesis framework presented in this chapter. The first is a simple circuit which illustrates the complete admittance-to-impedance formulation of Sect. 2.3.1 and the RLCSYN unstamping procedure described in Sect. 2.4.1. The second example is a SISO transmission line model, while the third is a MIMO model of a spiral inductor. For the single-input-single-output (SISO) examples, one can easily provide synthesized models via

both Foster and RLCSYN. For the multi-input-multi-output (MIMO) example, a synthesized model can be obtained straightforwardly with RLCSYN, thus RLCSYN synthesis is preferred over Foster synthesis.

2.5.1 Illustrative example

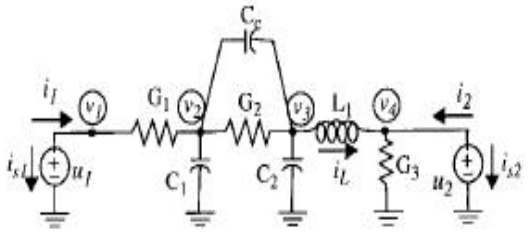


Figure 2.4: *The admittance-type circuit driven by input voltages from [71].* $G_{1,2,3} = 0.1S$, $L_1 = 10^{-3}H$, $C_{1,2} = 10^{-6}$, $C_c = 10^{-4}$, $\|u_{1,2}\| = 1$.

The circuit in Fig. 2.4 is a simple *RLC* netlist driven by two voltage sources, u_1 and u_2. The terminals are v_1 and v_4, and the current flowing out of the voltage sources into the terminals are i_1 and i_2 respectively. The MNA equations in admittance form (2.2) are:

$$\left[\begin{array}{cccc|cc|c} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & C_1+C_c & -C_c & 0 & 0 & 0 \\ 0 & 0 & -C_c & C_2+C_c & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & L \end{array}\right] \left[\begin{array}{c} \dot{v}_1 \\ \dot{v}_4 \\ \dot{v}_2 \\ \dot{v}_3 \\ \hline \dot{i}_{S_1} \\ \dot{i}_{S_2} \\ \hline \dot{i}_L \end{array}\right] + \left[\begin{array}{cccc|cc|c} G_1 & 0 & -G_1 & 0 & 1 & 0 & 0 \\ 0 & G_3 & 0 & 0 & 0 & 1 & -1 \\ -G_1 & 0 & G_1+G_2 & -G_2 & 0 & 0 & 0 \\ 0 & 0 & -G_2 & G_2 & 0 & 0 & 1 \\ \hline -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 1 & 0 & -1 & 0 & 0 & 0 \end{array}\right] \left[\begin{array}{c} v_1 \\ v_4 \\ v_2 \\ v_3 \\ \hline i_{S_1} \\ i_{S_2} \\ \hline i_L \end{array}\right] = \left[\begin{array}{cc} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ \hline -1 & 0 \\ 0 & -1 \\ \hline 0 & 0 \end{array}\right] \left[\begin{array}{c} u_1 \\ u_2 \end{array}\right] \quad (2.13)$$

Notice that:

$$\mathbf{i}_{in} = \left[\begin{array}{c} i_1 \\ i_2 \end{array}\right] = -\left[\begin{array}{c} i_{S_1} \\ i_{S_2} \end{array}\right] \quad (2.14)$$

$$\mathbf{u} = \left[\begin{array}{c} u_1 \\ u_2 \end{array}\right] = \left[\begin{array}{c} v_1 \\ v_4 \end{array}\right], \quad (2.15)$$

thus the external nodes (input nodes/terminals) are v_1 and v_4, and the internal nodes are v_2 and v_3. As described in Sect. 2.3.1, (2.13) has an equivalent impedance formula-

tion (2.4), with:

$$\mathcal{C} = \begin{bmatrix} 0\,0 & 0 & 0 \\ 0\,0 & 0 & 0 \\ 0\,0 & C_1+C_c & -C_c \\ 0\,0 & -C_c & C_2+C_c \end{bmatrix}, \quad \mathcal{G} = \begin{bmatrix} G_1 & 0 & -G_1 & 0 \\ 0 & G_3 & 0 & 0 \\ -G_1 & 0 & G_1+G_2 & -G_2 \\ 0 & 0 & -G_2 & G_2 \end{bmatrix} \tag{2.16}$$

$$\mathcal{L} = \begin{bmatrix} L \end{bmatrix}, \ \mathcal{E}_L = \begin{bmatrix} 0 \\ -1 \\ 0 \\ 1 \end{bmatrix}, \ \mathcal{E}_v = \begin{bmatrix} 1\,0 \\ 0\,1 \\ 0\,0 \\ 0\,0 \end{bmatrix}, \quad \mathcal{B} = \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix}, \ \mathcal{B}_v = -\mathcal{B} \tag{2.17}$$

Matrices (2.16), (2.17) are reduced in first order form using SPRIM/IOPOR according to Sect. 2.3.2, then synthesized by unstamping via RLCSYN. Note that there is an L directly connected to the second input node v_4, thus assumption 4, Sect. 2.3.2 from RLCSYN is not satisfied. For simplicity, we thus reduce and synthesize the single-input-single-output version of (2.13) only, where the second input i_2 is removed. Therefore the new incidence matrices are:

$$\mathcal{E}_{v_1} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \mathcal{B}_1 = \begin{pmatrix} -1 \end{pmatrix}, \ \mathcal{B}_{v_1} = -\mathcal{B}_1. \tag{2.18}$$

We choose an underlying PRIMA projection matrix $\mathbf{V} \in \mathbb{C}^{n \times k}$ spanning a $k = 2$-dimensional Krylov subspace (with expansion point $s_0 = 0$). According to Sect. 2.3.2, after splitting $\mathbf{V}$ and appropriate re-orthonormalization, the dimensions of the input-output structure preserving partitioning are :

$$n_1 = 1, \ n_2 = 3, \ n_L = 1, \ k_2 = 2, \ k_3 = 1, \tag{2.19}$$

and the SPRIM/IOPOR projection is:

$$\widetilde{\mathbf{W}} = \left[\begin{array}{ccc|c} 1 & 0 & 0 & 0 \\ 0 & 4.082 \cdot 10^{-1} & -4.861 \cdot 10^{-1} & 0 \\ 0 & 8.164 \cdot 10^{-1} & 5.729 \cdot 10^{-1} & 0 \\ 0 & 4.082 \cdot 10^{-1} & -6.597 \cdot 10^{-1} & 0 \\ \hline 0 & 0 & 0 & 1 \end{array}\right] \in \mathbb{C}^{5 \times 4}, \text{ with } \mathbf{W} \in \mathbb{C}^{4 \times 3}. \tag{2.20}$$

After diagonalization and regularization, the SPRIM/IOPOR reduced system matrices

in (2.10) are:

$$\widetilde{\mathcal{G}} = \begin{bmatrix} 1 & 8.165\cdot 10^{-2} & -5.729\cdot 10^{-2} \\ 8.165\cdot 10^{-2} & 9.999\cdot 10^{-2} & -7.726\cdot 10^{-2} \\ -5.7295\cdot 10^{-2} & -7.7265\cdot 10^{-2} & 2.084\cdot 10^{-1} \end{bmatrix}, \quad \widetilde{\mathcal{E}}_{v_1} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

$$\widetilde{\mathcal{C}} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1.749\cdot 10^{-5} & -5.052\cdot 10^{-5} \\ 0 & -5.052\cdot 10^{-5} & 1.527\cdot 10^{-4} \end{bmatrix}, \quad \widetilde{\Gamma} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 30.14 \end{bmatrix}. \tag{2.21}$$

Reduced matrices (2.21) are now unstamped individually using RLCSYN. The reduced system dimension in second order form is thus $N = 3$, indicating that the reduced netlist will have three nodes and an additional ground node. In the following, we denote by $M_{i,j}$ $i = 1 \ldots N$, $j = 0 \ldots N-1$ a circuit element connected between nodes (i, j) in the resulting netlist. M represents a circuit element of the type: R,L,C or current source J.

By unstamping $\widetilde{\mathcal{G}}$, we obtain the following R values :

$$R_{1,0} = \left[\sum_{k=1}^{3} \widetilde{\mathcal{G}}_{(1,k)}\right]^{-1} = 8.0417\ \Omega, \quad R_{1,2} = -\left[\widetilde{\mathcal{G}}_{(1,2)}\right]^{-1} = -12.247\ \Omega, \quad R_{1,3} = -\left[\widetilde{\mathcal{G}}_{(1,3)}\right]^{-1} = 17.452\ \Omega,$$

$$R_{2,0} = \left[\sum_{k=1}^{3} \widetilde{\mathcal{G}}_{(2,k)}\right]^{-1} = 9.5798\ \Omega, \quad R_{2,3} = -\left[\widetilde{\mathcal{G}}_{(2,3)}\right]^{-1} = 12.942\ \Omega, \quad R_{3,0} = \left[\sum_{k=1}^{3} \widetilde{\mathcal{G}}_{(3,k)}\right]^{-1} = 13.535\ \Omega.$$

By unstamping $\widetilde{\mathcal{C}}$, we obtain the following C values:

$$\begin{aligned} C_{2,0} &= \sum_{k=1}^{3} \widetilde{\mathcal{C}}_{(2,k)} = -3.3026\cdot 10^{-5}\ F, \quad C_{2,3} = -\widetilde{\mathcal{C}}_{(2,3)} = 5.0526\cdot 10^{-5},\ F, \\ C_{3,0} &= \left[\sum_{k=1}^{3} \widetilde{\mathcal{C}}_{(3,k)}\right]^{-1} = 1.0221\cdot 10^{-4}\ F. \end{aligned}$$

By unstamping $\widetilde{\Gamma}$, we obtain the following L values:

$$L_{3,0} = \left[\sum_{k=1}^{3} \widetilde{\Gamma}_{(3,k)}\right]^{-1} = 3.317\cdot 10^{-2}\ H.$$

By unstamping $\widetilde{\mathcal{E}}_{v_1}$, we obtain the current source $J_{1,0}$ of amplitude 1 A.

The equivalent netlist written in circuit simulation language (SPICE [92]) is shown in Fig. 2.5

Table 2.3 summarizes the reduction and synthesis results. Even though the number of internal variables (states) generated by the simulator is smaller for the SPRIM/IOPOR model than for the original, the number of circuit elements generated by RLCSYN is larger in the reduced model than in the original (note that this is a demonstrative exam-

```
.subckt small_rlc 1 0
     r_1_0 1 0 8.0417250765565598e+000
     r_1_2 1 2 -1.2247448713915894e+001
     r_1_3 1 3 1.7452546181796258e+001
     r_2_0 2 0 9.5798755840972589e+000
     r_2_3 2 3 1.2942609947762115e+001
     r_3_0 3 0 1.3535652691596653e+001
     l_3_0 3 0 3.3170000000000033e-002
     c_2_0 2 0 -3.3026513336014821e-005
     c_2_3 2 3 5.0526513336014765e-005
     c_3_0 3 0 1.0221180442099465e-004
     * Connect source between terminals 1 and 0
     * Resistors 6
     * Capacitors 3
     * Inductors 1
.ends small_rlc
```

Figure 2.5: *Netlist description obtained from RLCSYN unstamping for the reduced model (2.21).*

ple only hence no real candidate for reduction). Chapters 4, 5 and 6 address explicitly the problem of minimizing the fill generated inside the reduced matrices. Fig. 2.6 shows that approximation with SPRIM/IOPOR is more accurate than with PRIMA. The circuit re-simulation of the RLCSYN synthesized model also matches the MATLAB simulation of the reduced transfer function.

Table 2.3: *Input impedance reduction (SPRIM/IOPOR) and synthesis (RLCSYN)*

System	Dimension	R	C	L	States	Inputs/Outputs
Original	5	3	3	1	5	1
SPRIM/IOPOR	4	6	3	1	4	1

2.5.2 SISO *RLC* network

We reduce the SISO *RLC* transmission line in Fig. 2.7. Note that the circuit is driven by the voltage $\mathbf{u}$, thus it is of admittance type (2.2). In [44], reduction via the admittance transfer function was shown using various methods, in particular with the *dominant spectral zero method (Dominant SZM)*. Here we also reduce the circuit via its impedance formulation (2.4) as described in Sect. 2.3.1, by removing the voltage source and driving the circuit via a current flowing into its terminal. After reduction and synthesis via the input impedance, we show that the reduced admittance is recovered as well in simulation. This is done easily by driving the reduced synthesized impedance model via a voltage at the input terminal.

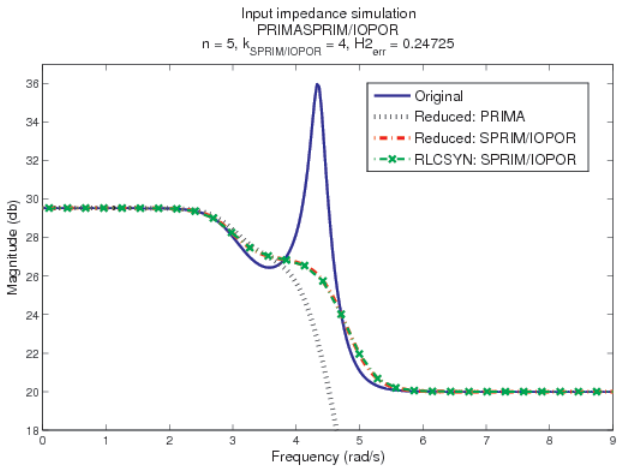


Figure 2.6: *Original, reduced and synthesized systems: PRIMA, SPRIM/IOPOR. The MATLAB reduced (red) and SPICE synthesized (green) models overlap, as expected.*

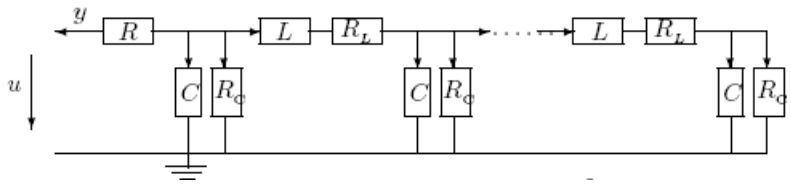


Figure 2.7: *Transmission line for example of Sect. 2.5.2.*

Fig. 2.8 shows the input impedance of the original impedance model (2.4) and reduced with Dominant SZM. The model is passive and stable [44] and is synthesized using the impedance Foster realization in Sect. 2.4.2 (the circuit simulation of the synthesized model is superimposed with the MATLAB simulation). For comparison, the direct admittance-based Dominant SZM reduction [44], synthesized with the Foster admittance approach (Sect. 2.4.2), is shown in Fig. 2.10. Both the reduced impedance (Fig. 2.8) and admittance (Fig. 2.10) capture the behavior of the original model well for the entire frequency range, and can also reproduce oscillations at dominant frequency points.

The SPRIM/IOPOR reduced impedance model is shown in Fig. 2.9, together with the response obtained in the re-simulation of the synthesized model obtained with RLCSYN. Note that if the original circuit had been reduced directly from the admittance form (2.2), synthesis by unstamping via RLCSYN would have required controlled sources to model the input of the reduced network [36].

The benefit of the admittance-to-impedance transformation described in Sect. 2.3.1 is seen in Fig. 2.11. By reducing the system in impedance form with SPRIM/IOPOR and synthesizing (2.5) [via RLCSYN unstamping of the second order form (2.10)], we are

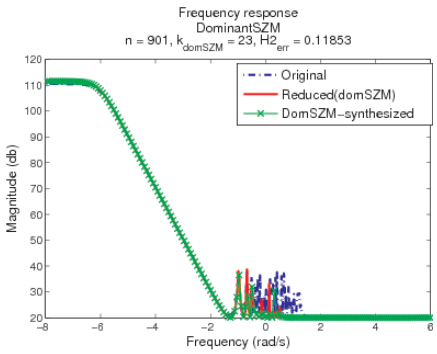


Figure 2.8: *Input impedance transfer function: original, reduced with Dominant SZM (red) and synthesized via Foster impedance (green).*

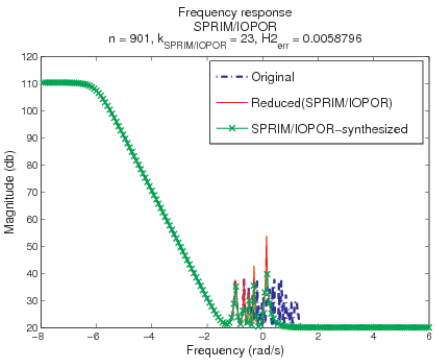


Figure 2.9: *Input impedance transfer function: original, reduced with SPRIM/IOPOR (red) and synthesized with RLCSYN (green).*

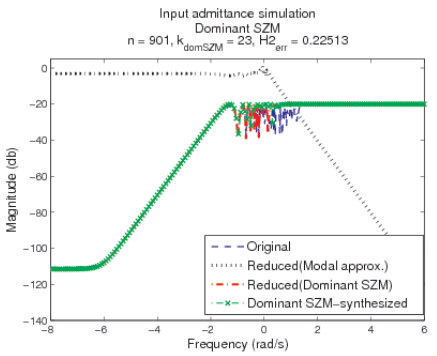


Figure 2.10: *Input admittance transfer function: original, reduced with Dominant SZM in admittance form (red) and synthesized with Foster admittance (green).*

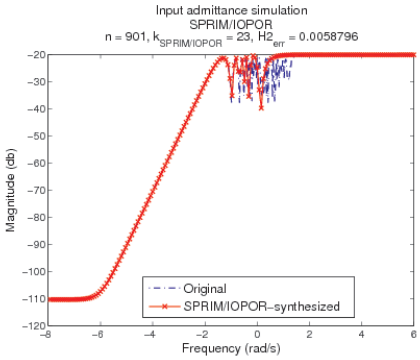


Figure 2.11: *Input admittance transfer function: original admittance response is compared to the admittance response of the reduced, RLCSYN synthesized SPRIM/IOPOR model.*

able to recover the reduced admittance (2.6) as well. This result is shown in Fig. 2.11: the original admittance transfer function (2.2) is compared to the circuit re-simulation of the synthesized model, after reinserting the voltage source as an input according to (2.6). The approximation is good for the entire frequency range, except close to 0 where the oscillating behavior is over-estimated in the reduced model. Table 2.4 summarizes the reduction and synthesis. With both methods, the dimension (number of nodes in the netlist), the number of internal variables (states), as well as the number of circuit elements were reduced.

Table 2.4: *Impedance reduction and synthesis for transmission line in Fig. 2.5.2*

System	Dimension	R	C	L	States	Simulation time
Original	902	500	300	300	901	1.5 s
Dominant SZM	23	22	11	10	34	0.02 s
SPRIM/IOPOR	23	78	66	6	18	0.02 s

2.5.3 MIMO RLC network

We reduce the MIMO RLC netlist resulting from the parasitic extraction of a coil structure [33]. The model has 4 pins (external nodes). Pin 4 is connected to other circuit nodes only via C's, which causes the original model (2.4) to have a pole at 0. The example shows that: (1) the SPRIM/IOPOR model preserves the terminals and is synthesizable with RLCSYN without controlled sources, and (2) structure preserving projections may perturb the location of poles and zeros 0, affecting the approximation quality at low frequencies (a technique which remedies this effect is proposed in Chapter 5).

Fig. 2.12 shows the simulation of the transfer function from input 4 to output 4, which clearly reflects the presence of the pole at 0 due to the large response magnitude for low frequencies. By inspecting the response from input 3 to output 3 however, we notice in Fig. 2.13 that the SPRIM/IOPOR model is less accurate around DC than PRIMA. The SPRIM/IOPOR (2.7) projection approximates the original pole at 0 by a small pole (not at 0), but still places zeros at 0 due to the dependencies created in the Rosenbrock matrix $\widetilde{\mathbf{A}}_z$ (see Sect. 2.3.2). Such pole-zero pairs can no longer cancel numerically, affecting the approximation quality for low frequencies.

The alternative is to ground pin 4 prior to reduction. This will remove the pole at 0 from the original model, resolve the corresponding numerical cancellation effects, and improve approximation around DC. As seen from Fig. 2.15, SPRIM/IOPOR applied on the remaining 3-terminal system gives a better approximation than PRIMA for the entire frequency range. The transient simulation in Fig. 2.14 confirms that the SPRIM/IOPOR model is both accurate and stable. With pin 4 grounded however, we loose the ability to (re)connect the synthesized model in simulation via all the terminals. This scheme was adopted here only to demonstrate possible numerical sensitivities of certain reduction projections to the presence/absence of system poles at 0. In Chapters 4, 5 robust reduction methods are presented which overcome such numerical limitations when reducing large, multi-terminal circuits, while preserving the connectivity at all terminal nodes.

2.6 Concluding remarks

A framework was presented for the reduction and synthesis of multi-terminal systems arising in circuit simulation. An admittance to impedance conversion was proposed as a pre-model reduction step and shown to enable synthesis without controlled sources.

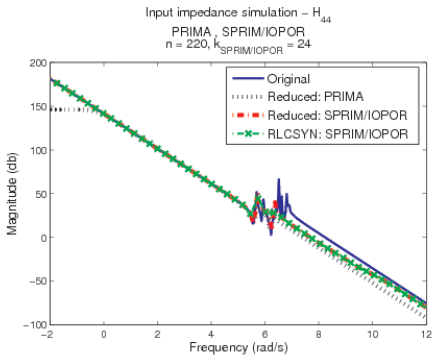


Figure 2.12: *Input impedance transfer function with "v_4" kept:* $\mathbf{H}_{44}$ *for PRIMA, SPRIM/IOPOR and RLCSYN realization.*

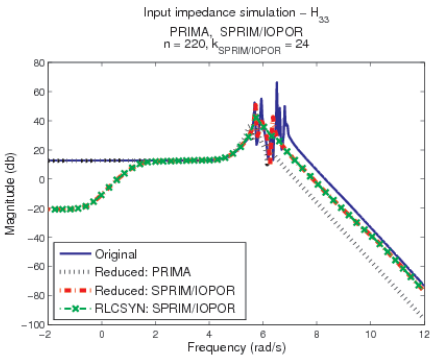


Figure 2.13: *Input impedance transfer function with "v_4" kept:* $\mathbf{H}_{33}$ *for PRIMA, SPRIM/IOPOR and RLCSYN realization.*

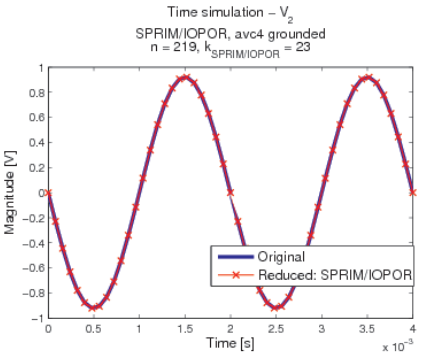


Figure 2.14: *Transient simulation with "v_4" grounded: voltage measured at node* 2 *for SPRIM/IOPOR (from RLCSYN realization).*

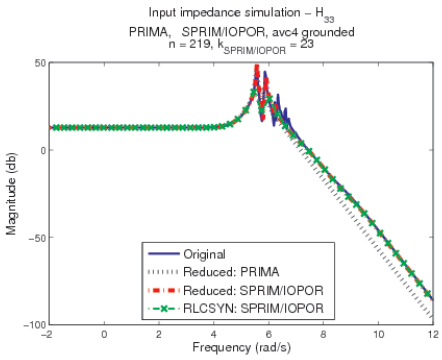


Figure 2.15: *Input impedance transfer function with "v_4" grounded:* $\mathbf{H}_{33}$ *for PRIMA, SPRIM/IOPOR and RLCSYN realization.*

This simple yet elegant approach gives the theoretical foundation for detaching voltage sources or non-linear elements before the reduction phase and reinserting them easily afterwards. Two synthesis approaches were described: RLCSYN [93] synthesis by unstamping (for MIMO systems) and Foster realization (for SISO transfer functions). The reduction and synthesis framework was tested on several examples of moderate size. The framework forms the basis for the multi-terminal reduction methods of Chapters 4, 3, 5 and 6 which are focused on more challenging industrial problems.

Chapter 3

Reduction of multi-terminal R/RC networks

A multi-terminal reduction framework for R/RC circuits is presented, based on two related methodologies, ReduceR [80] and PACT [56]. The underlying reducing projection is shown here to preserve the positiveness of resistors, the path resistance between circuit terminals, and also to support a partition-based implementation. This is an important result which guarantees that, when circuits are reduced by parts, the same desirable properties are retained (moment-matching, passivity, terminal connectivity) which are normally ensured by an unpartitioned approach. Several realistic multi-terminal parasitic networks are reduced in this framework, demonstrating approximation quality and significant computational speed-ups in re-simulations.

3.1 Introduction

With the decrease in feature sizes and the increasing complexity of VLSI circuit designs, parasitic effects have to be analyzed as to properly understand and control the relative impact between different components on a chip. The simulation of parasitic R and RC networks is however often a difficult task, as these networks may contain millions of nodes among which thousands are connection nodes (*terminals*) to nonlinear devices such as diodes or transistors. Reducing such *multi-terminal* networks to circuits with fewer nodes and elements can help to perform the desired simulations at much lower computational cost.

In this chapter, the focus is on a multi-terminal reduction framework based on [80] for for R and [56] for RC networks, denoted here as *ReduceR* and *PACT*, respectively. These

methods turn out to be governed by the same reducing projection, whose important advantages are: passivity preservation, moment matching at DC, and an immediate, natural, preservation of the connectivity at the terminal nodes (what in Chapter 2 is referred to as input/output structure preservation). ReduceR achieves tremendous reduction rates for very large resistor networks with thousands of terminals, but the question of whether the resulting reduced resistor networks have only positive resistors remains open in [80]. Here, it is shown that the resistors in the reduced network are indeed positive (a different proof was derived independently in [91]). This result extends immediately for the reduction of multi-terminal *RC* networks which are governed by the same underlying projection as ReduceR, for instance that underlying PACT, although this was not shown in the original reference [56]. Turning the attention to the PACT-based reduction of multi-terminal *RC* networks, this chapter proposes a partition-based implementation of PACT which has the potential to reduce more efficiently very large *RC* networks, for which forming a reducing projection directly on the entire network is too costly or unfeasible. It can also improve the overall sparsity of the reduced network, by applying fill reducing node reorderings per subnet to identify fill-creating nodes. More advanced partitioning and reordering strategies, especially aimed at improving sparsity, are given in Chapter 4. The partition-based *RC* reduction in addition reveals a simple and efficient solution to the problem of computing path resistances between terminals.

This chapter is organized as follows. In Sect. 3.1.1, the general setup for R/RC is described. Sect. 3.2 gives the proof on the positiveness of resistors. Sect. 3.3 derives the partition-based reduction of *RC* circuits, shows its mathematical equivalence to the PACT projection, provides the solution for the computation of path resistances, and describes the actions for further improving sparsity. Numerical results are provided in Sect. 3.4 and Sect. 3.5 concludes.

3.1.1 Reduction setup for R/RC networks

The model reduction setup for *R* and *RC* networks in this chapter follows [80] and [56] respectively, and is presented here in preparation for the additional derivations of this chapter.

Consider the modified nodal analysis (MNA) [37] description of an *RC* circuit:

$$(\mathbf{G} + s\mathbf{C})\mathbf{x}(s) = \mathbf{B}\mathbf{u}(s), \tag{3.1}$$

where MNA matrices $\mathbf{G}$, $\mathbf{C}$ are symmetric, non-negative definite, corresponding to the stamps of resistor and capacitor values respectively. $\mathbf{x} \in \mathbb{R}^{m+p}$ denote the node voltages (measured at the m internal nodes and the p terminals) and $m + p$ is the dimension of (3.1). $\mathbf{u} \in \mathbb{R}^p$ are the currents injected into the terminals. The outputs are the voltage drops at the terminal nodes: $\mathbf{y}(s) = \mathbf{B}^T\mathbf{x}(s)$. The underlying matrix dimensions are: $\mathbf{G}, \mathbf{C} \in \mathbb{R}^{(m+p)\times(m+p)}$, $\mathbf{B} \in \mathbb{R}^{(m+p)\times p}$. Let the nodes $\mathbf{x}$ be split into terminal nodes $\mathbf{x}_p$ to

be preserved, and $\mathbf{x}_I$ internal nodes to be eliminated:

$$\left(\begin{bmatrix} \mathbf{G}_I & \mathbf{G}_C \\ \mathbf{G}_C^T & \mathbf{G}_P \end{bmatrix} + s \begin{bmatrix} \mathbf{C}_I & \mathbf{C}_K \\ \mathbf{C}_K^T & \mathbf{C}_P \end{bmatrix} \right) \begin{bmatrix} \mathbf{x}_I \\ \mathbf{x}_P \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{B}_P \end{bmatrix} \mathbf{u}. \tag{3.2}$$

Let the following projection:

$$\mathbf{V} = \begin{bmatrix} -\mathbf{G}_I^{-1}\mathbf{G}_C \\ \mathbf{I} \end{bmatrix}, \tag{3.3}$$

where $\mathbf{I} \in \mathbb{R}^{p \times p}$ is the identity matrix, be applied to (3.2). From simple linear algebra, the Galerkin projection: $\widehat{\mathbf{G}} = \mathbf{V}^T\mathbf{G}\mathbf{V}$, $\widehat{\mathbf{C}} = \mathbf{V}^T\mathbf{C}\mathbf{V}$, $\widehat{\mathbf{B}} = \mathbf{V}^T\mathbf{B}$ reduces (3.2) to:

$$(\widehat{\mathbf{G}} + s\widehat{\mathbf{C}})\mathbf{x}_P = \widehat{\mathbf{B}}\mathbf{u}, \text{ where :} \tag{3.4}$$

$$\widehat{\mathbf{G}} = \mathbf{G}_P - \mathbf{G}_C^T\mathbf{G}_I^{-1}\mathbf{G}_C \quad , \quad \widehat{\mathbf{B}} = \mathbf{B}_P \tag{3.5}$$

$$\widehat{\mathbf{C}} = \mathbf{C}_P + \mathbf{C}_C^T\mathbf{W} + \mathbf{W}^T\mathbf{C}_C + \mathbf{W}^T\mathbf{C}_I\mathbf{W} \quad , \quad \mathbf{W} = -\mathbf{G}_I^{-1}\mathbf{G}_C. \tag{3.6}$$

The projection matrix (3.3) is the one governing ReduceR [80] for the reduction of resistor networks, and PACT [56] for *RC* networks. An important property of (3.3) is that it preserves the path resistance (defined in Sect. 3.3.3) of the original network (3.2). Especially relevant for *RC* reduction is that (3.3) preserves the 0'th and 1'st moments at *DC* ($s = 0$) of the original network, where the path resistance is precisely the first moment. This is formalized as Proposition 4.2.1 in Chapter 4, where a more detailed presentation of PACT is provided. Furthermore, being a congruence transformation acting on non-negative definite matrices, (3.3) preserves the passivity of (3.2) (the proof is provided in [56]). Finally, notice that, since $\widehat{\mathbf{B}} = \mathbf{B}_P$, the input incidence matrix of the reduced network is the same as in the original, and so the connectivity via terminal nodes is preserved (the importance of this feature, also known as *input/output structure preservation* in the context of synthesis is discussed in Chapter 2).

In the following sections, the solution to two problems is presented. In Sect. 3.2, it is shown that the resistor network characterized by the reduced conductance matrix (3.5) contains only positive resistor values. This result guarantees that the ReduceR [80] methodology generates positive-only resistors, and also holds for the multi-terminal *RC* reduction methods of Sect. 3.3 and Chapter 4. In Sect. 3.3, a partition-based reduction of multi-terminal *RC* networks is derived, shown to be mathematically equivalent to PACT, thus inheriting its afore-mentioned properties. The partition-based reduction provides additional advantages such as: the ability to split a network into sub-parts and reduce them individually while satisfying the same requirements on accuracy, moment matching, passivity preservation and terminal connectivity.

3.2 Reduction of R/RC networks with positive-only resistors

The graph-based multi-terminal model reduction methods for R [80] and RC networks [49] are based on the projection (3.3). Notice from (3.5) that the reduced $\widehat{\mathbf{G}}$ is the Schur complement of block $\mathbf{G}_I$ in $\mathbf{G}$. It is demonstrated here that, if the resistors in the original network are positive, the resistors in the reduced circuit characterized by $\widehat{\mathbf{G}}$ are also guaranteed to be positive (for a different proof derived independently see [91]). As a special case, it is also shown that if the original network has no resistors connected to the ground node, the reduced network will also have no resistors to ground. For simplicity of presentation, the following derivations are based on the reduction of purely resistive networks. The generalization to RC follows immediately, as the underlying projection matrix (3.3) is the same in both cases.

Consider a resistor network described by the conductance matrix $\mathbf{G}$ of (3.1) with $n = m + p$ nodes [from which m are internal nodes and p are terminals (external nodes)]. Assume that the graph associated with this resistor network is strongly connected, i.e. there is a path of resistive connections between every pair of nodes in the network. Otherwise the network is split into its strongly connected components, and the derivations are applied per component; this is the subject of Sect 3.3. Naturally, assume also that all resistors in the original network are positive. We show that if the original conductance matrix $\mathbf{G}$ is characterized by positive resistors only, the reduced conductance matrix $\widehat{\mathbf{G}} = \mathbf{G}_P - \mathbf{G}_C^T \mathbf{G}_I^{-1} \mathbf{G}_C$ is unstamped into positive resistors as well.

3.2.1 Properties of G and unstamping

Before giving the proof, we describe the structure of the conductance matrix $\mathbf{G}$ and identify its special properties. $\mathbf{G}$ has the following form:

$$\mathbf{G} = \begin{bmatrix} g_{1,1} & -g_{1,2} & \cdots & -g_{1,n} \\ -g_{2,1} & g_{2,2} & \cdots & -g_{2,n} \\ \vdots & \vdots & \cdots & \vdots \\ -g_{n,1} & -g_{n,2} & \cdots & g_{n,n} \end{bmatrix}, \tag{3.7}$$

where $g_{i,j} > 0$ for $\forall i, j = \overline{1,n}$[1]. The resistor values in the network and their topology are unstamped ("read-off") from $\mathbf{G}$ as follows:

1. the resistor between node i and node j is: $r_{i,j} = g_{i,j}^{-1}$, $i \neq j$.

[1] $i = \overline{1,n}$ stands for $i = 1 \ldots n$

2. the resistor between node i and the ground node is: $r_i = \left(g_{i,i} - \sum_{j=1, j\neq i}^{n} g_{i,j} \right)^{-1}$;
   - if there is no resistor between node i and ground ($r_i = \infty$), then the sum of the i'th row of $\mathbf{G}$ is zero: $g_{i,i} - \sum_{j=1, j\neq i}^{n} g_{i,j} = 0$;

Throughout the following derivations, we denote as $\mathbf{G}_{i,j}$ the (i,j)'th entry in matrix $\mathbf{G}$, thus $\mathbf{G}_{i,j} = -g_{i,j}$ for $i \neq j$ and $\mathbf{G}_{i,i} = g_{i,i}$. The properties of a resistor network characterized by the conductance matrix (3.7) are summarized in Claim 3.2.1.

Claim 3.2.1 *The conductance matrix $\mathbf{G}$ of a resistive network with positive resistors satisfies the following properties:*

1. $\mathbf{G} \geq 0$, $\mathbf{G} = \mathbf{G}^T$ *i.e., $\mathbf{G}$ is symmetric positive semi-definite*
   (a) *If the network is grounded (i.e. voltages are measured with respect to a reference node), then $\mathbf{G} > 0$, i.e., $\mathbf{G}$ is positive definite*
2. $\mathbf{G}_{i,i} > 0$, $i = \overline{1,n}$, *i.e., diagonal entries are positive*
3. $\mathbf{G}_{i,j} \leq 0$, $i, j = \overline{1,n}, i \neq j$, *i.e., off-diagonal entries are negative (or 0) and at least one is non-zero*
4. $\mathbf{G}$ *is diagonally dominant, i.e :*

$$|\mathbf{G}_{i,i}| \geq \sum_{j=1, j\neq i}^{n} |\mathbf{G}_{i,j}| \tag{3.8}$$

   (a) *If the network is ungrounded, then (3.8) is satisfied with equality (the row/column sum of $\mathbf{G}$ is zero).*

3.2.2 Unstamping the reduced $\widehat{\mathbf{G}}$

For the reduced resistive network to be characterized by positive resistors only, $\widehat{\mathbf{G}}$ must also satisfy the properties of Claim (3.2.1). We prove the following:

Theorem 3.2.1 *Given a conductance matrix $\mathbf{G}$ which satisfies Claim 3.2.1, the reduced matrix $\widehat{\mathbf{G}} = \mathbf{G}_P - \mathbf{G}_C^T \mathbf{G}_I^{-1} \mathbf{G}_C$ obtained by eliminating the internal nodes from the network also satisfies Claim 3.2.1. Hence the reduced $\widehat{\mathbf{G}}$ is unstamped into positive resistors only.*

Proof 3.2.1 *The proof follows by induction. To this end, we introduce the following notation: let $\mathbf{G}^{(1)}, \mathbf{G}^{(2)}, \ldots, \mathbf{G}^{(m)}$ denote the reduced matrices obtained by eliminating one, two, ..., m internal nodes respectively from $\mathbf{G}$. With this notation, $\mathbf{G}^{(0)} = \mathbf{G}$ is the original conductance matrix and $\mathbf{G}^{(m)} = \widehat{\mathbf{G}}$ is the final reduced conductance matrix, with all m internal nodes eliminated.*

Case $m = 1$ eliminated internal node

We show that Theorem (3.2.1) holds for the reduced network obtained by eliminating the first internal node.

$$\mathbf{G} = \left[\begin{array}{c|ccc} \mathbf{G}_{1,1} & \mathbf{G}_{1,2} & \cdots & \mathbf{G}_{1,n} \\ \hline \mathbf{G}_{2,1} & \mathbf{G}_{2,2} & \cdots & \mathbf{G}_{2,n} \\ \vdots & \vdots & \cdots & \vdots \\ \mathbf{G}_{n,1} & \mathbf{G}_{n,2} & \cdots & \mathbf{G}_{n,n} \end{array}\right] = \left[\begin{array}{c|c} \mathbf{G}_{1,1} & \mathbf{G}_{c_1} \\ \hline \mathbf{G}_{c_1}^T & \mathbf{G}_{P_1} \end{array}\right], \quad \mathbf{X}_1 = \begin{bmatrix} 1 & -\mathbf{G}_{1,1}^{-1}\mathbf{G}_{c_1} \\ \mathbf{0} & \mathbf{I}_{n-1} \end{bmatrix} \tag{3.9}$$

$$\widetilde{\mathbf{G}}_1 = \mathbf{X}_1^T\mathbf{G}\mathbf{X}_1 = \left[\begin{array}{c|c} \mathbf{G}_{1,1} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{G}_{P_1} - \mathbf{G}_{1,1}^{-1}\mathbf{G}_{c_1}^T\mathbf{G}_{c_1} \end{array}\right]. \tag{3.10}$$

Let $\mathbf{G}^{(1)} := \mathbf{G}_{P_1} - \mathbf{G}_{1,1}^{-1}\mathbf{G}_{c_1}^T\mathbf{G}_{c_1}$, $\mathbf{G}^{(1)} \in \mathbb{R}^{(n-1)\times(n-1)}$, which represents the reduced network with the first internal node eliminated. We have:

$$\mathbf{G}^{(1)} = \mathbf{G}_{P_1} - \mathbf{G}_{1,1}^{-1}\mathbf{G}_{c_1}^T\mathbf{G}_{c_1} = \tag{3.11}$$

$$= \begin{bmatrix} \mathbf{G}_{2,2} - \frac{1}{\mathbf{G}_{1,1}}\mathbf{G}_{2,1}\mathbf{G}_{1,2} & \mathbf{G}_{2,3} - \frac{1}{\mathbf{G}_{1,1}}\mathbf{G}_{2,1}\mathbf{G}_{1,3} & \cdots & \mathbf{G}_{2,n} - \frac{1}{\mathbf{G}_{1,1}}\mathbf{G}_{2,1}\mathbf{G}_{1,n} \\ \mathbf{G}_{3,2} - \frac{1}{\mathbf{G}_{1,1}}\mathbf{G}_{3,1}\mathbf{G}_{1,2} & \mathbf{G}_{3,3} - \frac{1}{\mathbf{G}_{1,1}}\mathbf{G}_{3,1}\mathbf{G}_{1,3} & \cdots & \mathbf{G}_{3,n} - \frac{1}{\mathbf{G}_{1,1}}\mathbf{G}_{3,1}\mathbf{G}_{1,n} \\ \vdots & \vdots & \cdots & \vdots \\ \mathbf{G}_{n,2} - \frac{1}{\mathbf{G}_{1,1}}\mathbf{G}_{n,1}\mathbf{G}_{1,2} & \mathbf{G}_{n,3} - \frac{1}{\mathbf{G}_{1,1}}\mathbf{G}_{n,1}\mathbf{G}_{1,3} & \cdots & \mathbf{G}_{n,n} - \frac{1}{\mathbf{G}_{1,1}}\mathbf{G}_{n,1}\mathbf{G}_{1,n} \end{bmatrix} \tag{3.12}$$

We show that $\mathbf{G}^{(1)}$ as in (3.12) satisfies the properties of Claim 3.2.1:

1. $\mathbf{G}^{(1)} \geq 0$, *i.e.*, $\mathbf{G}^{(1)}$ *is positive semi-definite:*
   $\mathbf{X}_1$ *from (3.10) is a congruence transformation, hence, because* $\mathbf{G} \geq 0$, $\widetilde{\mathbf{G}}_1 \geq 0$. *This implies that* $\forall \mathbf{Y}_1$, $\mathbf{Y}_1^T\widetilde{\mathbf{G}}_1\mathbf{Y}_1 \geq 0$. *In particular, let* $\mathbf{Y}_1 = \begin{bmatrix} \mathbf{0} \\ \mathbf{I} \end{bmatrix} \Rightarrow$

   $$\mathbf{Y}_1^T\widetilde{\mathbf{G}}_1\mathbf{Y}_1 = \mathbf{G}_{P_1} - \mathbf{G}_{1,1}^{-1}\mathbf{G}_{c_1}^T\mathbf{G}_{c_1} = \mathbf{G}^{(1)} \geq 0.$$

   ■

2. $\mathbf{G}^{(1)}{}_{k,k} > 0$, $k = \overline{1, n-1}$, *i.e.* $\mathbf{G}^{(1)}$ *has positive diagonal entries:*
   This follows easily from the fact that $\mathbf{G}^{(1)} \geq 0$ *is positive definite, thus* $\mathbf{y}^T\mathbf{G}^{(1)}\mathbf{y} \geq 0$ *for* $\forall \mathbf{y} \in \mathbb{C}^{n-1}$. *In particular let* $\mathbf{y} = \mathbf{e}_k$, *i.e. the k'th unit vector. Then* $\mathbf{G}^{(1)}{}_{k,k} = \mathbf{e}_k^T\mathbf{G}^{(1)}\mathbf{e}_k > 0$.
   Note that $\mathbf{e}_k^T\mathbf{G}^{(1)}\mathbf{e}_k \neq 0$. *To show this assume for a moment that* $\mathbf{e}_k^T\mathbf{G}^{(1)}\mathbf{e}_k = 0$. *Then:*

   $$\mathbf{e}_k^T\mathbf{G}^{(1)}\mathbf{e}_k = \mathbf{G}^{(1)}{}_{k,k} = \mathbf{G}_{i,i} - \frac{1}{\mathbf{G}_{1,1}}\mathbf{G}_{i,1}\mathbf{G}_{1,i} = 0, \quad i = \overline{2, n} \quad \Leftrightarrow \tag{3.13}$$

   $$\Leftrightarrow \quad \mathbf{G}_{i,i}\mathbf{G}_{1,1} = \mathbf{G}_{i,1}\mathbf{G}_{1,i} \tag{3.14}$$

Due to the diagonal dominance (3.8), we know $\mathbf{G}_{i,i} > |\mathbf{G}_{i,1}|$ and $\mathbf{G}_{1,1} > |\mathbf{G}_{i,1}|$. Hence (3.14) is false, and the assumption that $\mathbf{e}_k^T \mathbf{G}^{(1)} \mathbf{e}_k = \mathbf{G}^{(1)}{}_{k,k} = 0$ is false.
N. B. *A final note on the possibility of (3.14) being true: this holds only if $\mathbf{G}_{i,i} = \mathbf{G}_{1,1} = |\mathbf{G}_{i,1}| = |\mathbf{G}_{i,1}|$ (recall from $\mathbf{G} = \mathbf{G}^T$ that $|\mathbf{G}_{i,1}| = |\mathbf{G}_{i,1}|$). This would mean that, prior to the elimination of node 1, nodes i and node 1 form one strongly connected component, with a resistor of value $\frac{1}{\mathbf{G}_{1,1}}$ as the edge between them, and no other outgoing edges from nodes 1 and i to the other circuit nodes. This however violates the assumption that $\mathbf{G}$ forms one strongly connected component.* ■

3. $\mathbf{G}^{(1)}{}_{k,l} \leq 0$ *for* $k, l = \overline{1, n-1}, k \neq l$ *i.e.* $\mathbf{G}^{(1)}$ *has non-positive off-diagonal entries:*

$$\mathbf{G}^{(1)}{}_{\substack{k,l \\ k \neq l}} = \mathbf{G}_{i,j} - \frac{1}{\mathbf{G}_{1,1}} \mathbf{G}_{i,1} \mathbf{G}_{1,j}, \;\; i, j = \overline{2, n}, \;\; i \neq j$$

Clearly, from the properties of $\mathbf{G}$ *(see Claim 3.2.1) for* $i, j = \overline{2, n}, \;\; i \neq j$ *we know that:* $\mathbf{G}_{i,j} \leq 0$, $\mathbf{G}_{i,1} \leq 0$, $\mathbf{G}_{1,j} \leq 0$ *and* $\mathbf{G}_{1,1} > 0$, *thus:*

$$\mathbf{G}_{i,j} - \frac{1}{\mathbf{G}_{1,1}} \mathbf{G}_{i,1} \mathbf{G}_{1,j} \leq 0.$$

■

4. $\mathbf{G}^{(1)}$ *is diagonally dominant*[2]*, i.e. :*

$$|\mathbf{G}^{(1)}{}_{k,k}| \geq \sum_{l=1, l \neq k}^{n-1} |\mathbf{G}^{(1)}{}_{k,l}|. \tag{3.15}$$

This is equivalent to showing that, for $i \geq 2$*:*

$$|\mathbf{G}_{i,i} - \frac{1}{\mathbf{G}_{1,1}} \mathbf{G}_{i,1} \mathbf{G}_{1,i}| \geq \sum_{j=2, j \neq i}^{n} |\mathbf{G}_{i,j} - \frac{1}{\mathbf{G}_{1,1}} \mathbf{G}_{i,1} \mathbf{G}_{1,j}|. \tag{3.16}$$

Using triangle's inequality:

$$\sum_{j=2, j \neq i}^{n} |\mathbf{G}_{i,j} - \frac{1}{\mathbf{G}_{1,1}} \mathbf{G}_{i,1} \mathbf{G}_{1,j}| \leq \underbrace{\sum_{j=2, j \neq i}^{n} |\mathbf{G}_{i,j}|}_{:= S_1} + \underbrace{\sum_{j=2, j \neq i}^{n} |\frac{1}{\mathbf{G}_{1,1}} \mathbf{G}_{i,1} \mathbf{G}_{1,j}|}_{:= S_2} \tag{3.17}$$

Recall from Claim 3.2.1 that $\mathbf{G}$ *is diagonally dominant, thus:*

$$|\mathbf{G}_{i,i}| \;\; \geq \;\; \sum_{j=1, j \neq i}^{n} |\mathbf{G}_{i,j}| = |G_{i,1}| + \sum_{j=2, j \neq i}^{n} |\mathbf{G}_{i,j}| \Rightarrow \tag{3.18}$$

$$\Rightarrow \;\; S_1 \leq |\mathbf{G}_{i,i}| - |\mathbf{G}_{i,1}| \tag{3.19}$$

[2]Part of this proof is inspired from a homework assigned during the course CAAM 453/553 "Numerical Analysis I", taken at Rice University in 2005, taught by Prof. Mark Embree.

Also:

$$S_2 = \sum_{j=2, j\neq i}^{n} |\frac{1}{\mathbf{G}_{1,1}}\mathbf{G}_{i,1}\mathbf{G}_{1,j}| = |\frac{1}{\mathbf{G}_{1,1}}\mathbf{G}_{i,1}| \sum_{j=2, j\neq i}^{n} |\mathbf{G}_{1,j}|. \tag{3.20}$$

Again, from $\mathbf{G}$ *diagonally dominant, we know:*

$$|\mathbf{G}_{1,1}| \geq \sum_{j=1, j\neq i}^{n} |\mathbf{G}_{1,j}| = |\mathbf{G}_{1,i}| + \sum_{j=2, j\neq i}^{n} |\mathbf{G}_{1,j}| \Rightarrow \tag{3.21}$$

$$\Rightarrow \sum_{j=2, j\neq i}^{n} |\mathbf{G}_{1,j}| \leq \mathbf{G}_{1,1} - |\mathbf{G}_{1,i}| \tag{3.22}$$

Inserting (3.22) into (3.20) yields:

$$S_2 \leq |\frac{1}{\mathbf{G}_{1,1}}\mathbf{G}_{i,1}|(|\mathbf{G}_{1,1}| - |\mathbf{G}_{1,i}|) = |\mathbf{G}_{i,1}| - \frac{1}{\mathbf{G}_{1,1}}|\mathbf{G}_{i,1}\mathbf{G}_{1,i}|. \tag{3.23}$$

Adding (3.19) and (3.23):

$$S_1 + S_2 \leq |\mathbf{G}_{i,i}| - \frac{1}{\mathbf{G}_{1,1}}|\mathbf{G}_{i,1}\mathbf{G}_{1,i}| \leq |\mathbf{G}_{i,i} - \frac{1}{\mathbf{G}_{1,1}}\mathbf{G}_{i,1}\mathbf{G}_{1,i}| \tag{3.24}$$

Recalling (3.17), we conclude that for $i \geq 2$*:*

$$\sum_{j=2, j\neq i}^{n} |\mathbf{G}_{i,j} - \frac{1}{\mathbf{G}_{1,1}}\mathbf{G}_{i,1}\mathbf{G}_{1,j}| \leq S_1 + S_2 \leq |\mathbf{G}_{i,i} - \frac{1}{\mathbf{G}_{1,1}}\mathbf{G}_{i,1}\mathbf{G}_{1,i}|,$$

which is nothing but (3.16). Hence $\mathbf{G}^{(1)}$ *is diagonally dominant.* ■

(a) We also show that if the original network in ungrounded [item $4(a)$ *of Claim (3.2.1) holds for* $\mathbf{G}$*], then the network determined by* $\mathbf{G}^{(1)}$ *is also ungrounded. In other words, if the row sum of* $\mathbf{G}$ *is zero, then the row sum of* $\mathbf{G}^{(1)}$ *is also zero, .i.e. :*

$$|{\mathbf{G}^{(1)}}_{k,k}| = \sum_{l=1, l\neq k}^{n-1} |{\mathbf{G}^{(1)}}_{k,l}| \tag{3.25}$$

From ${\mathbf{G}^{(1)}}_{k,k} > 0$ *and* ${\mathbf{G}^{(1)}}_{k,l} \leq 0$ *for* $k \neq l$ *(3.25) writes:*

$${\mathbf{G}^{(1)}}_{k,k} + \sum_{l=1, l\neq k}^{n-1} {\mathbf{G}^{(1)}}_{k,l} = 0 \Leftrightarrow \tag{3.26}$$

$$\mathbf{G}_{i,i} - \frac{1}{\mathbf{G}_{1,1}}\mathbf{G}_{i,1}\mathbf{G}_{1,i} + \sum_{j=2, j\neq i}^{n} \left[\mathbf{G}_{i,j} - \frac{1}{\mathbf{G}_{1,1}}\mathbf{G}_{i,1}\mathbf{G}_{1,j}\right] = 0 \Leftrightarrow \tag{3.27}$$

$$\left[\mathbf{G}_{i,i} + \sum_{j=2, j\neq i}^{n} \mathbf{G}_{i,j}\right] - \frac{1}{\mathbf{G}_{1,1}}\mathbf{G}_{i,1}\left[\mathbf{G}_{1,i} + \sum_{j=2, j\neq i}^{n} \mathbf{G}_{1,j}\right] = 0 \tag{3.28}$$

From (3.12), note that the above derivations hold for $i \geq 2$. Also recall the original assumption that the row/column sums of $\mathbf{G}$ are zero.
From the i'th row sum of $\mathbf{G}$ being zero, we have:

$$\mathbf{G}_{i,i} + \sum_{j=1, j\neq i}^{n} \mathbf{G}_{i,j} = 0 \quad \Rightarrow \quad \mathbf{G}_{i,1} = -\left[\mathbf{G}_{i,i} + \sum_{j=2, j\neq i}^{n} \mathbf{G}_{i,j}\right] \tag{3.29}$$

From the 1'st row sum of $\mathbf{G}$ being zero, we have:

$$\mathbf{G}_{1,i} + \sum_{j=1, j\neq i}^{n} \mathbf{G}_{1,j} = 0 \quad \Rightarrow \quad \mathbf{G}_{1,1} = -\left[\mathbf{G}_{1,i} + \sum_{j=2, j\neq i}^{n} \mathbf{G}_{1,j}\right] \tag{3.30}$$

Replacing (3.29) and (3.30) in (3.28) we arrive at:

$$-\mathbf{G}_{i,1} + \frac{1}{\mathbf{G}_{1,1}} \mathbf{G}_{i,1}\mathbf{G}_{i,1} = 0 \quad \Leftrightarrow \quad 0 \;=\; 0. \tag{3.31}$$

This shows that (3.25) holds, and that, if the row sum of $\mathbf{G}$ is zero, the row sum of $\mathbf{G}^{(1)}$ is also zero. This implies that the reduced $\mathbf{G}^{(1)}$ has no resistors to ground. ■

With all four properties of Claim (3.2.1) being shown for $\mathbf{G}^{(1)}$, this concludes the proof of the first induction step.

Case $m-1$ eliminated internal nodes

Assume that the reduced $\mathbf{G}^{(m-1)} \in \mathbb{R}^{(p+1)\times(p+1)}$ obtained by eliminating $m-1$ internal nodes satisfies the properties of Claim 3.2.1. Hence $\mathbf{G}^{(m-1)}$ is characterized by positive only resistors.

Case m eliminated internal nodes

At the m'th elimination step, the final reduced matrix $\mathbf{G}^{(m)}$ is obtained from $\mathbf{G}^{(m-1)}$ as follows.

$$\begin{aligned}
\mathbf{G}^{(m-1)} &= \left[\begin{array}{c|ccc} \mathbf{G}_{1,1}^{(m-1)} & \mathbf{G}_{1,2}^{(m-1)} & \cdots & \mathbf{G}_{1,p+1}^{(m-1)} \\ \hline \mathbf{G}_{2,1}^{(m-1)} & \mathbf{G}_{2,2}^{(m-1)} & \cdots & \mathbf{G}_{2,p+1}^{(m-1)} \\ \vdots & \vdots & \cdots & \vdots \\ \mathbf{G}_{p+1,1}^{(m-1)} & \mathbf{G}_{p+1,2}^{(m-1)} & \cdots & \mathbf{G}_{p+1,p+1}^{(m-1)} \end{array}\right] = \left[\begin{array}{c|c} \mathbf{G}_{1,1}^{(m-1)} & \mathbf{G}_{c_1}^{(m-1)} \\ \hline \mathbf{G}_{c_1}^{(m-1)^T} & \mathbf{G}_{P_1}^{(m-1)} \end{array}\right], \\
\mathbf{X}_m &= \begin{bmatrix} 1 & -\frac{1}{\mathbf{G}_{1,1}^{(m-1)}}\mathbf{G}_{c_1} \\ \mathbf{0} & \mathbf{I}_m \end{bmatrix}, \\
\widetilde{\mathbf{G}}_m &= \mathbf{X}_m^T \mathbf{G}^{(m-1)} \mathbf{X}_m = \left[\begin{array}{c|c} \mathbf{G}_{1,1}^{(m-1)} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{G}_{P_1}^{(m-1)} - \frac{1}{\mathbf{G}_{1,1}^{(m-1)}} \mathbf{G}_{c_1}^{(m-1)^T} \mathbf{G}_{c_1}^{(m-1)} \end{array}\right].
\end{aligned}$$

Then $\mathbf{G}^{(m)} := \mathbf{G}_{P_1}^{(m-1)} - \frac{1}{\mathbf{G}_{1,1}^{(m-1)}} \mathbf{G}_{c_1}^{(m-1)^T} \mathbf{G}_{c_1}^{(m-1)}$, $\mathbf{G}^{(m)} \in \mathbb{R}^{p \times p}$, *which represents the reduced network with all m internal nodes eliminated. Similarly to the case of one eliminated internal node (3.12), we have:*

$$\mathbf{G}^{(m)} = \mathbf{G}_{P_1}^{(m-1)} - \frac{1}{\mathbf{G}_{1,1}^{(m-1)}} \mathbf{G}_{c_1}^{(m-1)^T} \mathbf{G}_{c_1}^{(m-1)} = \tag{3.32}$$

$$= \begin{bmatrix} \mathbf{G}_{2,2}^{(m-1)} - \frac{1}{\mathbf{G}_{1,1}^{(m-1)}} \mathbf{G}_{2,1}^{(m-1)} \mathbf{G}_{1,2}^{(m-1)} & \cdots & \mathbf{G}_{2,p+1}^{(m-1)} - \frac{1}{\mathbf{G}_{1,1}^{(m-1)}} \mathbf{G}_{2,1}^{(m-1)} \mathbf{G}_{1,p+1} \\ \vdots & \ddots & \vdots \\ \mathbf{G}_{p+1,2}^{(m-1)} - \frac{1}{\mathbf{G}_{1,1}^{(m-1)}} \mathbf{G}_{p+1,1}^{(m-1)} \mathbf{G}_{1,2}^{(m-1)} & \cdots & \mathbf{G}_{p+1,p+1}^{(m-1)} - \frac{1}{\mathbf{G}_{1,1}^{(m-1)}} \mathbf{G}_{p+1,1}^{(m-1)} \mathbf{G}_{1,p+1}^{(m-1)} \end{bmatrix}. \tag{3.33}$$

Note that (3.33) has the same structure and properties as (3.12). Hence the proof that $\mathbf{G}^{(m)}$ *satisfies Claim (3.2.1) [based on the fact that* $\mathbf{G}^{(m-1)}$ *satisfies these properties] follows in the same manner as for* $\mathbf{G}^{(1)}$. ■

Having completed the final induction step, we conclude that the final reduced conductance matrix $\widehat{\mathbf{G}} = \mathbf{G}^{(m)}$ inherits the properties of the original $\mathbf{G}$ and is characterized by positive-only resistors. Also, if $\mathbf{G}$ has no resistors to ground, then $\widehat{\mathbf{G}} = \mathbf{G}^{(m)}$ will not have resistors to ground either.

3.3 Partition-based reduction of RC networks

In this section, a partition-based reduction for multi-terminal RC networks is derived, and its equivalence to PACT [56] is demonstrated. A change in notation is introduced from bold to caligraphic letters, as to more easily distinguish between blocks of an unpartitioned vs. a partitioned matrix respectively. The starting system matrices underlying (3.1) are now $\mathcal{G}, \mathcal{C}, \mathcal{B}$, to be partitioned and reduced as described next. Let $\mathcal{Q}$ be the permutation of the circuit nodes which reveals the strongly connected components[3] (SCCs) of $\mathcal{G}$, $\mathcal{Q}\mathcal{Q}^T = \mathbf{I}$. The original system permuted with $\mathcal{Q}$ is then:

$$(\mathcal{Q}^T \mathcal{G} \mathcal{Q} + s \mathcal{Q}^T \mathcal{C} \mathcal{Q}) \mathcal{Q}^T \mathbf{x}(s) = \mathcal{Q}^T \mathcal{B} \mathbf{u}(s) \Leftrightarrow \tag{3.34}$$

$$\left(\begin{bmatrix} \mathcal{G}_1 & \mathbf{0} \\ \mathbf{0} & \mathcal{G}_2 \end{bmatrix} + s \begin{bmatrix} \mathcal{C}_1 & \mathcal{C}_{12} \\ \mathcal{C}_{12}^T & \mathcal{C}_2 \end{bmatrix} \right) \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} = \begin{bmatrix} \mathcal{B}_1 \\ \mathcal{B}_2 \end{bmatrix} \tag{3.35}$$

If $\mathcal{G}$ has N SCCs, in (3.35) $\mathcal{G}_1$ denotes the first SCC of $\mathcal{G}$, and the remaining $N-1$ SCCs are grouped into $\mathcal{G}_2 = \text{blockdiag}(\mathcal{G}_{2,2}, \ldots, \mathcal{G}_{N,N})$. This splitting separates the original network into sub-networks that are only connected by capacitors: $\mathcal{C}_{12}$ is the connectivity block from subnet-1 (specified by $\mathcal{G}_1$, $\mathcal{C}_1$) to subnet-2 (specified by $\mathcal{G}_2$, $\mathcal{C}_2$). With (3.35) a

[3] A graph is called (strongly) connected if there exists a path from any vertex to any other vertex in the graph [19].

network topology has been identified, where subnetwork $\mathcal{G}_1$ can be reduced separately from the remaining $\mathcal{G}_2$. Even though $\mathcal{G}_2$ further contains SCCs of $\mathcal{G}$, it is interpreted as one block during the reduction of $\mathcal{G}_1$. $\mathcal{B}_1$ and $\mathcal{B}_2$ contain the input incidence of current injections into the terminals of subnetwork 1 and 2 respectively. Note that the number of terminals from the original netlist is split in between subnetwork 1 and 2. Thus the subnetworks have fewer nodes and terminals than the original. For clarity, from here onwards $\mathcal{G}$ and $\mathcal{C}$ are assumed to be already permuted according to the SCCs of $\mathcal{G}$, and have the structure (3.35).

3.3.1 Reduction per subnetwork

Let $\mathcal{V}$ = blockdiag($\mathcal{V}_1$, $\mathbf{I}_2$) be the projection reducing the original network (3.35), where $\mathcal{V}_1$ reduces subnetwork-1 and $\mathbf{I}_2$ is the identity matrix keeping subnetwork-2 unreduced. Projecting (3.35) with $\mathcal{V}$ one obtains the reduced model (4.13,4.14) in the form:

$$\left(\begin{bmatrix} \widehat{\mathcal{G}}_1 & \mathbf{0} \\ \mathbf{0} & \mathcal{G}_2 \end{bmatrix} + s \begin{bmatrix} \widehat{\mathcal{C}}_1 & \widehat{\mathcal{C}}_{12} \\ \widehat{\mathcal{C}}_{12}^T & \mathcal{C}_2 \end{bmatrix} \right) \begin{bmatrix} \widehat{\mathbf{x}}_1 \\ \mathbf{x}_2 \end{bmatrix} = \begin{bmatrix} \widehat{\mathcal{B}}_1 \\ \mathcal{B}_2 \end{bmatrix}, \tag{3.36}$$

where subnetwork-1 is reduced to:

$$\widehat{\mathcal{G}}_1 = \mathcal{V}_1^T \mathcal{G}_1 \mathcal{V}_1, \ \widehat{\mathcal{C}}_1 = \mathcal{V}_1^T \mathcal{C}_1 \mathcal{V}_1 \tag{3.37}$$

$$\widehat{\mathcal{C}}_{12} = \mathcal{V}_1^T \mathcal{C}_{12}, \ \widehat{\mathcal{B}}_1 = \mathcal{V}_1^T \mathcal{B}_1, \ \widehat{\mathbf{x}}_1 = \mathcal{V}_1^T \mathbf{x}_1, \tag{3.38}$$

The appropriate $\mathcal{V}_1$ can be chosen according to the preferred MOR method. Here, if the ratio #terminals/#nodes in subnetwork-1 is small, $\mathcal{V}_1$ is constructed with the method from Sect. 4.2.2 as follows.

The initial partitioning according to the SCCs of $\mathcal{G}$ has split the network in such a way, that applying PACT successively on all subnetworks (i.e. SCCs of $\mathcal{G}$) would be equivalent to applying PACT directly on the full circuit. This is formalized in Theorem 3.3.1. The partitioning has computational and structural advantages: (a) PACT reduction applied on each subnetwork becomes cheaper computationally than on the full problem, (b) the matrix fill-in introduced by PACT can be monitored on each subnetwork separately (see Sect. 3.3.5).

Theorem 3.3.1 *Given is an RC circuit as in (3.35), partitioned into subnetworks corresponding to the strongly components of the conductance matrix $\mathcal{G}$. The reduced model obtained from applying PACT successively on each subnetwork of (3.35) is the same (up to permutations) as the reduced model obtained from applying PACT directly on (4.12).*

Proof 3.3.1 *The proof follows by induction. We show that the claim of Theorem 3.3.1 holds while reducing the subnetwork corresponding to the first strongly connected component (SCC) of $\mathcal{G}$. Then, assuming that $N-1$ subnets corresponding to the first $N-1$ SCCs of $\mathcal{G}$ have been reduced, we show that Theorem 3.3.1 holds when reducing subnet N corresponding to the N'th SCC of $\mathcal{G}$.*

Step 1

At the first reduction step, we have subnetwork-1 defined by $\mathcal{G}_1 := \mathcal{G}_{1,1}$ (the first SCC of $\mathcal{G}$), and subnetwork-2 defined by $\mathcal{G}_2$:=blockdiag($\mathcal{G}_{2,2}, \dots \mathcal{G}_{N,N}$) [the remaining $N-1$ SCCs of $\mathcal{G}$]:

$$\mathcal{G} = \left[\begin{array}{c|cccc} \mathcal{G}_{1,1} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} \\ \hline \mathbf{0} & \mathcal{G}_{2,2} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathcal{G}_{3,3} & \dots & \mathbf{0} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathcal{G}_{N,N} \end{array}\right] := \left[\begin{array}{c|c} \mathcal{G}_1 & \mathbf{0} \\ \hline \mathbf{0} & \mathcal{G}_2 \end{array}\right] \tag{3.39}$$

$$\mathcal{C} = \left[\begin{array}{c|cccc} \mathcal{C}_{1,1} & \mathcal{C}_{1,2} & \mathcal{C}_{1,3} & \dots & \mathcal{C}_{1,N} \\ \hline \mathcal{C}_{1,2}^T & \mathcal{C}_{2,2} & \mathcal{C}_{2,3} & \dots & \mathcal{C}_{2,N} \\ \mathcal{C}_{1,3}^T & \mathcal{C}_{2,3}^T & \mathcal{C}_{3,3} & \dots & \mathcal{C}_{3,N} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathcal{C}_{1,N}^T & \mathcal{C}_{2,N}^T & \mathcal{C}_{3,N}^T & \dots & \mathcal{C}_{N,N} \end{array}\right] := \left[\begin{array}{c|c} \mathcal{C}_1 & \mathcal{C}_{12} \\ \hline \mathcal{C}_{12}^T & \mathcal{C}_2 \end{array}\right], \tag{3.40}$$

The goal is to show that the claim of Theorem 3.3.1 holds when reducing subnetwork-1 while keeping subnetwork-2 unreduced. This will be done by constructing two reduced models, shown to be the same: (1) M1: subnetwork-1 is considered individually and reduced with PACT, and (2) M2: the entire network is considered, from which the internal nodes of subnetwork-1 are eliminated, while the terminals of subnetwork-1 together with all the nodes of subnetwork-2 are preserved. For each of these cases, consider (3.35) with the splitting of nodes: $\mathbf{x} = [\mathbf{x}_1^T, \mathbf{x}_2^T]^T$, where $\mathbf{x}_1 = [\mathbf{x}_{1_R}^T, \mathbf{x}_{1_S}^T]^T$.

1. *M1: Subnetwork-1 is considered individually and reduced with PACT using the partitioning of $\mathbf{x}_1$: into internal nodes to be eliminated $\mathbf{x}_R := \mathbf{x}_{1_R}$, and selected nodes to be preserved $\mathbf{x}_S := \mathbf{x}_{1_S}$. Subnetwork-2 is kept unreduced. To visualize this, consider (3.35) with the following splitting:*

$$\underbrace{\left(\left[\begin{array}{cc|c} \mathcal{G}_R & \mathcal{G}_K & \mathbf{0} \\ \mathcal{G}_K^T & \mathcal{G}_S & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} & \mathcal{G}_2 \end{array}\right] + s\left[\begin{array}{cc|c} \mathcal{C}_R & \mathcal{C}_K & \mathcal{C}_{12_R} \\ \mathcal{C}_K^T & \mathcal{C}_S & \mathcal{C}_{12_S} \\ \hline \mathcal{C}_{12_R}^T & \mathcal{C}_{12_S}^T & \mathcal{C}_2 \end{array}\right]\right)}_{\mathcal{G}+s\mathcal{C}} \underbrace{\left[\begin{array}{c} \mathbf{x}_{1_R} \\ \mathbf{x}_{1_S} \\ \hline \mathbf{x}_2 \end{array}\right]}_{\mathbf{x}} = \underbrace{\left[\begin{array}{c} \mathbf{0} \\ \mathcal{B}_{1_S} \\ \hline \mathcal{B}_2 \end{array}\right]}_{\mathcal{B}} \mathbf{u}. \tag{3.41}$$

2. *M2: The entire original network (3.35) is reduced directly with PACT using the partitioning of $\mathbf{x}$ into: internal nodes to be eliminated $\mathbf{x}_R := \mathbf{x}_{1_R}$, and selected nodes to be preserved $\mathbf{x}_S := [\mathbf{x}_{1_S}^T, \mathbf{x}_2^T]^T$. Again, to visualize consider (3.35) with the corresponding splitting:*

$$\underbrace{\left(\left[\begin{array}{c|cc} \mathcal{G}_R & \mathcal{G}_K & \mathbf{0} \\ \hline \mathcal{G}_K^T & \mathcal{G}_S & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathcal{G}_2 \end{array}\right] + s\left[\begin{array}{c|cc} \mathcal{C}_R & \mathcal{C}_K & \mathcal{C}_{12_R} \\ \hline \mathcal{C}_K^T & \mathcal{C}_S & \mathcal{C}_{12_S} \\ \mathcal{C}_{12_R}^T & \mathcal{C}_{12_S}^T & \mathcal{C}_2 \end{array}\right]\right)}_{\mathcal{G}+s\mathcal{C}} \underbrace{\left[\begin{array}{c} \mathbf{x}_{1_R} \\ \hline \mathbf{x}_{1_S} \\ \mathbf{x}_2 \end{array}\right]}_{\mathbf{x}} = \underbrace{\left[\begin{array}{c} \mathbf{0} \\ \hline \mathcal{B}_{1_S} \\ \mathcal{B}_2 \end{array}\right]}_{\mathcal{B}} \mathbf{u}. \tag{3.42}$$

We show that the PACT transformation $\mathcal{V}$ which reduces the network by eliminating $\mathbf{x}_{R_1}$ is the same for both M1 and M2. Therefore the reduced models M1 and M2 will be the same at the end of Step 1.

1. *In view of obtaining M1, we start from (3.41) and let the reducing projection $\mathcal{V}$ be partitioned into:*

$$\mathcal{V} = \left[\begin{array}{c|c} -\mathcal{G}_R^{-1}\mathcal{G}_K & \mathbf{0} \\ \mathbf{I}_S & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{I}_2 \end{array}\right] := \left[\begin{array}{c|c} \mathcal{V}_1 & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{I}_2 \end{array}\right], \tag{3.43}$$

where $\mathcal{V}_1 := \left[\begin{array}{c} -\mathcal{G}_R^{-1}\mathcal{G}_K \\ \mathbf{I}_S \end{array}\right] := \left[\begin{array}{c} \mathcal{W}_1 \\ \hline \mathbf{I}_S \end{array}\right]$ reduces subnetwork-1 with PACT, and $\mathbf{I}_2$ keeps subnetwork-2 unreduced. The reduced matrices for subnetwork-1 are formed using (3.37-3.38):

$$\begin{aligned} \widehat{\mathcal{G}}_1 &= \mathcal{G}_S - \mathcal{G}_K^T\mathcal{G}_R^{-1}\mathcal{G}_K, \;\; \mathcal{W}_1 := -\mathcal{G}_R^{-1}\mathcal{G}_K & (3.44)\\ \widehat{\mathcal{C}}_1 &= \mathcal{C}_S + \mathcal{W}_1^T\mathcal{C}_R\mathcal{W}_1 + \mathcal{W}_1^T\mathcal{C}_K + \mathcal{C}_K^T\mathcal{W}_1, & (3.45)\\ \widehat{\mathcal{C}}_{12} &= \mathcal{C}_{12_S} + \mathcal{W}_1^T\mathcal{C}_{12_R} & (3.46)\\ \widehat{\mathcal{B}}_1 &= \mathcal{B}_{1_S}, \widehat{\mathbf{x}}_1 = \mathbf{x}'_{1_S}. & (3.47) \end{aligned}$$

The reduced model M1 for (3.41) follows from (3.36):

$$\widehat{\mathcal{G}} = \mathcal{V}^T\mathcal{G}\mathcal{V} = \left[\begin{array}{cc} \widehat{\mathcal{G}}_1 & \mathbf{0} \\ \mathbf{0} & \mathcal{G}_2 \end{array}\right] = \left[\begin{array}{c|cccc} \widehat{\mathcal{G}}_{1,1} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} \\ \hline \mathbf{0} & \mathcal{G}_{2,2} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathcal{G}_{3,3} & \dots & \mathbf{0} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathcal{G}_{N,N} \end{array}\right], \tag{3.48}$$

$$\widehat{\mathcal{C}} = \mathcal{V}^T\mathcal{C}\mathcal{V} = \left[\begin{array}{cc} \widehat{\mathcal{C}}_1 & \widehat{\mathcal{C}}_{12} \\ \widehat{\mathcal{C}}_{12}^T & \mathcal{C}_2 \end{array}\right] = \left[\begin{array}{c|cccc} \widehat{\mathcal{C}}_{1,1} & \widehat{\mathcal{C}}_{1,2} & \widehat{\mathcal{C}}_{1,3} & \dots & \widehat{\mathcal{C}}_{1,N} \\ \hline \widehat{\mathcal{C}}_{1,2}^T & \mathcal{C}_{2,2} & \mathcal{C}_{2,3} & \dots & \mathcal{C}_{2,N} \\ \widehat{\mathcal{C}}_{1,3}^T & \mathcal{C}_{2,3}^T & \mathcal{C}_{3,3} & \dots & \mathcal{C}_{3,N} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \widehat{\mathcal{C}}_{1,N}^T & \mathcal{C}_{2,N}^T & \mathcal{C}_{3,N}^T & \dots & \mathcal{C}_{N,N} \end{array}\right], \tag{3.49}$$

$$\widehat{\mathbf{x}} = \mathcal{V}^T\mathbf{x} = \left[\begin{array}{c} \widehat{\mathbf{x}}_1 \\ \mathbf{x}_2 \end{array}\right], \;\; \widehat{\mathcal{B}} = \mathcal{V}^T\mathcal{B} = \left[\begin{array}{c} \widehat{\mathcal{B}}_1 \\ \mathcal{B}_2 \end{array}\right] \tag{3.50}$$

2. *In view of obtaining M2, consider (3.42) with the following block assignments:*

$$\mathbf{G}_R := \mathcal{G}_R \;\; , \;\; \mathbf{G}_K := \left[\begin{array}{cc} \mathcal{G}_K & \mathbf{0} \end{array}\right], \tag{3.51}$$

$$\mathbf{C}_R := \mathcal{C}_R \;\; , \;\; \mathbf{C}_K := \left[\begin{array}{cc} \mathcal{C}_K & \mathcal{C}_{12_R} \end{array}\right] \tag{3.52}$$

$$\mathbf{G}_S := \left[\begin{array}{cc} \mathcal{G}_S & \mathbf{0} \\ \mathbf{0} & \mathcal{G}_2 \end{array}\right] \;\; , \;\; \mathbf{C}_S := \left[\begin{array}{cc} \mathcal{C}_S & \mathcal{C}_{12_S} \\ \mathcal{C}_{12_S}^T & \mathcal{C}_2 \end{array}\right]. \tag{3.53}$$

As described in Sect. 4.2.2, the PACT transformation which reduces the network (3.42) up to the selected nodes $\mathbf{x}_S = [\mathbf{x}_{1_S}^T, \mathbf{x}_2^T]^T$ is given by (4.8 - 4.10). Recalling (3.51) and denoting $\mathbf{I}_{S2}$:=blockdiag$(\mathbf{I}_S, \mathbf{I}_2)$, this becomes:

$$\mathbf{V} = \begin{bmatrix} -\mathbf{G}_R^{-1}\mathbf{G}_K \\ \mathbf{I}_{S2} \end{bmatrix} = \left[\begin{array}{cc} -\mathcal{G}_R^{-1}\mathcal{G}_K & \mathbf{0} \\ \hline \mathbf{I}_S & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_2 \end{array}\right] = \mathcal{V}, \tag{3.54}$$

the same as (3.43). The reduced M2 model for (3.42) is thus computed from (4.5 - 4.7) using the assignments (3.51 - 3.53). Performing the computations reveals that the M2 reduced PACT system is:

$$\mathbf{G}_S^{'} = \begin{bmatrix} \widehat{\mathcal{G}}_1 & \mathbf{0} \\ \mathbf{0} & \mathcal{G}_2 \end{bmatrix} = \widehat{\mathcal{G}} \ , \ \mathbf{C}_S^{'} = \begin{bmatrix} \widehat{\mathcal{C}}_1 & \widehat{\mathcal{C}}_{12} \\ \widehat{\mathcal{C}}_{12}^T & \mathcal{C}_2 \end{bmatrix} = \widehat{\mathcal{C}} \tag{3.55}$$

$$\mathbf{B}_S = \begin{bmatrix} \widehat{\mathcal{B}}_1 \\ \mathcal{B}_2 \end{bmatrix} = \widehat{\mathcal{B}} \ , \quad \mathbf{x}_S^{'} = \begin{bmatrix} \widehat{\mathbf{x}} \\ \mathbf{x}_2 \end{bmatrix} = \widehat{\mathbf{x}}, \tag{3.56}$$

where (3.44 - 3.47) hold. This is the same as the reduced model M1 (3.48-3.50).

We have shown the equivalence between reduced models M1 and M2 after the reduction of the first subnet. We proceed for reducing the next candidate, subnet-2 corresponding to the 2nd SCC of $\mathcal{G}$. Recall that $\mathcal{G}_2$ = blockdiag$(\mathcal{G}_{2,2}, \ldots \mathcal{G}_{N,N})$ contains the unreduced SCCs of $\mathcal{G}$. The reduced system (3.48 -3.50) is permuted next, so that the reduced subnetwork-1 moves to the bottom and $\mathcal{G}_{2,2}$ (and the corresponding blocks) is promoted as the next candidate for reduction:

$$\widehat{\mathcal{P}} = \begin{bmatrix} \mathbf{0} & \widehat{\mathbf{I}}_1 \\ \mathbf{I}_2 & \mathbf{0} \end{bmatrix}, \ \widehat{\mathcal{P}}^T\widehat{\mathcal{G}}\widehat{\mathcal{P}}^T = \begin{bmatrix} \mathcal{G}_2 & \mathbf{0} \\ \mathbf{0} & \widehat{\mathcal{G}}_1 \end{bmatrix} := \mathcal{G}, \tag{3.57}$$

$$\widehat{\mathcal{P}}^T\widehat{\mathcal{C}}\widehat{\mathcal{P}}^T = \begin{bmatrix} \mathcal{C}_2 & \widehat{\mathcal{C}}_{12}^T \\ \widehat{\mathcal{C}}_{12} & \widehat{\mathcal{C}}_1 \end{bmatrix} := \mathcal{C}, \ \widehat{\mathcal{P}}^T\widehat{\mathbf{x}} = \begin{bmatrix} \mathbf{x}_2 \\ \widehat{\mathbf{x}}_1 \end{bmatrix} := \mathbf{x}, \ \widehat{\mathcal{P}}^T\widehat{\mathcal{B}} = \begin{bmatrix} \mathcal{B}_2 \\ \widehat{\mathcal{B}}_1 \end{bmatrix} := \mathcal{B}. \tag{3.58}$$

The matrices are redefined for the reduction at step 2 as follows:

$$\mathcal{G} = \left[\begin{array}{c|ccc|c} \mathcal{G}_{2,2} & \mathbf{0} & \ldots & \mathbf{0} & \mathbf{0} \\ \hline \mathbf{0} & \mathcal{G}_{3,3} & \ldots & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \ldots & \mathcal{G}_{N,N} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \widehat{\mathcal{G}}_{1,1} \end{array}\right] := \left[\begin{array}{c|c} \mathcal{G}_1 & \mathbf{0} \\ \hline \mathbf{0} & \mathcal{G}_2 \end{array}\right] \tag{3.59}$$

$$\mathcal{C} = \left[\begin{array}{c|ccc|c} \mathcal{C}_{2,2} & \mathcal{C}_{2,3} & \ldots & \mathcal{C}_{2,N} & \widehat{\mathcal{C}}_{2,1} \\ \hline \mathcal{C}_{2,3}^T & \mathcal{C}_{3,3} & \ldots & \mathcal{C}_{3,N} & \widehat{\mathcal{C}}_{3,1} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathcal{C}_{2,N}^T & \mathcal{C}_{3,N}^T & \ldots & \mathcal{C}_{N,N} & \widehat{\mathcal{C}}_{N,1} \\ \hline \widehat{\mathcal{C}}_{2,1}^T & \widehat{\mathcal{C}}_{3,1}^T & \ldots & \widehat{\mathcal{C}}_{N,1}^T & \widehat{\mathcal{C}}_{1,1} \end{array}\right] := \left[\begin{array}{c|c} \mathcal{C}_1 & \mathcal{C}_{12} \\ \hline \mathcal{C}_{12}^T & \mathcal{C}_2 \end{array}\right], \tag{3.60}$$

where $\mathcal{G}_1, \mathcal{C}_1$ describe the new subnetwork-1 to be reduced, $\mathcal{G}_2, \mathcal{C}_2$ describe the new subnetwork-2, and $\mathcal{C}_{12}$ is the new connectivity block between them.

Step N

At the end of reduction step $N-1$, we have the reduced matrices:

$$\widehat{\mathcal{G}} = \left[\begin{array}{c|cccc} \widehat{\mathcal{G}}_{N-1,N-1} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} \\ \hline \mathbf{0} & \mathcal{G}_{N,N} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \widehat{\mathcal{G}}_{1,1} & \dots & \mathbf{0} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \dots & \widehat{\mathcal{G}}_{N-2,N-2} \end{array}\right] \tag{3.61}$$

$$\widehat{\mathcal{C}} = \left[\begin{array}{c|cccc} \widehat{\mathcal{C}}_{N-1,N-1} & \widehat{\mathcal{C}}_{N-1,N} & \widehat{\mathcal{C}}_{N-1,1} & \dots & \widehat{\mathcal{C}}_{N-1,N-2} \\ \hline \widehat{\mathcal{C}}_{N-1,N}^T & \mathcal{C}_{N,N} & \widehat{\mathcal{C}}_{N,1} & \dots & \widehat{\mathcal{C}}_{N,N-2} \\ \widehat{\mathcal{C}}_{N-1,1}^T & \widehat{\mathcal{C}}_{N,1}^T & \widehat{\mathcal{C}}_{1,1} & \dots & \widehat{\mathcal{C}}_{1,N-2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \widehat{\mathcal{C}}_{N-1,N-2}^T & \widehat{\mathcal{C}}_{N,N-2}^T & \widehat{\mathcal{C}}_{1,N-2}^T & \dots & \widehat{\mathcal{C}}_{N-2,N-2} \end{array}\right], \tag{3.62}$$

With appropriate dimensions, the permutation $\widehat{\mathcal{P}}$ (3.57-3.58) is applied to move the reduced subnetwork-1 (now defined by the reduced $\widehat{\mathcal{G}}_{N-1,N-1}$ and the corresponding matrix blocks from $\widehat{\mathcal{C}}$) to the bottom and promote $\mathcal{G}_{N,N}$ as the next (final) subnetwork for reduction. After the permutation, the active matrices are redefined:

$$\mathcal{G} := \left[\begin{array}{c|ccc} \mathcal{G}_{N,N} & \mathbf{0} & \dots & \mathbf{0} \\ \hline \mathbf{0} & \widehat{\mathcal{G}}_{1,1} & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \widehat{\mathcal{G}}_{N-2,N-2} \end{array}\right] := \left[\begin{array}{c|c} \mathcal{G}_1 & \mathbf{0} \\ \hline \mathbf{0} & \mathcal{G}_2 \end{array}\right] \tag{3.63}$$

$$\mathcal{C} := \left[\begin{array}{c|ccc} \mathcal{C}_{N,N} & \widehat{\mathcal{C}}_{N,1} & \dots & \widehat{\mathcal{C}}_{N,N-2} \\ \hline \widehat{\mathcal{C}}_{N,1}^T & \widehat{\mathcal{C}}_{1,1} & \dots & \widehat{\mathcal{C}}_{1,N-2} \\ \vdots & \vdots & \ddots & \vdots \\ \widehat{\mathcal{C}}_{N,N-2}^T & \widehat{\mathcal{C}}_{1,N-2}^T & \dots & \widehat{\mathcal{C}}_{N-2,N-2} \\ \widehat{\mathcal{C}}_{N,N-1}^T & \widehat{\mathcal{C}}_{1,N-1}^T & \dots & \widehat{\mathcal{C}}_{N-2,N-1}^T \end{array}\right] := \left[\begin{array}{c|c} \mathcal{C}_1 & \mathcal{C}_{12} \\ \hline \mathcal{C}_{12}^T & \mathcal{C}_2 \end{array}\right] \tag{3.64}$$

The procedure of Step 3.3.1 for obtaining the reduced models M1 and M2 is repeated on the new system: (3.63-3.64) with subnetwork-1 defined by $\mathcal{G}_1 := \mathcal{G}_{N,N}$ (the last SCC of $\mathcal{G}$), and subnetwork-2 defined by $\mathcal{G}_2$:=blockdiag($\widehat{\mathcal{G}}_{1,1}, \dots \widehat{\mathcal{G}}_{N-1,N-1}$), each with the corresponding blocks from $\mathcal{C}$. Thus, after N reduction steps the M1 and M2 models remain identical. In particular, after the appropriate permutation $\widehat{\mathcal{P}}$ (3.57-3.58), the final reduced form is:

$$\widehat{\mathcal{G}} = \left[\begin{array}{c|cccc} \widehat{\mathcal{G}}_{1,1} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} \\ \hline \mathbf{0} & \widehat{\mathcal{G}}_{2,2} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \widehat{\mathcal{G}}_{3,3} & \dots & \mathbf{0} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \widehat{\mathcal{G}}_{N,N} \end{array}\right], \quad \widehat{\mathcal{C}} = \left[\begin{array}{c|cccc} \widehat{\mathcal{C}}_{1,1} & \widehat{\mathcal{C}}_{1,2} & \widehat{\mathcal{C}}_{1,3} & \dots & \widehat{\mathcal{C}}_{1,N} \\ \hline \widehat{\mathcal{C}}_{1,2}^T & \widehat{\mathcal{C}}_{2,2} & \widehat{\mathcal{C}}_{2,3} & \dots & \widehat{\mathcal{C}}_{2,N} \\ \widehat{\mathcal{C}}_{1,3}^T & \widehat{\mathcal{C}}_{2,3}^T & \widehat{\mathcal{C}}_{3,3} & \dots & \widehat{\mathcal{C}}_{3,N} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \widehat{\mathcal{C}}_{1,N}^T & \widehat{\mathcal{C}}_{2,N}^T & \widehat{\mathcal{C}}_{3,N}^T & \dots & \widehat{\mathcal{C}}_{N,N} \end{array}\right]. \tag{3.65}$$

■

3.3.2 Moment matching, passivity, terminal connectivity

Theorem 3.3.1 shows that the reduced model obtained based on the SCC($\mathcal{G}$) partitioning is the same as the PACT reduced model derived in Sect. 4.2.2. By Prop. 4.2.1, the reduced model (3.65) thus matches the two multi-port admittance moments at $s = 0$ of the original (3.40). The consequence of Theorem 3.3.1 is that all afore-mentioned properties of the PACT reduced model, such as passivity preservation and preservation of terminal connectivity hold for the partitioned-based reduced model as well.

3.3.3 Revealing path resistances

An important analysis step during the design of interconnect structures is the computation of path resistances between terminals [80]. Here we show how to compute path resistances in a straightforward manner, for RC networks that are reduced with the SCC($\mathcal{G}$)-based partitioning method. First we define the path resistance of a general network. Then, we show that the multi-port RC reduction proposed in this chapter preserves the path resistance. Finally, we show how path resistances are computed from the reduced network.

For a general multi-port resistor network, Kirchhoff's current law gives:

$$\mathcal{G}\mathbf{x} = \mathcal{B}\mathbf{u}, \tag{3.66}$$

where the columns of $\mathcal{B}$ are the unit vectors $\mathcal{B}_{[:,i]} = \mathbf{e}_i$ describing the incidence of current injections into terminals. Within the scope of this section, it is assumed that the conductance matrix $\mathcal{G}$ is invertible, which is the case if: (1) the network is grounded and one of the following holds: (2) the network is completely connected via resistors, in other words the graph defined by the non-zero pattern of $\mathcal{G}$ has one strongly connected component or (3) $\mathcal{G}$ is block diagonal and each block is itself invertible.

The path resistance between two nodes of a circuit is defined as the ratio of voltage across the nodes to the current flow injected into them [28]. The path resistance from terminal i to terminal j is given by:

$$r_{ij} = (\mathbf{e}_i - \mathbf{e}_j)^T \mathcal{G}^{-1} (\mathbf{e}_i - \mathbf{e}_j), \tag{3.67}$$

where $\mathbf{e}_i$, $\mathbf{e}_j$ are the i'th and j'th unit vectors respectively. If neither i nor j is the grounded node, (3.67) becomes:

$$r_{ij} = \mathcal{G}^{-1}_{[i,i]} + \mathcal{G}^{-1}_{[j,j]} - 2\mathcal{G}^{-1}_{[i,j]}, \tag{3.68}$$

where $\mathcal{G}_{[i,j]}$ is the entry in the i'th row and j'th column of $\mathcal{G}$ [not to be mistaken for $\mathcal{G}_{i,i}$ denoting the i'th strongly connected component of $\mathcal{G}$ as in (3.39)]. If j is the grounded

terminal, thus $e_j = \mathbf{0}$, then (3.67) is simply:

$$r_{ii} = \mathcal{G}^{-1}_{[i,i]}. \tag{3.69}$$

A more compact form entering the computation of (3.67), (3.68) and (3.69) is the matrix of resistive paths:

$$\mathcal{R} \coloneqq \mathcal{B}^T \mathcal{G}^{-1} \mathcal{B}. \tag{3.70}$$

As explained in [80], since $\mathcal{G}$ is positive semi-definite, in practice the computation of (3.70) is based on the Cholesky factorization [29] of $\mathcal{G}$: $\mathcal{G} = \mathbf{L}\mathbf{L}^T$, $\mathbf{Q} = \mathbf{L}^{-1}\mathcal{B}$, $\mathcal{R} = \mathbf{Q}^T\mathbf{Q}$. For the sake of capturing the contribution of all terminals to the resistive path computation, further-on we shall denote the matrix quantity (3.70) as the *path resistance* rather than (3.67).

Let (3.66) be partitioned according to voltages measured at terminals $\mathbf{x}_P$ and voltages measured at the internal nodes $\mathbf{x}_I$:

$$\begin{bmatrix} \mathcal{G}_I & \mathcal{G}_C \\ \mathcal{G}_C^T & \mathcal{G}_P \end{bmatrix} \begin{bmatrix} \mathbf{x}_I \\ \mathbf{x}_P \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathcal{B}_P \end{bmatrix}, \tag{3.71}$$

where $\mathcal{B}_P = \mathbf{I}$, the identity block corresponding to current injections into each of the $\mathbf{x}_P$ terminal nodes. Then:

$$\mathcal{G}^{-1} = \begin{bmatrix} \mathbf{I} & -\mathcal{G}_I^{-1}\mathcal{G}_C \\ \mathbf{0} & \mathbf{I} \end{bmatrix}^T \begin{bmatrix} \mathcal{G}_I^{-1} & \mathbf{0} \\ \mathbf{0} & (\mathcal{G}_P - \mathcal{G}_C\mathcal{G}_I^{-1}\mathcal{G}_C)^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{I} & -\mathcal{G}_I^{-1}\mathcal{G}_C \\ \mathbf{0} & \mathbf{I} \end{bmatrix}. \tag{3.72}$$

Plugging (3.72), into (3.70) gives:

$$\mathcal{R} = \mathcal{B}^T\mathcal{G}^{-1}\mathcal{B} = \mathcal{B}_P^T(\mathcal{G}_P - \mathcal{G}_C^T\mathcal{G}_I^{-1}\mathcal{G}_C)^{-1}\mathcal{B}_P^T = (\mathcal{G}_P - \mathcal{G}_C^T\mathcal{G}_I^{-1}\mathcal{G}_C)^{-1}. \tag{3.73}$$

Notice that (3.73) is actually the inverse of the first moment at $s = 0$ of the multi-port admittance as shown in the proof of Prop. 4.2.1, and also the inverse of the reduced conductance matrix of the PACT reduced model. This demonstrates that *PACT reduction preserves the original path resistances between terminals*. Theorem (3.3.1) showed that the PACT reduced model for an unpartitioned network is equivalent to the reduced model obtained via the SCC($\mathcal{G}$)-based partitioning, and as a consequence this will also preserve the path resistances.

Recall however that path resistances are only computable if the inverse of $\mathcal{G}$ exists, which does not generally hold for *RC* circuits, due to the fact that not all nodes are connected to each other via resistors. If this is the case, then the conductance matrix $\mathcal{G}$ has more strongly connected components. The SCC($\mathcal{G}$)-based partitioning and reduction therefore provides an additional structural advantage: *the path resistances between*

terminal nodes in the circuit become readily computable per component. The immediate consequence of partitioning the network according to the strongly connected components of $\mathcal{G}$ is that subnets are identified that are not connected to each-other via resistors. In matrix terms this corresponds to the block diagonal structure of (3.39). In other words there is no resistive path from nodes in one component to nodes in another component (there may be capacitive connections, but these do not enter the computation of path resistances). As reduction progressed per component, the reduced model will retain the SCC($\widehat{\mathcal{G}}$) on the block diagonal, as seen from (3.65).

The path resistances for an RC network (3.39) where $\mathcal{G}$ has N strongly connected components can thus be computed per component (after appropriate grounding) as follows:

$$\begin{aligned} \mathcal{R}_i &= \mathcal{B}_i^T \mathcal{G}_{i,i}^{-1} \mathcal{B}_i^T = && (3.74) \\ &= \widehat{\mathcal{G}}_{i,i}^{-1}, \ i = 1 \dots N. && (3.75) \end{aligned}$$

The last equality in (3.74) follows from (3.73) where $\mathcal{G}_{i,i}$, $\mathcal{B}_i$ are split as in (3.71), and $\widehat{\mathcal{G}}_{i,i} = \mathcal{G}_{i_p} - \mathcal{G}_{i_C}^T \mathcal{G}_{i_I}^{-1} \mathcal{G}_{i_C}$ are the diagonal blocks from the reduced model (3.65). The computation of path resistances is performed for the example of Sect. 3.4.3.

3.3.4 Computational complexity

With the SCC($\mathcal{G}$)-based reduction being mathematically equivalent to PACT, there remains to quantify the computational advantage of the former compared to the latter. Consider the original model before the first reduction step (3.41), and the corresponding reducing projection (3.43), and note that only the submatrices of $\mathcal{G}_1$ enter in forming the projection. This holds during the reduction of each component, whose computational cost is determined by forming $\mathcal{G}_R^{-1}\mathcal{G}_S$.

Assuming that each component $i = 1 \dots N$, has m_i internal nodes and p_i terminals, the corresponding sub-block dimensions are $\mathcal{G}_R \in \mathbb{R}^{(m_i \times m_i)}$, $\mathcal{G}_S \in \mathbb{R}^{(p_i \times p_i)}$. The reduction cost per component is thus $\mathrm{O}(m_i^{\alpha_i} p_i)$, where typically $1 < \alpha_i \leq 2$ for circuit matrices [75]. For a total of N components, the total cost is $O(\sum_{i=1}^{N} m_i^{\alpha_i} p_i)$. For an unpartitioned circuit with n internal nodes and p terminals, the cost of PACT is $\mathrm{O}(m^{\alpha} p)$, but after the SCC($\mathcal{G}$) partitioning one attains $m_i < m$ and $p_i < p$. Therefore especially when m and p are large, the cost for the component-wise reduction will be smaller than the $\mathrm{O}(m^{\alpha} p)$ of PACT. The large examples from Sect. 3.4.4 confirm this in practice.

3.3.5 Improving sparsity per subnetwork

The partitioning according to the SCCs of $\mathcal{G}$ provides structural and computational advantages, as smaller subnets are reduced individually, with the appropriate reduction

of the capacitive blocks connecting the subnets. This block-wise reduction alone however does not directly minimize fill-in: circuit partitioning according to the SCCs($\mathcal{G}$) only identifies sub-networks to be reduced individually, but has no immediate effect on the sparsity of the reduced model. As seen from Theorem 3.3.1, the PACT reduction per subnet is mathematically equivalent to applying PACT (see Sect. 4.2.2) directly on the unpartitioned network. Therefore reduced models obtained with the SCC($\mathcal{G}$)-based partitioning will have the same fill-in as otherwise obtained from applying PACT without partitioning. Nevertheless, overall sparsity can be improved in this framework as well by applying fill-reducing orderings per subnet, as described next.

Reconsider the partitioning (3.35), where subnet-1 is to be reduced. Applying a *constrained approximate minimum degree reordering (CAMD)* [1,2] on the graph $G1 := \mathrm{nzp}(\mathcal{G}_1 + \mathcal{C}_1)$ will reorder the nodes $\mathbf{x}_1$ so that fill-in is minimized during the PACT reduction of subnet-1 [where *nzp* stands for the "non-zero pattern" of a matrix]. In particular, the special internal nodes of subnet-1 are identified, which if eliminated would introduce the most fill-in in $\widehat{\mathcal{G}}_1$ and $\widehat{\mathcal{C}}_1$. Consequently, these internal nodes are preserved and promoted along with the terminals to form the selected nodes $\mathbf{x}_{1_S}$ of subnet-1. The PACT reduction per subnet-1 follows as in Sect. 3.3.1. The effect of the CAMD reordering per subnet is demonstrated in Example 3.4.2.

As the CAMD reordering is applied only to the individual graphs $G_i \coloneqq \mathrm{nzp}(\mathcal{G}_{ii} + \mathcal{C}_{ii})$ of each subnet, the corresponding ordering of the capacitive connection blocks $\mathcal{C}_{ij}$, $i \neq j$ between the subnets however is not minimizing their fill-in. Consequently the reduced $\widehat{\mathcal{C}}_{ij}$ blocks would be too dense if the $\mathcal{C}_{ij}$ blocks are large. Especially for circuits with $p > 100$, sparsity is better improved by considering the graph G associated with the entire RC topology (i.e. by considering all R and the C connections from the start). While the permutation according to the strongly connected components of $G \coloneqq \mathrm{nzp}(\mathcal{G})$ revealed a partitioning with $\mathcal{G}$ in block diagonal form (with no R communication between blocks), the extension to the general case is the BBD-based partitioning of $G \coloneqq \mathrm{nzp}(\mathcal{G} + \mathcal{C})$ from chapter 4. The methodology therein achieves better sparsity levels and is thus recommended for reducing very large circuits with terminals exceeding thousands.

3.4 Numerical results

A selection of circuits from the electronics industry was reduced with the SCC-based partitioning proposed here. The examples shown have relatively few terminals $p = O(10^2)$, thus suitable for this framework, as sparsity preservation is not of major concern. Rather, the examples demonstrate the performance of the SCC-based reduction strategy in terms of approximation quality, preservation of terminal connectivity, synthesis and re-simulation. As in chapter 4, the reduced netlists are obtained by unstamping the reduced model (3.65) with RLCSYN [93]. In the following tables, several parameters are recorded: p-number of terminals, $n(k)$-number of internal nodes in the original (reduced) circuit respectively, $P_{nk} = \frac{100(n-k)}{n}$-the percentage reduction in internal nodes

,#R-number of resistors, $P_R = \frac{100(\#R_{orig} - \#R_{red})}{\#R_{orig}}$-the percentage reduction/increase in the number of resistors, #C-number of capacitors, $P_R = \frac{100(\#C_{orig} - \#C_{red})}{\#C_{orig}}$-the percentage reduction/increase in the number of capacitors, CPUt- CPU time simulation time for the original and reduced circuits, Speed-up $= \frac{CPUt_{orig}}{CPUt_{red}}$. It should be noted that the recorded #C quantities for the reduced circuits also include possibly generated negative capacitance values. For reasons explained in Sect. 4.3.3, these are kept in the reduced netlist.

3.4.1 Low Noise Amplifier (LNA) circuit (CMOS045)

Three RC parasitic extracted models ($TL_{1,2,3}$) of the LNA circuit are reduced according to Sect. 3.3.1. Belonging to the same family, the emphasis is placed here on the first, TL_1. The original $\mathcal{G}$ and $\mathcal{C}$ matrices are shown in Fig. 3.1, while in Fig. 3.2 they are permuted and partitioned according to the SCCs($\mathcal{G}$) [24 components]. These subnetworks are visible in the diagonal blocks of $\widehat{\mathcal{G}}$. The reduced $\widehat{\mathcal{G}}$ and $\widehat{\mathcal{C}}$ retain this structure (see Fig. 3.3). The reduced block diagonals of $\widehat{\mathcal{G}}$ (and correspondingly of $\widehat{\mathcal{C}}$) are entirely associated with the preserved terminal nodes from each subnet (no internal nodes are preserved).

Table 3.1 collects the reduction and re-simulations results of the 3 circuits. As no fill-in minimizing node reorderings were used, all nodes were eliminated except the terminals (100% reduction rate in internal nodes). From the element reduction rates P_R and P_C, it is clear that the reduced circuits contain much fewer circuit elements than the original. The benefit of reduction is immediately reflected in the tremendous speed-ups attained, when the reduced circuits were re-simulated compared to the original simulations. The reduction time itself is very small (below 1s), and the re-simulation time for the reduced circuits is also more than 100 times smaller than the original simulations. Even more, simulating the original TL_3 circuit was only possible after reduction (the original simulation failed in finding a DC solution, possibly due to the ill-conditioning of the underlying matrices).

The simulations consisted of the following analysis: AC, Noise, SP (S-parameter), PSS (periodic steady state). Figures 3.4-3.5 compare, for several simulation types, the responses of the original vs. the reduced circuit. These match perfectly up to very high frequencies.

Figures 3.6-3.7 show the waveforms obtained from simulating circuit TL_3 after reduction. Even though the original simulation failed due to a convergence error in computing the DC solution, the reduced simulation ran without errors. Without a reference solution to compare the reduced simulation against, these responses were nevertheless appreciated by the circuit designers to follow the desired behavior. Furthermore, the accuracy of the results obtained from reducing TL_1 and TL_2 (of the same family as TL_3) predicts desirable performance from reducing TL_3.

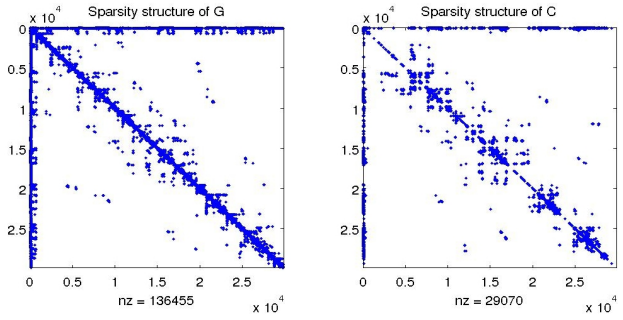


Figure 3.1: *Example 3.4.1-TL_1: original, unordered $\mathcal{G}$ (left) and $\mathcal{C}$ (right) matrices.*

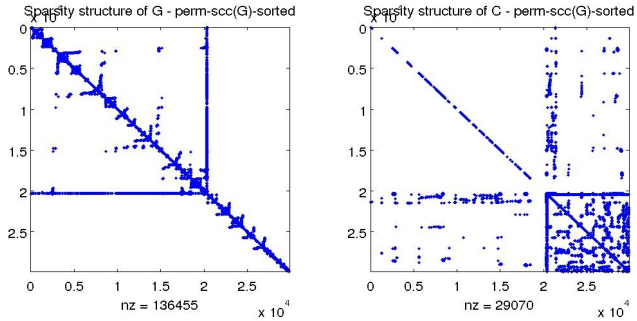


Figure 3.2: *Example 3.4.1-TL_1: $\mathcal{G}$ (left) and $\mathcal{C}$ (right) reordered according to SCCs($\mathcal{G}$).*

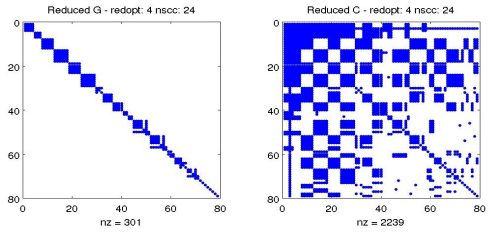


Figure 3.3: *Example 3.4.1-TL_1: reduced $\widehat{\mathcal{G}}$ (left) and $\widehat{\mathcal{C}}$ (right). Note the preserved block diagonal structure of $\widehat{\mathcal{G}}$, showing each reduced strongly connected component.*

3.4.2 Mixer circuit

The mixer circuit (layout in Fig. 3.8) example shows how using fill-in minimizing node reorderings (CAMD) on each subnetwork can improve the sparsity of the reduced model, by identifying special internal nodes which are not eliminated. Table 3.2 collects the reduction results for the two strategies: method 3.3.5, where CAMD reorders the nodes in each subnet for minimum fill-in, and method 3.3.1 (without reorderings). The reduced models were 2 times faster to simulate than the original.

Table 3.1: *Reduction summary for Example 3.4.1*

	TL_1		TL_2		TL_3	
	Orig.	Red.	Orig.	Red.	Orig.	Red.
p	79		75		79	
$n \mid k$	29806	0	33818	0	27962	0
P_{nk}	100%		100%		100%	
#R	70338	117	81843	99	66068	117
P_R	99.8%		99.8%		99.8%	
#C	12038	1047	12145	920	9786	1032
P_C	91.3%		92.4%		89.5%	
Reduction time	0.44 s		0.39s		0.46 s	
Sim.	CPUt					
AC	65.83 s	0.12 s	60.11	0.17 s	NA[4]	0.12 s
Noise	59.68 s	0.13 s	54.82	0.18 s	NA	0.14 s
SP	82.81 s	0.21 s	110.7	0.19 s	NA	0.22 s
PSS	793.13 s	2.93 s	424.28	3.33 s	NA	3.05 s
Speed-up	> 270x		> 127x		∞	

Table 3.2: *Reduction summary for Example 3.4.2*

	Orig.	Red. 3.3.5	Perc.	Red. 3.3.1	Perc.
p	110	110		110	
$n \mid k$	757	13	98.2%	0	100%
#R	1393	110	92.8%	111	92.0%
#C	2353	872	62.9%	1117	52.5%
Sim.	CPUt		Speed Up	CPUt	Speed Up
QPSS	23.48 s	10.84 s	$\sim$2x	13.2 s	$\sim$2x
QPSP	1735 s	754 s	$\sim$2x	746 s	$\sim$2x

Fig. 3.9 shows how the minimum fill-in is monitored during one subnet reduction (here, subnet-11): the x-axis shows the number of the node to be eliminated, after the re-ordering of nodes based on CAMD($\mathcal{G}_{11} + \mathcal{C}_{11}$) ; the y-axis shows the $\#R + \#C$ as node elimination progresses. The red circle shows the minimum value of circuit elements $\min(\#R + \#C)$, here attained after eliminating node 8. Further eliminating node 9 would significantly increase the $\#R + \#C$ in the reduced subnet-11. Thus node 9 of subnet-11 is not eliminated. Tracking the fill-in per subnet in this manner has identified the total $k = 13$ internal nodes preserved in the final reduced model 3.3.5 (see Table 3.2, first column). The improvement in sparsity is reflected in the fact that reduced model Red. 3.3.5 has fewer circuit elements than model Red. 3.3.1.

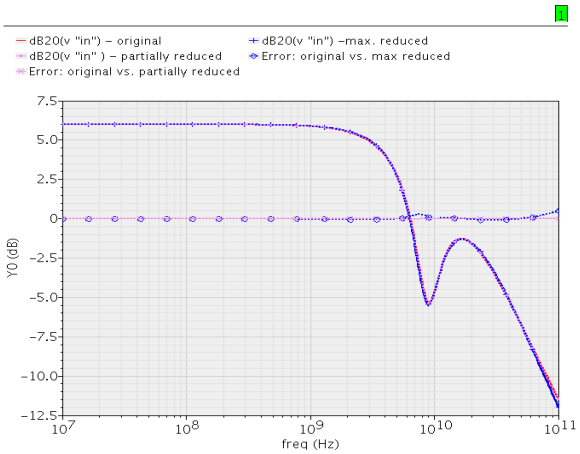


Figure 3.4: *TL_1: AC analysis - node "in": magnitude. Comparison: original vs. reduced, together with the approximation error which is very small (difference between "original" and "reduced" curves is around 0dB).*

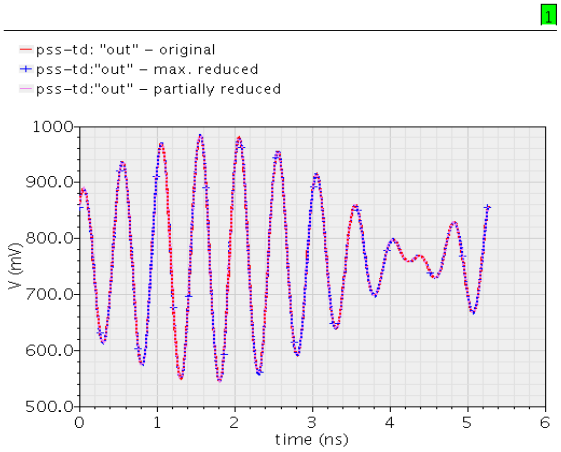


Figure 3.5: *TL_1: PSS analysis, time domain - node "out". Comparison: original vs. reduced.*

3.4.3 Interconnect structure

The *RC* extracted parasitics of four interconnect structures, each with $p = 12$ terminals, were reduced with the proposed strategy, with results recorded in Table 3.3. The reduction rates for nodes and circuit elements are above 95%, and the reduction time is in the order of 1s. Two types of analysis are required, an AC simulation and the computation of path resistances R_{path} between terminals. The AC analysis is used to compare the response of the original netlist to the reduced netlist for a large frequency range. As seen from Fig. 3.10, these match perfectly.

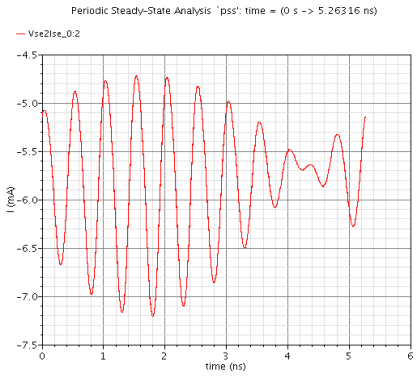


Figure 3.6: *TL_3: PSS: node "Vse2lse_0:2".*

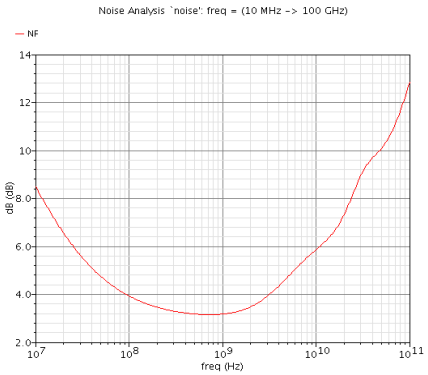


Figure 3.7: *TL_3: Noise: NF.*

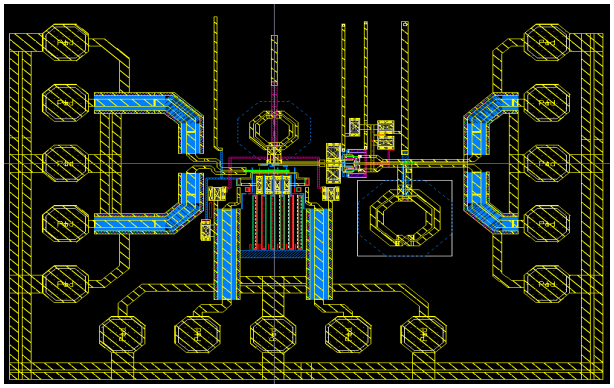


Figure 3.8: *Example 3.4.2: mixer circuit layout.*

Table 3.3: *Reduction summary for Example 3.4.3*

	ICL_1		ICL_2		ICL_3		ICL_4	
	Orig.	Red.	Orig.	Red.	Orig.	Red.	Orig.	Red.
p	12		12		12		12	
$n \mid k$	2706	1	547	1	67103	1	62368	1
P_{nk}	99.9%		99.9%		99.9%		99.9%	
#R	3143	6	984	6	97573	6	92838	6
P_R	99.8%		99.3%		99.9%		99.9%	
#C	1896	31	1229	31	223974	31	113887	31
P_C	98.3%		97.5%		99.9%		99.9%	
Reduction time	0.06 s		0.05 s		1.2 s		0.8 s	
Numerical solution	CPU time							
R_{path}	0.009 s	0 s	0.009 s	0 s	0.19 s	0 s	0.16 s	0 s

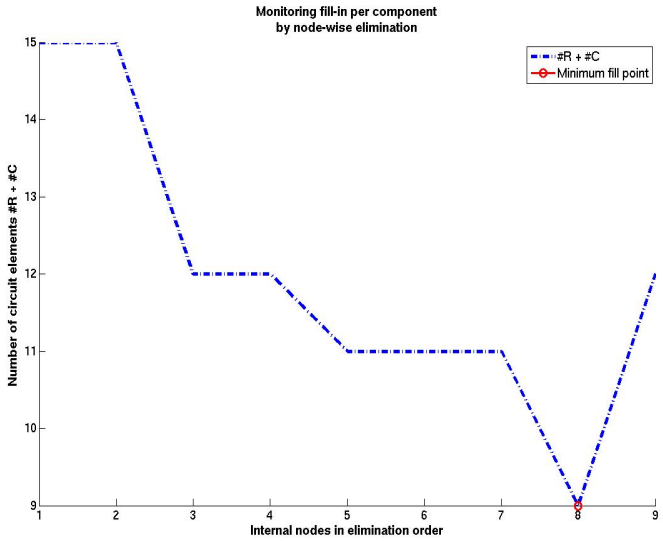


Figure 3.9: *Example 3.4.2. Monitoring the fill-in generated during one subnet reduction. The minimum value of #R + #C is attained after eliminating node 8, while node 9 is preserved.*

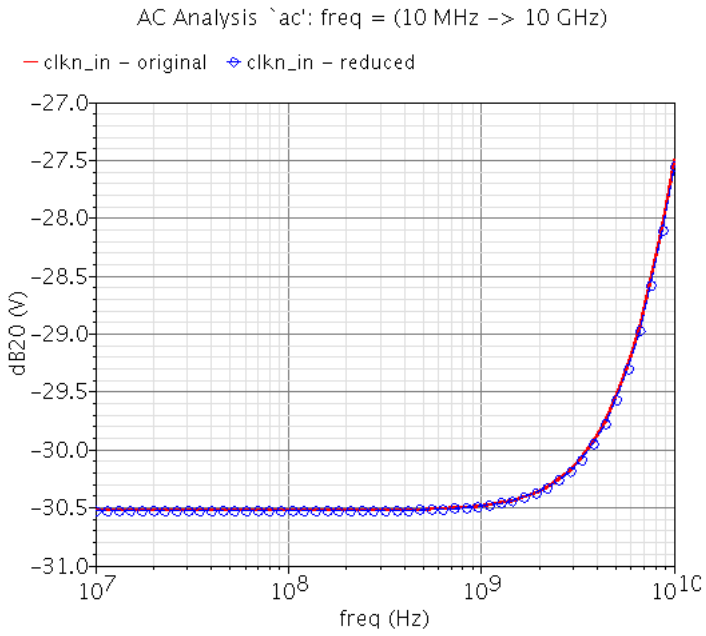


Figure 3.10: *Example 3.4.3. AC analysis for original and reduced netlists ICL1 showing a perfect match.*

While an AC analysis is an operation performed with the circuit simulator (here, Spectre [16]), the computation of path resistances is sometimes not a direct functionality of

commercial tools. This however is easily performed numerically via the partitioned conductance matrix, as described in Sect. 3.3.3. For ICL_3, the original matrices are shown in Fig. 3.11. The $SCC(\mathcal{G})$ partitioning reveals six components, each with only two terminals. As seen in Fig. 3.12, the reduced model preserves the six components, each component having only two nodes (i.e. the terminal nodes) as expected. The path resistance between each pair of terminals is computed using formula (3.74) for each component, after appropriate grounding. For this example actually, these path resistances can be simply read-off from the upper diagonal of $\widehat{\mathcal{G}}$.

Note in Table 3.3 that the time for computing the path resistances for the original net via (3.74) is very small, and close to the time required to compute them for the reduced net via (3.75). This is due to the fact that the inverses in (3.74), (3.75) are formed via the backslash operator (`mldivide`) in Matlab [which itself exploits the sparsity of $\mathcal{G}$ and performs a sparse $\mathbf{L}\mathbf{L}^T$ factorization based on AMD reordering [67]] hence turns out to be very efficient, especially as the number of columns of $\mathcal{B}$ is small (only 12 terminals). As has been already emphasized in [80], DC problems such as the computation of path resistances can often be solved efficiently as solutions to sparse linear systems involving the $\mathcal{G}$ matrix directly (and a network reduction is unnecessary). In contrast, simulations requiring a sweep over a large frequency range (such as an AC analysis), require that solutions to linear systems involving a denser problem $(\mathcal{G} + s_i\mathcal{C})^{-1}\mathcal{B}$ are computed for each frequency point s_i. These repeated operations are inefficient especially when the number of terminals exceeds thousands, hence the need for reduced networks.

3.4.4 Very large networks

Two very large networks [part of a phase locked loop (*PLL*) and a receiver design (*RX*) respectively] with more than 10^5 internal nodes and 10^3 terminals were reduced based on the SCC($\mathcal{G}$) partitioning, with results recorded in Table 3.4. Due to their large dimension, the direct PACT reduction fails due to insufficient computational resources in forming (4.5)-(4.7). The block-wise reduction of subnets based on the SCC($\mathcal{G}$) partitioning in contrast was able to reduce these networks within an order of minutes, which is explained by the analysis in Sect. 3.3.4. Reduction rates of $\approx 90\%$ in internal nodes, resistors and capacitors were thus achieved.

3.5 Concluding remarks

The reduction of multi-terminal *R* and *RC* networks which share a common projection matrix was addressed in this chapter. Based on the properties of the conductance matrix **G** underlying these circuits, and the special structure of the projection matrix, it was shown that the reduced models obtained in this framework have only positive resistors. A partition-based implementation for multi-terminal *RC* reduction was also derived,

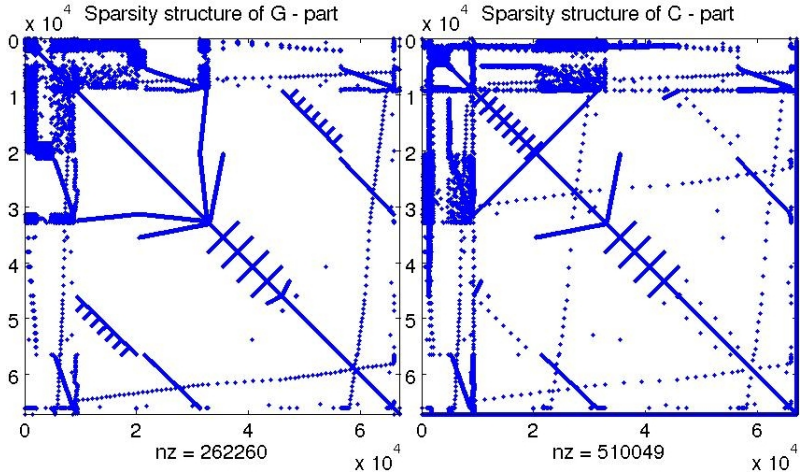


Figure 3.11: *Example 3.4.3. Original $\mathcal{G}$ (left) and $\mathcal{C}$ (right) matrices.*

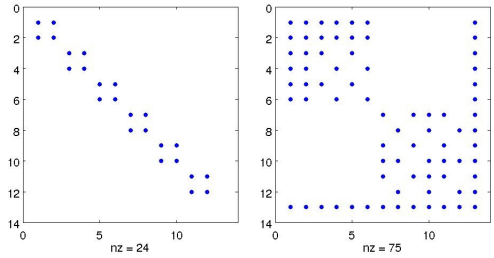


Figure 3.12: *Example 3.4.3. Reduced $\widehat{\mathcal{G}}$ (left) and $\widehat{\mathcal{C}}$ (right) matrices. The strongly connected components of $\widehat{\mathcal{G}}$ appear as blocks on the diagonal. These are directly used to compute the path resistances (3.74).*

Table 3.4: *Reduction summary for Example 3.4.4*

	PLL		*Receiver*	
	Orig.	Red.	Orig.	Red.
p	1134		1794	
$n \mid k$	380340	205	801458	818
P_{nk}	99.9%		99.9%	
#R	593786	7458	1416454	11727
P_R	99.8%		99.1%	
#C	555553	59463	1961224	129978
P_C	89%		93.3%	
Reduction time SCC($\mathcal{G}$)-based	42.4 s		760s	
Reduction: PACT	NA		NA	

based on the strongly connected components of the conductance matrix $\mathcal{G}$. A proof on the mathematical equivalence between the unpartitioned versus partitioned approach was provided. This ensures that the two approaches share the same moment matching, passivity preservation and terminal reconnectivity properties. The partition provides additional structural advantages, such as a simple solution for computing path resistances in a network component-wise. Numerical experiments show that the reduced circuits thus obtained contain much fewer elements than the original and attain significant speed-ups in re-simulation. The methods in this chapter inspired the development of the sparsity preserving *RC* reduction methodology of Chapter 4, especially suited for circuits with terminals exceeding thousands.

Chapter 4

SparseRC: Sparsity preserving model reduction for multi-terminal RC networks

A novel model order reduction (MOR) method for multi-terminal *RC* circuits is proposed: SparseRC. Specifically tailored to systems with many terminals, SparseRC employs graph-partitioning and fill-in reducing orderings to improve sparsity during model reduction, while maintaining accuracy via moment matching. The reduced models are easily converted to their circuit representation. These contain much fewer nodes and circuit elements than otherwise obtained with conventional MOR techniques, allowing faster simulations at little accuracy loss.

4.1 Introduction

During the design and verification phase of VLSI circuits, coupling effects between various components on a chip have to be analyzed. This requires simulation of electrical circuits consisting of many non-linear devices together with extracted *parasitics*. Due to the increasing amount of parasitics, full device-parasitic simulations are too costly and often impossible. Hence, reduced models are sought for the parasitics, which when re-coupled to the devices can reproduce the original circuit behavior.

Parasitic circuits are very large network models containing millions of nodes interconnected via basic circuit elements: R, RC or $RLC(k)$. Of the circuit nodes, a special subset form the *terminals*, which are the designer specified input/output nodes and the nodes connecting the parasitics to the non-linear devices. Parasitic networks with millions

of nodes, *RC* elements, and thousands of terminals are often encountered in real chip designs. A *reduced order model* for the parasitics ideally has fewer nodes and circuit elements than the original, and preserves the terminal nodes for re-connectivity. The presence of many terminals introduces additional structural and computational challenges during *model order reduction (MOR)*. Existing MOR methods may be unsuitable for circuits with many terminals as they produce *dense* reduced models. These correspond to circuits with fewer circuit nodes, but more circuit elements (*R*s, *C*s) than the original circuit, and may even require longer simulation times than originally. Furthermore, if terminal connectivity is affected, additional elements such as current/voltage controlled sources must be introduced to appropriately model the re-connection of reduced parasitics to other devices.

The emerging problem is to develop *efficient* model reduction schemes for large multi-terminal circuits that are *accurate*, *sparsity preserving* and also *preserve terminal connectivity*. The method proposed here, SparseRC, achieves these goals by efficiently combining the strengths of existing MOR methodology with graph-partitioning and fill-reducing node reordering strategies, achieving tremendous reduction rates even for circuits with terminal numbers exceeding thousands. Reduced *RC* models thus obtained are sparser than those computed via conventional techniques, have shorter simulation timings, and also accurately approximate the input/output behavior of the original *RC* circuit. In addition, the reduced *RC* parasitics can be reconnected directly via the terminal nodes to remaining circuitry (e.g. non-linear devices), without introducing new circuit elements.

A comprehensive coverage of established MOR methods is available in [4], while [82] collects more circuit simulation specific contributions. Mainly, MOR methods are classified into truncation-based (modal approximation [79]/balancing [77]) and Krylov-based methods, from which we mention PRIMA [71], SPRIM [27], or the dominant spectral zero method [44] as they are passivity preserving[1]. Generally however, the applicability of traditional MOR techniques to very large circuits with many terminals is limited due to computational limitations together with the afore-mentioned sparsity and re-connectivity considerations. While the multi-terminal problem has been addressed in numerous works such as [24], [13] [95], [59], it is usually less clear whether their performance scales with the number of ports, especially as this exceeds thousands.

Recent developments in model reduction for very large multi-terminal *R*-networks were achieved in [80] (denoted here as *ReduceR*), which uses graph theoretical tools, fill-in minimizing node reorderings and node elimination to obtain sparse reduced *R*-networks. Towards obtaining sparse reduced models for multi-terminal $RC(L)$ networks, the *Sparse implicit projection (SIP)* method [94] also proposes reordering actions prior to eliminating unimportant internal nodes, and makes several important analogies between related node elimination-based methods (e.g. TICER [84], and [22]) and moment-matching MOR by projection (e.g. PRIMA [71]). In fact, the fundamental projection behind SIP can be traced back in the PACT methods [56,57] for reducing multi-terminal $RC(L)$ networks.

[1] Only passive reduced order models guarantee stable results when re-coupled to other circuit blocks in subsequent simulation stages [71].

As will be shown, SparseRC combines the advantages of ReduceR and SIP/PACT into an efficient procedure, while overcoming some of their computational limitations: using graph partitioning, circuit components are identified which are reduced individually via a PACT-like projection [denoted here as the *extended moment matching projection, (EMMP)*] while appropriately accounting for the interconnection between components. The reduction process is simplified computationally, as smaller components are reduced individually. The final SparseRC reduced circuit matches by default two moments at DC of the original circuit's multi-port admittance, and can be extended with dominant poles [79] or additional moments to improve accuracy at higher frequency points if needed. Through partitioning, the relevant nodes responsible for fill-in are identified *automatically*; SparseRC preserves these along with the terminals to ensure the sparsity of the reduced model. This feature makes SparseRC more efficient than ReduceR or SIP: it avoids the unnecessary computational cost of monitoring fill-in at each node elimination step.

A related method is PartMOR [70], which is based on the same partitioning philosophy, but constructs the reduced models in a different manner. PartMOR realizes selected moments from each subnet into a netlist equivalent, while SparseRC is implemented as block moment matching projection operating on the matrix hierarchy which results from partitioning. This construction enables SparseRC to match admittance moments per subnet as well as for the recombined network. With global accuracy thus ensured, the approximation quality of the final SparseRC reduced model is guaranteed irrespective of the partitioning tool used or the number/sizes of partitions.

This work focuses on *RC* reduction. For *RC*, a reducing transformation which matches multi-port admittance moments is sufficient to ensure accuracy *and* can be so constructed as to improve sparsity. For *RLC* however, additional accuracy considerations have to be accounted for as to capture oscillatory behavior. Hence, constructing a projection which simultaneously ensures accuracy and sparsity is more involved. *RLC* circuits can be partitioned with the same framework using a second order susceptance based system formulation which reveals the network topology [93]. On the other hand, the dense nature of inductive couplings for *RLCK* circuits may prevent finding a good partition. These topics are treated in Chapter 5; for an existing *RLC* partitioned-based approach we refer to PartMOR [70].

The rest of the chapter is structured as follows. Sect. 4.2 formulates the multi-terminal model reduction problem. The SparseRC partitioning based strategy is detailed in Sect. 4.3, the main focus of the chapter. Numerical results and circuits simulations are presented in Sect. 4.4. Sect. 4.5 concludes. Some conventions on notation and terminology follow next. **Matrices**: $\mathbf{G}$ and $\mathcal{G}$ are used interchangeably for the conductance matrix, depending on whether the context refers to unpartitioned or partitioned matrices respectively (similarly for the capacitance matrix $\mathbf{C}$, $\mathcal{C}$ or the incidence matrix $\mathbf{B}$, $\mathcal{B}$). **Graphs**: G (non-bold, non-calligraphic) is a graph associated with the non-zero pattern of the circuit matrices, C (non-bold, non-calligraphic) is a component of G, nzp is the non-zero-pattern of a matrix, i.e. its graph topology. **Dimensions**: p-number of circuit terminals (external nodes), the same for the original and reduced matrices/circuit, n-

the number of internal nodes of the original circuit, k-the number of internal nodes of the reduced circuit, N-number of matrix partitions. **Nodes**: circuit nodes prior to partitioning are classified into *terminals*, and *internal nodes*; *separator nodes* are a subset of the original nodes identified through partitioning as communication nodes among individual components. **Terminology**: a *partition/subnet/block* describes the same concept: an individual graph/circuit/matrix component identified from partitioning; similarly, a *separator, border* is a component containing only separator nodes.

4.2 Problem formulation

This section provides the preliminaries for model reduction of general *RC* circuits and identifies the challenges emerging in multi-terminal model reduction. The building block for SparseRC is described: EMMP, an extended moment matching projection for reducing multi-terminal *RC* circuits.

4.2.1 Model reduction

Similarly to [56], consider the modified nodal analysis (MNA) description of an *RC* circuit:

$$(\mathbf{G} + s\mathbf{C})\mathbf{x}(s) = \mathbf{B}\mathbf{u}(s), \tag{4.1}$$

where MNA matrices $\mathbf{G}$, $\mathbf{C}$ are symmetric, non-negative definite, corresponding to the stamps of resistor and capacitor values respectively. $\mathbf{x} \in \mathbb{R}^{n+p}$ denote the node voltages (measured at the n internal nodes and the p terminals) and $n + p$ is the dimension of (6.1). $\mathbf{u} \in \mathbb{R}^p$ are the currents injected into the terminals. The outputs are the voltage drops at the terminal nodes: $\mathbf{y}(s) = \mathbf{B}^T\mathbf{x}(s)$. The underlying matrix dimensions are: $\mathbf{G}, \mathbf{C} \in \mathbb{R}^{(n+p)\times(n+p)}, \mathbf{B} \in \mathbb{R}^{(n+p)\times p}$. In model reduction, an appropriate $\mathbf{V} \in \mathbb{R}^{(n+p)\times(k+p)}, k \geq 0$ is sought, such that the system matrices and unknowns are reduced to:

$$\begin{aligned} \widehat{\mathbf{G}} &= \mathbf{V}^T\mathbf{G}\mathbf{V},\ \widehat{\mathbf{C}} = \mathbf{V}^T\mathbf{C}\mathbf{V} \ \in \mathbb{R}^{(k+p)\times(k+p)} \\ \widehat{\mathbf{B}} &= \mathbf{V}^T\mathbf{B} \in \mathbb{R}^{(k+p)\times p},\ \widehat{\mathbf{x}} = \mathbf{V}^T\mathbf{x} \in \mathbb{R}^{k+p} \end{aligned}$$

and satisfy: $(\widehat{\mathbf{G}} + s\widehat{\mathbf{C}})\widehat{\mathbf{x}}(s) = \widehat{\mathbf{B}}\mathbf{u}(s)$.

The transfer function $\mathbf{H}(s) = \mathbf{B}^T(\mathbf{G} + s\mathbf{C})^{-1}\mathbf{B}$ characterizes the system's behavior at the input/output ports (here, at the terminal nodes) over the frequency sweep s. After reduction, this becomes: $\widehat{\mathbf{H}}(s) = \widehat{\mathbf{B}}^T(\widehat{\mathbf{G}} + s\widehat{\mathbf{C}})^{-1}\widehat{\mathbf{B}}$. A "good" reduced model/circuit generally satisfies the following:

(a) gives a small approximation error $\|\mathbf{H} - \widehat{\mathbf{H}}\|$ in a suitably chosen norm, for instance by ensuring that $\widehat{\mathbf{H}}$ matches moments of the original $\mathbf{H}$ at selected frequency points,

(b) preserves passivity (and stability implicitly) and,

(c) can be computed efficiently.

For multi-terminal circuits especially, new conditions emerge:

(d) for reconnectivity purposes, the incidence of current injections into terminal nodes is preserved (i.e. $\widehat{\mathbf{B}}$ is a submatrix of $\mathbf{B}$) and

(e) $\widehat{\mathbf{G}}$ and $\widehat{\mathbf{C}}$ are sparse.

SparseRC is a multi-terminal *RC* reduction method which meets targets (a)-(e), as will be shown.

4.2.2 Multi-terminal *RC* reduction with moment matching

The *extended moment matching projection (EMMP)* is a moment matching reduction method for multi-terminal *RC* circuits derived from PACT [56] (and conceptually similar to SIP [94]). Being suitable for multi-terminal RC circuits with relatively few terminals only, this projection will be applied, after partitioning, in a block-wise manner inside SparseRC, as described in Sect. 4.3.2. The description here covers only material from [56] that is relevant for SparseRC.

Original circuit model (6.1): $\mathbf{G}, \mathbf{C} \in \mathbb{R}^{(n+p)\times(n+p)}$, $\mathbf{B} \in \mathbb{R}^{(n+p)\times p}$ Recalling (6.1), let the nodes $\mathbf{x}$ be split into *selected nodes* $\mathbf{x}_S$ (terminals and separator nodes[2]) to be preserved, and internal nodes to be eliminated $\mathbf{x}_R$, revealing the following structure:

$$\left(\begin{bmatrix} \mathbf{G}_R & \mathbf{G}_K \\ \mathbf{G}_K^T & \mathbf{G}_S \end{bmatrix} + s \begin{bmatrix} \mathbf{C}_R & \mathbf{C}_K \\ \mathbf{C}_K^T & \mathbf{C}_S \end{bmatrix} \right) \begin{bmatrix} \mathbf{x}_R \\ \mathbf{x}_S \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{B}_S \end{bmatrix} \mathbf{u}. \tag{4.2}$$

Note that [56] uses a simple block separation into "purely" terminal nodes $\mathbf{x}_S$ and internal nodes $\mathbf{x}_R$. Promoting separator nodes along with terminals inside $\mathbf{x}_S$ will ultimately positively influence the sparsity of the reduced model. The congruence transform applied to (4.2), $\mathbf{X}^T\mathbf{G}\mathbf{X}$, $\mathbf{X}^T\mathbf{C}\mathbf{X}$, $\mathbf{X}^T\mathbf{B}$, where [56]:

$$\mathbf{X} \;=\; \left[\begin{array}{c|c} \mathbf{I} & -\mathbf{G}_R^{-1}\mathbf{G}_K \\ \mathbf{0} & \mathbf{I} \end{array}\right], \quad \mathbf{x}' = \mathbf{X}^T\mathbf{x}, \quad \text{yields :} \tag{4.3}$$

[2] See Sect. 4.1 and Sect.4.3.1 for the definition of separator nodes

$$\left(\left[\begin{array}{c|c} \mathbf{G}_R & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{G}_S^{'} \end{array}\right] + s \left[\begin{array}{c|c} \mathbf{C}_R & \mathbf{C}_K^{'} \\ \hline \mathbf{C}_K^{'T} & \mathbf{C}_S^{'} \end{array}\right] \right) \left[\begin{array}{c} \mathbf{x}_R \\ \hline \mathbf{x}_S^{'} \end{array}\right] = \left[\begin{array}{c} \mathbf{0} \\ \hline \mathbf{B}_S \end{array}\right] \mathbf{u} \tag{4.4}$$

where:

$$\begin{aligned} \mathbf{G}_S^{'} &= \mathbf{G}_S - \mathbf{G}_K^T \mathbf{G}_R^{-1} \mathbf{G}_K, \ \mathbf{W} = -\mathbf{G}_R^{-1} \mathbf{G}_K & (4.5) \\ \mathbf{C}_S^{'} &= \mathbf{C}_S + \mathbf{W}^T \mathbf{C}_R \mathbf{W} + \mathbf{W}^T \mathbf{C}_K + \mathbf{C}_K^T \mathbf{W}, & (4.6) \\ \mathbf{C}_K^{'} &= \mathbf{C}_K + \mathbf{C}_R \mathbf{W}. \end{aligned}$$

Expressing $\mathbf{x}_R$ in terms of $\mathbf{x}_S^{'}$ from the first block-row of (4.4), and replacing it in the second gives:

$$[\underbrace{(\mathbf{G}_S^{'} + s\mathbf{C}_S^{'})}_{\mathbf{Y}_S^{'}(s)} - s^2 \underbrace{\mathbf{C}_K^{'T} (\mathbf{G}_R^{'} + s\mathbf{C}_R^{'})^{-1} \mathbf{C}_K^{'}}_{\mathbf{Y}_R^{'}(s)}] \mathbf{x}_S^{'} = \mathbf{B}_S \mathbf{u}.$$

The expression (4.7) represents the circuit's *multi-port admittance*, defined with respect to the selected nodes $\mathbf{x}_S$. $\mathbf{Y}^{'}(s)$ captures the first two multi-port admittance moments at $s = 0$ [56]. This is formalized as Proposition 4.2.1.

Proposition 4.2.1 *For a multi-terminal RC circuit of the form (4.2), the first two moments at $s = 0$ of the multi-port admittance are given by $\mathbf{G}_S^{'}$ and $\mathbf{C}_S^{'}$ from (4.5), (4.6).*

Proof 4.2.1 (Proof of Prop. 4.2.1) *Expressing $\mathbf{x}_R$ in terms of $\mathbf{x}_S$ from the first equation of (4.2), and replacing it the second gives the circuit's multi-port admittance:*

$$\begin{aligned} &\mathbf{Y}(s)\mathbf{x}_S = \mathbf{B}_S \mathbf{u}, \ \text{where} \\ &\mathbf{Y}(s) = (\mathbf{G}_S + s\mathbf{C}_S) - (\mathbf{G}_K + s\mathbf{C}_K)^T (\mathbf{G}_R + s\mathbf{C}_R)^{-1} (\mathbf{G}_K + s\mathbf{C}_K) \end{aligned} \tag{4.7}$$

The first 2 moments of $\mathbf{Y}(s)$ at $s = 0$ are computed from $\mathbf{Y}(s)|_{s=0}$ and $\frac{d\mathbf{Y}}{ds}(s)|_{s=0}$:

$$\begin{aligned} \mathbf{Y}(s)|_{s=0} &= \mathbf{G}_S - \mathbf{G}_K^T \mathbf{G}_R^{-1} \mathbf{G}_K = \mathbf{G}_S^{'}, \\ \frac{d\mathbf{Y}}{ds}(s) &= \mathbf{C}_S - \mathbf{C}_K^T (\mathbf{G}_R + s\mathbf{C}_R)^{-1} (\mathbf{G}_K + s\mathbf{C}_K) - (\mathbf{G}_K + s\mathbf{C}_K)^T (\mathbf{G}_R + s\mathbf{C}_R)^{-1} \mathbf{C}_K + \\ &+ (\mathbf{G}_K + s\mathbf{C}_K)^T (\mathbf{G}_R + s\mathbf{C}_R)^{-1} \mathbf{C}_R (\mathbf{G}_R + s\mathbf{C}_R)^{-1} (\mathbf{G}_K + s\mathbf{C}_K) \\ \Rightarrow \ \frac{d\mathbf{Y}}{ds}(s)|_{s=0} &= \mathbf{C}_S - \mathbf{C}_K^T \mathbf{G}_R^{-1} \mathbf{G}_K - \mathbf{G}_K^T \mathbf{G}_R^{-1} \mathbf{C}_K + \mathbf{G}_K^T \mathbf{G}_R^{-1} \mathbf{C}_R \mathbf{G}_R^{-1} \mathbf{G}_K = \mathbf{C}_S^{'}, \end{aligned}$$

the same as $\mathbf{G}_S^{'}$ and $\mathbf{C}_S^{'}$ from (4.5), (4.6).

The practical consequence of Prop. 4.2.1 is that, as with ReduceR, the path resistance of the original circuit is precisely (4.5) and, as shown next, is preserved by the reduced

model. In addition to the path resistance, the slope of an RC circuit's response is captured by the second moment, namely (4.6).

Reduced circuit model: $\widehat{\mathbf{G}}, \widehat{\mathbf{C}} \in \mathbb{R}^{(k+p)\times(k+p)}$, $k \geq 0$ By Prop. 4.2.1, the reduced model which preserves the first two admittance moments of the original (4.2) is revealed: eliminate nodes $\mathbf{x}_R$ (and the contribution $\mathbf{Y}_R^{'}$) and retain nodes $\mathbf{x}_S^{'}$. The corresponding moment matching projection is obtained by removing from $\mathbf{X}$ of (4.3) the columns corresponding to $\mathbf{x}_R$:

$$\mathbf{V} = \begin{bmatrix} -\mathbf{G}_R^{-1}\mathbf{G}_K \\ \mathbf{I} \end{bmatrix}, \tag{4.8}$$

$$\widehat{\mathbf{G}} = \mathbf{V}^T\mathbf{G}\mathbf{V} = \mathbf{G}_S^{'}, \ \widehat{\mathbf{C}} = \mathbf{V}^T\mathbf{C}\mathbf{V} = \mathbf{C}_S^{'}, \tag{4.9}$$

$$\widehat{\mathbf{B}} = \mathbf{V}^T\mathbf{B} = \mathbf{B}_S, \ \widehat{\mathbf{x}} = \mathbf{V}^T\mathbf{x} = \mathbf{x}_S^{'} \tag{4.10}$$

For simplicity the reducing projection $\mathbf{V}$ from (4.8) shall be referred to further-on as the extended moment matching projection (EMMP). The term "extended" denotes that moments are matched of the multi-port admittance defined by terminals *and* the preserved internal nodes, rather than, as in PACT, by terminal nodes only. In other words, EMMP is the extension of the original projection from PACT [56] or SIP [94] to include the separator nodes.

On the singularity of G

Conductance $\mathbf{G}$ and capacitance $\mathbf{C}$ matrices describing parasitic *RC* circuits in MNA form are often singular, thus one must ensure that the EMMP projection (4.8) inverts only non-singular $\mathbf{G}_R$ blocks. This is easily achieved by exploiting the properties of MNA matrices (e.g., definiteness, diagonal dominance), and a simple grouping of nodes so that internal nodes (i.e. rows/columns) responsible for the singularity of $\mathbf{G}$ are excluded from $\mathbf{G}_R$ (and promoted to $\mathbf{G}_S$) without any accuracy loss. Lemma 4.2.1 formalizes these actions. Similar actions for ensuring the invertibility of $\mathbf{G}_R$ are detailed in [56].

Lemma 4.2.1 *From the singular conductance matrix* $\mathbf{G}$ *underlying the MNA circuit equations (4.2), an invertible sub-block* $\mathbf{G}_R$ *can always be found.*

Proof 4.2.2 *Towards showing the invertibility of* $\mathbf{G}_R$*, an analogy is first emphasized between the circuit's topology and the properties of the* $\mathbf{G}, \mathbf{C}$ *matrices describing the MNA equations (6.1).* $\mathbf{G}$ *and* $\mathbf{C}$ *correspond to ungrounded*[3] *circuits and consequently the matrix pencil (*$\mathbf{G}, \mathbf{C}$*) is singular*

[3] A ground node is only chosen in the actual circuit simulation phase, depending on the simulation requirements and how the *RC* parasitics are connected to other circuitry. The ground node is thus interpreted

(the vector of all 1's form their column null-space). Therefore traditional MOR approaches which assume that ($\mathbf{G}, \mathbf{C}$*) is a regular pencil describing a grounded circuit cannot be directly applied in this context. SparseRC however can handle singular (*$\mathbf{G}$, $\mathbf{C}$*) matrix pencils, as long as the* $\mathbf{G}_R$ *block involved in the EMMP projection is invertible, as is shown next.*

Matrices $\mathbf{G}$, $\mathbf{C}$ *underlying an ungrounded circuit have the following properties:* $\mathbf{G}_{i,i} \geq 0$, $\mathbf{C}_{i,i} \geq 0$, $\mathbf{G}_{i,j} \leq 0$, $\mathbf{C}_{i,j} \leq 0$ *for* $i \neq j$*. Furthermore they are diagonally dominant satisfying* $\mathbf{G}_{i,i} + \Sigma_{i \neq j}\mathbf{G}_{i,j} = 0$, $\mathbf{C}_{i,i} + \Sigma_{i \neq j}\mathbf{C}_{i,j} = 0$*. In other words they are symmetric, positive-semi-definite Z-matrices, with positive diagonal entries, negative (or 0) off-diagonal entries, and possibly rows/columns entirely with 0's*[4]*. As an illustrative example, the circuit in Fig. 4.1 with nodes ordered according to (4.2) has the MNA representation (4.11). In general* $\mathbf{G}_R$*, being a sub-*

$$\underbrace{\left(\left[\begin{array}{cc|cccc} g_1{+}g_2{+}g_3{+}g_4{+}g_6 & -g_4 & -g_1 & -g_2 & -g_3 & -g_6 \\ -g_4 & g_4 & 0 & 0 & 0 & 0 \\ \hline -g_1 & 0 & g_1 & 0 & 0 & 0 \\ -g_2 & 0 & 0 & g_2 & 0 & 0 \\ -g_3 & 0 & 0 & 0 & g_3{+}g_5 & -g_5 \\ -g_6 & 0 & 0 & 0 & -g_5 & g_5{+}g_6 \end{array}\right]}_{\mathbf{G}} + \cdots \right.$$

$$\left. \underbrace{s\left[\begin{array}{cc|cccc} c_1{+}c_2 & 0 & 0 & -c_2 & 0 & -c_1 \\ 0 & c_3 & 0 & -c_3 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 \\ -c_2 & -c_3 & 0 & c_3{+}c_2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ -c_1 & 0 & 0 & 0 & 0 & c_1 \end{array}\right]}_{\mathbf{C}} \right) \underbrace{\left[\begin{array}{c} V_3(s) \\ V_4(s) \\ \hline V_1(s) \\ V_2(s) \\ V_5(s) \\ V_6(s) \end{array}\right]}_{\mathbf{x}} = \underbrace{\left[\begin{array}{cccc} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \hline 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{array}\right]}_{\mathbf{B}} \underbrace{\left[\begin{array}{c} i_1 \\ i_2 \\ i_5 \\ i_6 \end{array}\right]}_{\mathbf{u}} \tag{4.11}$$

block of $\mathbf{G}$ *will also have positive diagonal entries, negative (or 0) off-diagonals and possibly 0 rows/columns. The "structural" singularity of* $\mathbf{G}_R$ *due to the 0 rows/columns is easily removed by promoting the empty rows/columns to the* $\mathbf{G}_S$ *block. Assuming this procedure performed, the following scenarios apply for the remaining structure of* $\mathbf{G}_R$ *(as a sub-block of* $\mathbf{G}$*):*

1. $\mathbf{G}_{R_{i,i}} > \Sigma_{i \neq j}\mathbf{G}_{R_{i,j}}$*, i.e.,* $\mathbf{G}_R$ *is strictly diagonally dominant thus invertible*

2. $\mathbf{G}_R$ *may still contain some (not all) rows/columns whose sum is 0. In this case* $\mathbf{G}_R$ *satisfies the following property: the diagonal entries of* $\mathbf{G}_R$ *are positive and there exists a diagonal matrix D such that* $\mathbf{G}_R \cdot D$ *is strictly diagonally dominant [assume w.l.o.g row i is such that* $\mathbf{G}_{R_{i,i}} + \Sigma_{i \neq j}\mathbf{G}_{R_{i,j}} = 0$ *and* $\mathbf{G}_{R_{j,j}} \geq \Sigma_{j \neq k}\mathbf{G}_{R_{j,k}}$ *for all other rows* $j \neq i$*; then one*

as a terminal and must be preserved after reduction. Grounding the netlist before MOR would fix a particular node, remove the corresponding row and column from $\mathbf{G}$, $\mathbf{C}$ and destroy connectivity via this node, which is undesirable in practice.

[4] A 0 row/column in $\mathbf{G}$ ($\mathbf{C}$ respectively) denotes a node to which no resistor (or capacitor respectively) is connected, which is common in practice.

can take $D = \mathrm{diag}(\mathbf{I}_{i\neq j}, g_l)$ *for some arbitrary* $g_l > \mathbf{G}_{i,i}$*]. This is equivalent to* $\mathbf{G}_R$ *invertible [25].*

3. $\mathbf{G}_R$ *contains only row/columns whose sum is* 0 *(thus still singular). In this case the corresponding* $\mathbf{x}_R$ *nodes are eliminated for free (since the corresponding* $\mathbf{G}_K = \mathbf{0}$ *and the two moments at* $s = 0$ *of the multi-port admittance are preserved automatically). Thus no inversion of* $\mathbf{G}_R$ *is required.*

The $\mathbf{G}_R$ *block thus remains invertible. Singularity in* $\mathbf{C}$ *can be treated similarly, however not needed in this chapter.*

Reduction via the EMMP already meets some of the challenges defined at the beginning of Sect. 4.2: (a) two multi-port admittance moments are preserved irrespective of the separation level of $\mathbf{x}$ into $\mathbf{x}_R$ and $\mathbf{x}_S$, provided that $\mathbf{x}_R$ are internal nodes (thus the incidence matrix $\mathbf{B}_R = \mathbf{0}$); this ensures that accuracy is maintained via moment matching also when EMMP is later applied in the partitioned framework (see Sect. 4.3.2), (b) passivity is preserved [56], as $\mathbf{V}$ is a congruence transformation projecting the original positive semi-definite matrices $\mathbf{G}$ and $\mathbf{C}$ into reduced matrices $\widehat{\mathbf{G}}$, $\widehat{\mathbf{C}}$ which remain positive semi-definite, and (d) the input/output incidence matrix $\mathbf{B}_S$ remains un-altered after reduction; consequently, the reduced model can be reconnected directly via the terminal nodes to remaining circuitry (e.g. non-linear devices), without introducing new circuit elements such as controlled sources. The efficiency (c) and sparsity (e) considerations however are not met by EMMP alone when the circuits to be reduced have nodes, circuit elements, and terminals exceeding thousands. On one hand, constructing $\mathbf{G}_R^{-1}\mathbf{G}_K$ is either too costly or unfeasible, on the other the $\widehat{\mathbf{G}}$, $\widehat{\mathbf{C}}$ resulting from (4.9) may become too dense.

4.2.3 Fill-in and its effect in synthesis

Usually, $\mathbf{G}$ and $\mathbf{C}$ describing circuits from real chip designs are large and sparse, while the $\widehat{\mathbf{G}}$ and $\widehat{\mathbf{C}}$ as obtained from (4.9) are small, but dense. Furthermore, they are only mathematical constructions, thus a *synthesis* procedure is required to convert the reduced matrix representation back into an *RC* netlist. This is obtained by unstamping the non-zero entries of $\widehat{\mathbf{G}}$ and $\widehat{\mathbf{C}}$ into the corresponding resistor and capacitor topology respectively, while $\widehat{\mathbf{B}}$ (being a sub-matrix of $\mathbf{B}_S$) is mapped directly into the original current injection at terminals. Unstamping is done via *RLCSYN* [93]. The dimension of $\widehat{\mathbf{G}}$ and $\widehat{\mathbf{C}}$ gives the number of nodes, while the number of their non-zero entries dictates how many *R*s and *C*s are present in the reduced netlist. Therefore, while limiting the size of $\widehat{\mathbf{G}}$ and $\widehat{\mathbf{C}}$, it is critical to also ensure their sparsity.

The simple example in Fig. 4.1 compares two reduced netlists derived from a small circuit. The dense reduced model has fewer nodes but more R, C elements than the

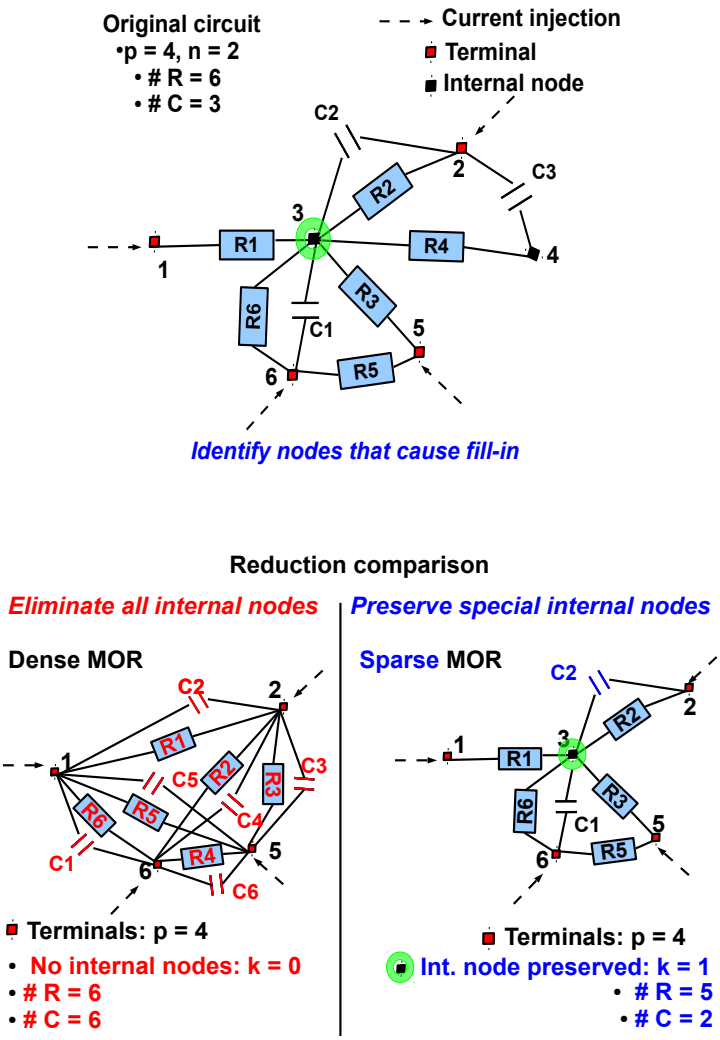


Figure 4.1: *Top: RC circuit to be reduced, containing $p = 4$ terminals and $n = 2$ internal nodes. Node 3 is a special internal node with many connections to other nodes. Bottom left: a dense reduced model, where all internal nodes (3 and 4) were eliminated, but more circuit elements are generated. Bottom right: a sparse reduced model (with fewer circuit elements) obtained from keeping node 3 and eliminating only node 4.*

original, while the sparse reduced model has both fewer nodes and R, C elements. The sparse model was obtained by preserving a node which would introduce fill-in if eliminated. Naturally, identifying such nodes by inspection is no longer possible for very large circuits. Next, it is explained how avoiding fill-creating nodes is possible in practice using reordering techniques.

Improving sparsity with node reorderings

At the basis of sparsity preserving MOR lies the following observation: the congruence transform $\mathbf{X}$ from (4.3) is analogous to a partial Cholesky factorization [29] of $\mathbf{G}$ [94]. Just as fill-reducing matrix reorderings are used for obtaining sparser factorizations, so can they be applied for sparsity preserving model reduction. These lie at the heart of ReduceR [80] and SIP [94], where the idea is to pre-order the matrix for instance with *Constrained Approximate Minimum Degree (CAMD)* [1, 2], so that nodes responsible for fill-in are placed towards the end of the elimination sequence, along with the terminals. By eliminating the nodes one by one and keeping track of the fill-in generated at each step, the circuit with the fewest number of elements can be determined. For circuits with challenging topologies however (i.e. with more internal nodes, terminals or circuit elements), these actions may be either too costly or even unfeasible (see the results in Sect. 4.4.2). SparseRC avoids such hurdles by exploiting graph partitioning and an appropriate matrix structure which allow for separator (fill-creating) nodes to be identified and skipped automatically in the reduction process, thus ensuring a desirable level of sparsity.

The following sections show how SparseRC, building upon the EMMP in combination with graph-partitioning and sometimes additional fill-reducing orderings, maintains the (a),(b),(d) and in addition meets the (c) efficiency and (e) sparsity requirements. These are crucial for successfully reducing very large networks with many terminals arising in industrial problems.

4.3 SparseRC reduction via graph partitioning

The analogy between circuits and graphs is immediate: the circuit nodes are the graph vertices, while the connections among them via R, C elements form the edges in the graph. For very large multi-terminal circuits to be manageable at all with limited computational resources, a global partitioning scheme is proposed, visualized with the help of Fig. 4.2[5]. Essentially, an original large-multi terminal circuit is split into subnetworks which have fewer nodes and terminals, are minimally connected among each-other via separator nodes, and are reduced individually up to the terminals and separator (cut) nodes. SparseRC reduces each subnet individually, up to terminals and separator nodes. As all separator nodes are automatically preserved, sparsity is improved.

[5]The two-way partitioning is presented here for simplicity; a natural extension is partitioning into $N > 2$ subnets.

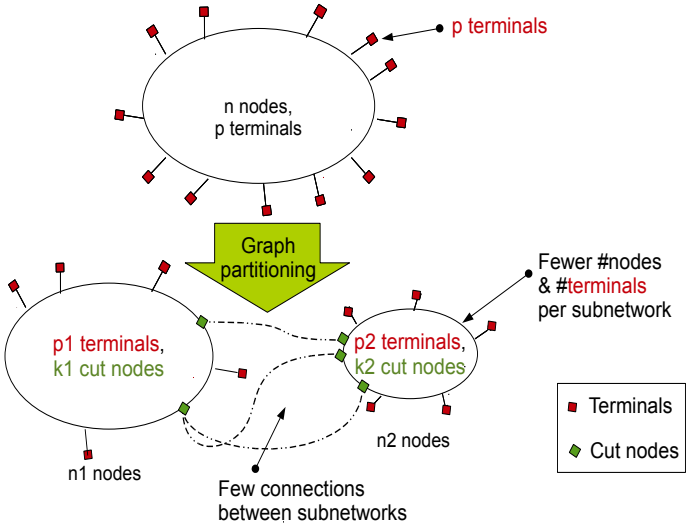


Figure 4.2: *Graph partitioning with separation of terminals.*

4.3.1 Partitioning and the BBD form

Implemented as a *divide and conquer* reduction strategy, SparseRC first uses graph decompositions [based on the *non-zero pattern (nzp)* of $\mathbf{G} + \mathbf{C}$] to identify individual subnets, as well as the separator nodes through which these communicate. Through partitioning, the original circuit matrices are reordered into the *bordered block diagonal (BBD)* [97] form: individual blocks form the subnets to be reduced, while the border blocks collect the separator nodes which are *all* preserved. The separator nodes are identified with the help of a *separator tree* indicating which of the subnets resulting from partitioning are in fact separators (this is usually a direct functionality of the partitioning algorithm, see for instance the MATLAB [67] manual pages of `nesdis` reordering). In the *conquer* phase, individual blocks are reduced with the EMMP from Sect. 4.2.2 and the border blocks are correspondingly updated to maintain the moment matching property. A graphical representation of the 7-component BBD partitioning and reduction is shown in Fig. 4.3[6]. It is emphasized that, as reduction progresses, fill-in is isolated in the reduced parts of C_1, C_2, C_4, C_5, the separator blocks C_3, C_6, C_7, and the corresponding connection borders. As components are minimally connected, fill-in generated on the border is minimized.

[6] In implementation both $\mathbf{G}$ and $\mathbf{C}$ are in BBD form, in Fig. 4.3 G denotes simultaneously the corresponding blocks from both matrices.

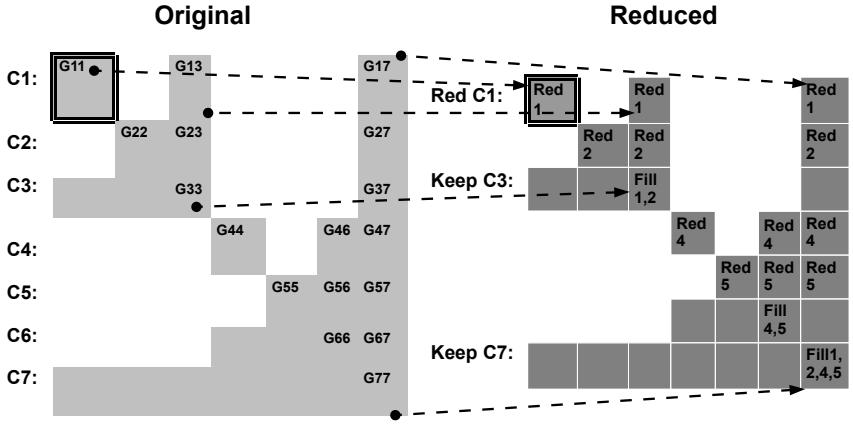


Figure 4.3: *Circuit matrices after partitioning, in BBD form (original-left vs. reduced-right). For clarity, block dimensions are not drawn to scale: in practice the separators are much smaller than the independent components, thus the borders are "thin". The individual blocks are reduced up to terminals, the borders are retained and updated. The number inside each independent component denotes the reduction step; this number is also stamped into the corresponding border blocks to mark fill-in. Example: reducing C_1 also updates the separators C_3 and C_7 and the corresponding borders. The "root" separator C_7 is updated from reducing all individual blocks $C_{1,2,4,5}$.*

4.3.2 Mathematical formulation

The mathematical derivation of SparseRC follows, showing how reduction progressively traverses the BBD matrix structure, reducing individual components and updating the connectivity among them. Herein, $\mathcal{G}$ and $\mathcal{C}$ shall denote the original circuit matrices, while $\mathbf{G}$, $\mathbf{C}$ shall directly refer to matrix blocks associated with the EMMP reduction from Sect. 4.2.2. Reconsider the original *RC* circuit in MNA form:

$$(\mathcal{G} + s\mathcal{C})\mathbf{x}(s) = \mathcal{B}\mathbf{u}(s), \tag{4.12}$$

of dimension $n + p$, where n are internal nodes, p are terminals. The appropriate projection $\mathcal{V} \in \mathbb{R}^{(n+p)\times(k+p)}$, $k \geq 0$, is sought, which reduces (4.12) to:

$$\widehat{\mathcal{G}}{=}\mathcal{V}^T\mathcal{G}\mathcal{V} \in \mathbb{R}^{(k+p)\times(k+p)}\,,\ \widehat{\mathcal{C}}{=}\mathcal{V}^T\mathcal{C}\mathcal{V} \in \mathbb{R}^{(k+p)\times(k+p)} \tag{4.13}$$

$$\widehat{\mathbf{x}}{=}\mathcal{V}^T\mathbf{x} \in \mathbb{R}^{(k+p)}\,,\ \widehat{\mathcal{B}}{=}\mathcal{V}^T\mathcal{B} \in \mathbb{R}^{(k+p)\times p} \tag{4.14}$$

As illustrated in Sect. 4.3.1, $\mathcal{V}$ is constructed step-wise using the BBD matrix reordering. Mathematically this is shown via the simplest example of a BBD partitioning: a *bisection* into two independent components communicating via one separator (border) block.

General reduction for a multi-level BBD partitioned system follows similarly. Consider the bisection of (4.12):

$$\begin{bmatrix} \mathcal{G}_{11} & 0 & \mathcal{G}_{13} \\ 0 & \mathcal{G}_{22} & \mathcal{G}_{23} \\ \mathcal{G}_{13}^T & \mathcal{G}_{13}^T & \mathcal{G}_{33} \end{bmatrix} + s \begin{bmatrix} \mathcal{C}_{11} & 0 & \mathcal{C}_{13} \\ 0 & \mathcal{C}_{22} & \mathcal{C}_{23} \\ \mathcal{C}_{13}^T & \mathcal{C}_{13}^T & \mathcal{C}_{33} \end{bmatrix} \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \mathbf{x}_3 \end{bmatrix} = \begin{bmatrix} \mathcal{B}_1 \\ \mathcal{B}_2 \\ \mathcal{B}_3 \end{bmatrix}. \tag{4.15}$$

Reducing (4.15) amounts to applying the EMMP from Sect. 4.2.2 on the individual components [here $C_1 := \text{nzp}(\mathcal{G}_{11} + \mathcal{C}_{11})$ and $C_2 := \text{nzp}(\mathcal{G}_{22} + \mathcal{C}_{22})$]. The separator [here $C_3 := \text{nzp}(\mathcal{G}_{33} + \mathcal{C}_{33})$] is kept and updated twice with the projections reducing C_1 and C_2 respectively. Naturally, the reduction is applied to the communication blocks $\mathcal{G}_{13}, \mathcal{C}_{13}, \mathcal{G}_{23}, \mathcal{C}_{23}$. Updating the separator and communication blocks at each individual reduction step ensures admittance moment preservation for the total recombined circuit (see Theorem 4.3.1 in Sect. 4.3.3).

Step 1

Consider the reduction of subnetwork C_1 with EMMP, based on splitting $\mathbf{x}_1$ of (4.15) into $\mathbf{x}_{1_R}$ and $\mathbf{x}_{1_S}$, i.e. into the internal nodes (to be eliminated) and selected nodes (to be preserved) of subnet C_1:

$$\left[\begin{array}{c|ccc} \mathcal{G}_{11_R} & \mathcal{G}_{11_K} & \mathbf{0} & \mathcal{G}_{13_R} \\ \hline \mathcal{G}_{11_K}^T & \mathcal{G}_{11_S} & \mathbf{0} & \mathcal{G}_{13_S} \\ \mathbf{0} & \mathbf{0} & \mathcal{G}_{22} & \mathcal{G}_{23} \\ \mathcal{G}_{13_R}^T & \mathcal{G}_{13_S}^T & \mathcal{G}_{23}^T & \mathcal{G}_{33} \end{array}\right] + s \left[\begin{array}{c|ccc} \mathcal{C}_{11_R} & \mathcal{C}_{11_K} & \mathbf{0} & \mathcal{C}_{13_R} \\ \hline \mathcal{C}_{11_K}^T & \mathcal{C}_{11_S} & \mathbf{0} & \mathcal{C}_{13_S} \\ \mathbf{0} & \mathbf{0} & \mathcal{C}_{22} & \mathcal{C}_{23} \\ \mathcal{C}_{13_R}^T & \mathcal{C}_{13_S}^T & \mathcal{C}_{23}^T & \mathcal{C}_{33} \end{array}\right] \tag{4.16}$$
$$\mathbf{x}^T = [\mathbf{x}_{1_R}^T, \mathbf{x}_{1_S}^T, \mathbf{x}_2^T, \mathbf{x}_3^T]^T, \;\; \mathcal{B}^T = [\mathbf{0}, \mathcal{B}_{1_S}^T, \mathcal{B}_2^T, \mathcal{B}_3^T]$$

Recognizing the analogy with (4.2), the EMMP-based transformation which reduces the network (4.16) by eliminating nodes the internal nodes $\mathbf{x}_{1_R}$ is given by:

$$\left[\begin{array}{ccc} -\mathcal{G}_{11_R}^{-1}\mathcal{G}_{11_K} & \mathbf{0} & -\mathcal{G}_{11_R}^{-1}\mathcal{G}_{13_R} \\ \hline \mathbf{I}_{S_1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_2 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I}_3 \end{array}\right] = \left[\begin{array}{c} -\mathbf{G}_R^{-1}\mathbf{G}_K \\ \hline \mathbf{I}_{S_123} \end{array}\right] = \mathbf{V}_1, \tag{4.17}$$

where $\mathbf{I}_{S_123} :=$blockdiag$(\mathbf{I}_{S_1}, \mathbf{I}_2, \mathbf{I}_3)$. Let:

$$\mathcal{W}_{11} = -\mathcal{G}_{11_R}^{-1}\mathcal{G}_{11_K}, \;\; \mathcal{W}_{13} = -\mathcal{G}_{11_R}^{-1}\mathcal{G}_{13_R}. \tag{4.18}$$

As with (4.8)-(4.10), the reduced model for (4.16) is computed from $\mathbf{G}_S^{'} = \mathbf{V}_1^T \mathcal{G} \mathbf{V}_1$, $\mathbf{C}_S^{'} = \mathbf{V}_1^T \mathcal{C} \mathbf{V}_1$. The reduced system at step 1 has the form:

$$\mathbf{G}_S^{'} = \begin{bmatrix} \widehat{\mathcal{G}}_{11} & \mathbf{0} & \widehat{\mathcal{G}}_{13} \\ \mathbf{0} & \mathcal{G}_{22} & \mathcal{G}_{23} \\ \widehat{\mathcal{G}}_{13}^T & \mathcal{G}_{23}^T & \widetilde{\mathcal{G}}_{33} \end{bmatrix} \quad , \quad \mathbf{C}_S^{'} = \begin{bmatrix} \widehat{\mathcal{C}}_{11} & \mathbf{0} & \widehat{\mathcal{C}}_{13} \\ \mathbf{0} & \mathcal{C}_{22} & \mathcal{C}_{23} \\ \widehat{\mathcal{C}}_{13}^T & \mathcal{C}_{23}^T & \widetilde{\mathcal{C}}_{33} \end{bmatrix} \tag{4.19}$$

$$\mathbf{B}_S = \begin{bmatrix} \widehat{\mathcal{B}}_1 \\ \mathcal{B}_2 \\ \mathcal{B}_3 \end{bmatrix} \quad , \quad \mathbf{x}_S^{'} = \begin{bmatrix} \widehat{\mathbf{x}}_1 \\ \mathbf{x}_2 \\ \mathbf{x}_3 \end{bmatrix}, \tag{4.20}$$

where:

$$\widehat{\mathcal{G}}_{11} = \mathcal{G}_{11_S} - \mathcal{G}_{11_K}^T \mathcal{G}_{11_R}^{-1} \mathcal{G}_{11_K}, \tag{4.21}$$

$$\widehat{\mathcal{G}}_{13} = \mathcal{G}_{13_S} - \mathcal{G}_{11_K}^T \mathcal{G}_{11_R}^{-1} \mathcal{G}_{13_R} \tag{4.22}$$

$$\widetilde{\mathcal{G}}_{33} = \mathcal{G}_{33} - \mathcal{G}_{13_R}^T \mathcal{G}_{11_R}^{-1} \mathcal{G}_{13_R} \tag{4.23}$$

$$\widehat{\mathcal{C}}_{11} = \mathcal{C}_{11_S} + \mathcal{W}_{11}^T \mathcal{C}_{11_R} \mathcal{W}_{11} + \mathcal{W}_{11}^T \mathcal{C}_{11_K} + \mathcal{C}_{11_K}^T \mathcal{W}_{11} \tag{4.24}$$

$$\widehat{\mathcal{C}}_{13} = \mathcal{C}_{13_S} + \mathcal{W}_{11}^T \mathcal{C}_{13_R} + \mathcal{C}_{11_K}^T \mathcal{W}_{13} + \mathcal{W}_{11}^T \mathcal{C}_{11_R} \mathcal{W}_{13} \tag{4.25}$$

$$\widetilde{\mathcal{C}}_{33} = \mathcal{C}_{33} + \mathcal{W}_{13}^T \mathcal{C}_{11_R} \mathcal{W}_{13} + \mathcal{W}_{13}^T \mathcal{C}_{13_R} + \mathcal{C}_{13_R}^T \mathcal{W}_{13} \tag{4.26}$$

$$\widehat{\mathcal{B}}_1 = \mathcal{B}_{1_S}, \ \widehat{\mathbf{x}}_1 = \mathbf{x}_{1_S}^{'}. \tag{4.27}$$

The BBD form provides an important structural advantage, both in terms of identifying fill-in, as well as in implementation: reducing one subnet only affects the entries of the corresponding separator and border blocks, leaving the rest of the independent subnets intact. Notice from (4.19)-(4.20) how reducing subnet C_1 has only affected its corresponding connection blocks to C_3, and the separator block C_3 itself. The blocks of subnet C_2 are not affected. This is because C_1 communicates with C_2 only via the separator C_3. Therefore while reducing C_2, the already computed blocks of the reduced C_1 will no longer be affected. Only the connection blocks from C_2 to C_3 and the separator C_3 itself will be updated. Mathematically, this is shown next.

Step 2

Partition now the reduced $\mathbf{G}_S^{'}$, $\mathbf{C}_S^{'}$ matrices (4.19)-(4.20) by splitting component C_2 according to $\mathbf{x}_{2_R}$ and $\mathbf{x}_{2_S}$:

$$\left[\begin{array}{cc|cc} \widehat{\mathcal{G}}_{11} & \mathbf{0} & \mathbf{0} & \widehat{\mathcal{G}}_{13} \\ \mathbf{0} & \mathcal{G}_{22_R} & \mathcal{G}_{22_K} & \mathcal{G}_{23_R} \\ \hline \mathbf{0} & \mathcal{G}_{22_K}^T & \mathcal{G}_{22_S} & \mathcal{G}_{23_S} \\ \widehat{\mathcal{G}}_{13}^T & \mathcal{G}_{23_R}^T & \mathcal{G}_{23_S}^T & \widetilde{\mathcal{G}}_{33} \end{array}\right] + s \left[\begin{array}{cc|cc} \widehat{\mathcal{C}}_{11} & \mathbf{0} & \mathbf{0} & \widehat{\mathcal{C}}_{13} \\ \mathbf{0} & \mathcal{C}_{22_R} & \mathcal{C}_{22_K} & \mathcal{C}_{23_R} \\ \hline \mathbf{0} & \mathcal{C}_{22_K}^T & \mathcal{C}_{22_S} & \mathcal{C}_{23_S} \\ \widehat{\mathcal{C}}_{13}^T & \mathcal{C}_{23_R}^T & \mathcal{C}_{23_S}^T & \widetilde{\mathcal{C}}_{33} \end{array}\right] \tag{4.28}$$

$$\mathbf{x}_S^{'T} = [\widehat{\mathbf{x}}_1^T, \mathbf{x}_{2_R}^T, \mathbf{x}_{2_S}^T, \mathbf{x}_3^T]^T, \ \mathbf{B}_S^{'T} = [\widehat{\mathcal{B}}_1^T, \mathbf{0}, \mathcal{B}_{1_S}^T, \mathcal{B}_3^T] \tag{4.29}$$

As before, the EMMP-based transformation which reduces the network (4.28) by eliminating nodes $\mathbf{x}_{2_R}$ is given by:

$$\left[\begin{array}{ccc} \mathbf{I}_{S_1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & -\mathcal{G}_{22_R}^{-1}\mathcal{G}_{22_K} & -\mathcal{G}_{22_R}^{-1}\mathcal{G}_{23_R} \\ \hline \mathbf{0} & \mathbf{I}_{S_2} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I}_3 \end{array}\right] = \left[\begin{array}{cc} \mathbf{I}_{S_1} & \mathbf{0} \\ \mathbf{0} & -\mathbf{G}_R^{-1}\mathbf{G}_K \\ \mathbf{0} & \mathbf{I}_{S_2 3} \end{array}\right] = \mathbf{V}_2, \tag{4.30}$$

The reduced model is obtained by projecting (4.28)-(4.29) with $\mathbf{V}_2$. $\widehat{\mathcal{G}} = \mathbf{V}_2^T \mathbf{G}_S^{'} \mathbf{V}_2$, $\widehat{\mathcal{C}} = \mathbf{V}_2^T \mathbf{C}_S^{'} \mathbf{V}_2$, $\widehat{\mathcal{B}} = \mathbf{V}_2^T \mathbf{B}_S$, $\widehat{\mathbf{x}} = \mathbf{V}_2^T \mathbf{x}_S^{'}$:

$$\widehat{\mathcal{G}} = \left[\begin{array}{ccc} \widehat{\mathcal{G}}_{11} & \mathbf{0} & \widehat{\mathcal{G}}_{13} \\ \mathbf{0} & \widehat{\mathcal{G}}_{22} & \widehat{\mathcal{G}}_{23} \\ \widehat{\mathcal{G}}_{13}^T & \widehat{\mathcal{G}}_{23}^T & \overline{\mathcal{G}}_{33} \end{array}\right], \quad \widehat{\mathcal{G}} = \left[\begin{array}{ccc} \widehat{\mathcal{C}}_{11} & \mathbf{0} & \widehat{\mathcal{C}}_{13} \\ \mathbf{0} & \widehat{\mathcal{C}}_{22} & \widehat{\mathcal{C}}_{23} \\ \widehat{\mathcal{C}}_{13}^T & \widehat{\mathcal{C}}_{23}^T & \overline{\mathcal{C}}_{33} \end{array}\right] \tag{4.31}$$

$$\widehat{\mathcal{B}} = \left[\begin{array}{c} \widehat{\mathcal{B}}_1 \\ \widehat{\mathcal{B}}_2 \\ \mathcal{B}_3 \end{array}\right], \quad \widehat{\mathbf{x}} = \left[\begin{array}{c} \widehat{\mathbf{x}}_1 \\ \widehat{\mathbf{x}}_2 \\ \mathbf{x}_3 \end{array}\right], \tag{4.32}$$

where (4.21)-(4.27) hold and:

$$\widehat{\mathcal{G}}_{22} = \mathcal{G}_{22_S} - \mathcal{G}_{22_K}^T \mathcal{G}_{22_R}^{-1} \mathcal{G}_{22_K}, \tag{4.33}$$

$$\widehat{\mathcal{G}}_{23} = \mathcal{G}_{23_S} - \mathcal{G}_{22_K}^T \mathcal{G}_{22_R}^{-1} \mathcal{G}_{23_R} \tag{4.34}$$

$$\overline{\mathcal{G}}_{33} = \widetilde{\mathcal{G}}_{33} - \mathcal{G}_{23_R}^T \mathcal{G}_{22_R}^{-1} \mathcal{G}_{23_R} \tag{4.35}$$

$$\widehat{\mathcal{C}}_{22} = \mathcal{C}_{22_S} + \mathcal{W}_{22}^T \mathcal{C}_{22_R} \mathcal{W}_{22} + \mathcal{W}_{22}^T \mathcal{C}_{22_K} + \mathcal{C}_{22_K}^T \mathcal{W}_{22} \tag{4.36}$$

$$\widehat{\mathcal{C}}_{23} = \mathcal{C}_{23_S} + \mathcal{W}_{22}^T \mathcal{C}_{23_R} + \mathcal{C}_{22_K}^T \mathcal{W}_{23} + \mathcal{W}_{22}^T \mathcal{C}_{22_R} \mathcal{W}_{23} \tag{4.37}$$

$$\overline{\mathcal{C}}_{33} = \widetilde{\mathcal{C}}_{33} + \mathcal{W}_{23}^T \mathcal{C}_{22_R} \mathcal{W}_{23} + \mathcal{W}_{23}^T \mathcal{C}_{23_R} + \mathcal{C}_{23_R}^T \mathcal{W}_{23} \tag{4.38}$$

$$\widehat{\mathcal{B}}_2 = \mathcal{B}_{2_S}, \ \widehat{\mathbf{x}}_2 = \mathbf{x}_{2_S}^{'}, \ \text{with} \tag{4.39}$$

$$\mathcal{W}_{22} = -\mathcal{G}_{22_R}^{-1} \mathcal{G}_{22_K}, \ \mathcal{W}_{23} = -\mathcal{G}_{22_R}^{-1} \mathcal{G}_{23_R}. \tag{4.40}$$

As seen from (4.35) and (4.38), separator blocks $\overline{\mathcal{G}}_{33}$ and $\overline{\mathcal{C}}_{33}$ are the further updated blocks $\widetilde{\mathcal{G}}_{33}, \widetilde{\mathcal{C}}_{33}$ (previously obtained from reducing C_1). The reduced model retains the BBD form, and the separator nodes are retained in the blocks corresponding to $\overline{\mathcal{G}}_{33}$ and $\overline{\mathcal{C}}_{33}$. The p terminals are distributed among C_1, C_2, C_3 as seen from the form of $\widehat{\mathcal{B}}$ in (4.32). Equations (4.27), (4.39) and (4.32) together show that the input/output incidence matrix is preserved after reduction, thus the reduced netlist obtained from RLCSYN [93] unstamping preserves connectivity via the terminal nodes. In the general case, block-wise reduction of finer BBD partitions (into $N > 3$ components) follows similarly, with the appropriate projections of separator and border blocks. The moment-matching, terminal connectivity and passivity requirements remain satisfied.

Options for further improving sparsity

Partitioning provides an additional structural and computational advantage: if necessary, additional reordering and minimum-fill tracking options such as those employed by ReduceR/SIP can be applied per subnet. Naturally, such operations come at additional computational cost, but are still more efficient than monitoring fill-in directly from the unpartitioned circuit. So, while separator nodes already improve sparsity *globally and automatically*, fill-monitoring actions may further identify additional internal nodes to be preserved *locally* in each subnet. For instance, in the reduction scenario of Step 1 (see Sect. 4.3.2), the $\mathcal{G}_{11}, \mathcal{C}_{11}$ blocks of (4.15) would be reordered with CAMD and fill-tracking would identify which additional internal nodes should be preserved along with terminals in $\mathbf{x}_{1_S}$. This may improve sparsity inside $\widehat{\mathcal{G}}_{11}$, $\widehat{\mathcal{C}}_{11}$ (and correspondingly in $\widehat{\mathcal{G}}_{13}$, $\widehat{\mathcal{C}}_{13}$, $\widetilde{\mathcal{G}}_{33}$, $\widetilde{\mathcal{C}}_{33}$) even beyond the level already achieved by preserving the separators nodes. In Sect. 4.4 examples are provided to illustrate this effect. Nevertheless, such fill-monitoring operations are not always necessary: the sparsity level achieved directly from partitioning and preserving separator nodes is often sufficient. This is discussed in Sect. 4.3.4.

4.3.3 Moment matching, passivity and synthesis

Note that as $\widehat{\mathcal{G}} = \mathbf{V}_2^T \mathbf{G}_S^{'} \mathbf{V}_2 = \mathbf{V}_2^T \mathbf{V}_1^T \mathcal{G} \mathbf{V}_1 \mathbf{V}_2$ (similarly for $\widehat{\mathcal{C}}$, $\widehat{\mathcal{B}}$), the reducing projection from (4.13)-(4.14) is $\mathcal{V} := \mathbf{V}_1 \mathbf{V}_2$, with $\mathbf{V}_1$, $\mathbf{V}_2$ as deduced in (4.17) and (4.30) respectively. In efficient implementations however $\mathbf{V}_1$, $\mathbf{V}_2$, and $\mathcal{V}$ are never formed directly, rather they are formed block-wise as just shown. Only $\mathcal{W}_{11}$, $\mathcal{W}_{13}$, $\mathcal{W}_{22}$, $\mathcal{W}_{23}$ from (4.18) and (4.40) respectively are explicitly formed. Next, it is shown that the $\mathcal{V}$ constructed from successive EMMPs matches the first two admittance moments at the end of the reduction procedure.

Theorem 4.3.1 *Consider the original circuit (4.15) with matrices partitioned and structured in BBD form, which is reduced by applying successive EMMP projections (see Sect. 4.2.2) on each subnet. The final reduced model (4.31)-(4.32) preserves the first two multi-port admittance moments around s=0 of each individual subnet and of the entire recombined circuit (4.15).*

Proof 4.3.1 (Proof of Theorem 4.3.1) *By Prop. 4.2.1, the reduced subnet 1 defined by $\widehat{\mathcal{G}}_{11}$ (4.21) and $\widehat{\mathcal{C}}_{11}$ (4.24), preserves the first two multi-port admittance moments at $s = 0$ of the original subnet 1 defined by $\mathcal{G}_{11}$, $\mathcal{C}_{11}$. The same holds for the reduced subnet 2, defined by $\widehat{\mathcal{G}}_{22}$ (4.33) and $\widehat{\mathcal{C}}_{22}$ (4.36). There remains to prove the admittance moment matching between the original (4.15) and reduced (4.31)-(4.32) recombined circuits. This is shown by reconstructing from the individual EMMPs, an EMMP projection $\mathcal{V}$ associated with entire circuit (4.15), as follows. Recall $\mathcal{G}$ from (4.16), where in addition nodes $\mathbf{x}_2$ of the second subnet are split into $\mathbf{x}_{2_R}$*

and $\mathbf{x}_{2_S}$ *as in Sect. 4.3.2:*

$$\mathcal{G}=\begin{bmatrix} \mathcal{G}_{11_R} & \mathcal{G}_{11_K} & \mathbf{0} & \mathbf{0} & \mathcal{G}_{13_R} \\ \mathcal{G}_{11_K}^T & \mathcal{G}_{11_S} & \mathbf{0} & \mathbf{0} & \mathcal{G}_{13_S} \\ \mathbf{0} & \mathbf{0} & \mathcal{G}_{22_R} & \mathcal{G}_{22_K} & \mathcal{G}_{23_R} \\ \mathbf{0} & \mathbf{0} & \mathcal{G}_{22_K}^T & \mathcal{G}_{22_S} & \mathcal{G}_{23_S} \\ \mathcal{G}_{13_R}^T & \mathcal{G}_{13_S}^T & \mathcal{G}_{23_R}^T & \mathcal{G}_{23_S}^T & \mathcal{G}_{33} \end{bmatrix}$$

Recall $\mathcal{V} = \mathbf{V}_1\mathbf{V}_2$ *= with* $\mathbf{V}_1$ *from (4.17) and* $\mathbf{V}_2$ *from (4.30). Inside the* $\mathbf{V}_1$*, let* $\mathbf{I}_2 = \mathrm{blockdiag}(\mathbf{I}_{R_2}, \mathbf{I}_{S_2})$ *be partitioned according to the splitting of* $\mathbf{x}_2$ *into* $\mathbf{x}_{2_R}$ *and* $\mathbf{x}_{2_S}$ *respectively. Then, by straightforward matrix multiplication:*

$$\mathcal{V} \;=\; \mathbf{V}_1\mathbf{V}_2 = \begin{bmatrix} -\mathcal{G}_{11_R}^{-1}\mathcal{G}_{11_K} & \mathbf{0} & -\mathcal{G}_{11_R}^{-1}\mathcal{G}_{13_R} \\ \mathbf{I}_{S_1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & -\mathcal{G}_{22_R}^{-1}\mathcal{G}_{22_K} & -\mathcal{G}_{22_R}^{-1}\mathcal{G}_{23_R} \\ \mathbf{0} & \mathbf{I}_{S_2} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I}_3 \end{bmatrix} \tag{4.41}$$

Let $\mathcal{P}$ *be the permutation swapping the second with the third block-rows of (4.41). Then, denoting* $\mathcal{V}_\mathcal{P} = \mathcal{P}\mathcal{V}$*:*

$$\mathcal{V}_\mathcal{P} \;=\; \begin{bmatrix} -\mathcal{G}_{11_R}^{-1}\mathcal{G}_{11_K} & \mathbf{0} & -\mathcal{G}_{11_R}^{-1}\mathcal{G}_{13_R} \\ \mathbf{0} & -\mathcal{G}_{22_R}^{-1}\mathcal{G}_{22_K} & -\mathcal{G}_{22_R}^{-1}\mathcal{G}_{23_R} \\ \mathbf{I}_{S_1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_{S_2} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I}_3 \end{bmatrix}. \tag{4.42}$$

Define the permuted matrices $\mathcal{G}_\mathcal{P} = \mathcal{P}\mathcal{G}\mathcal{P}^T$*,* $\mathcal{C}_\mathcal{P} = \mathcal{P}\mathcal{C}\mathcal{P}^T$*,* $\mathcal{B}_\mathcal{P} = \mathcal{P}\mathcal{B}$*, and notice their structure:*

$$\mathcal{G}_\mathcal{P}=\left[\begin{array}{cc|ccc} \mathcal{G}_{11_R} & \mathbf{0} & \mathcal{G}_{11_K} & \mathbf{0} & \mathcal{G}_{13_R} \\ \mathbf{0} & \mathcal{G}_{22_R} & \mathbf{0} & \mathcal{G}_{22_K} & \mathcal{G}_{23_R} \\ \hline \mathcal{G}_{11_K}^T & \mathbf{0} & \mathcal{G}_{11_S} & \mathbf{0} & \mathcal{G}_{13_S} \\ \mathbf{0} & \mathcal{G}_{22_K}^T & \mathbf{0} & \mathcal{G}_{22_S} & \mathcal{G}_{23_S} \\ \mathcal{G}_{13_R}^T & \mathcal{G}_{13_S}^T & \mathcal{G}_{23_R}^T & \mathcal{G}_{23_S}^T & \mathcal{G}_{33} \end{array}\right] := \left[\begin{array}{c|c} \mathbf{G}_R & \mathbf{G}_K \\ \hline \mathbf{G}_K^T & \mathbf{G}_S \end{array}\right], \tag{4.43}$$

$$\text{similarly } \mathcal{C}_\mathcal{P} := \left[\begin{array}{c|c} \mathbf{C}_R & \mathbf{C}_K \\ \hline \mathbf{C}_K^T & \mathbf{C}_S \end{array}\right], \quad \mathcal{B}_\mathcal{P} = \left[\begin{array}{c} \mathbf{0} \\ \mathbf{0} \\ \hline \mathcal{B}_{1_S} \\ \mathcal{B}_{2_S} \\ \mathcal{B}_{3_S} \end{array}\right] := \left[\begin{array}{c} \mathbf{0} \\ \hline \mathbf{B}_S \end{array}\right]. \tag{4.44}$$

From (4.43), (4.44) and the analogy with Sect. 4.2.2, one recognizes immediately in (4.42) the

EMMP:

$$\mathcal{V}_{\mathcal{P}}=\left[\begin{array}{c}-\mathbf{G}_R^{-1}\mathbf{G}_K\\ \hline \mathbf{I}_{S_1S_23}\end{array}\right],\ \mathbf{I}_{S_1S_23}:=\text{blockdiag}(\mathbf{I}_{S_1},\mathbf{I}_{S_2},\mathbf{I}_3). \tag{4.45}$$

From Prop. 4.2.1, the reduced model obtained by projecting (4.43), (4.44) with (4.45) matches the fist two moments around $s=0$ of the multi-port admittance (defined with respect to $\mathbf{B}_S$, the total number of terminals and separator nodes of the recombined circuit). Since $\mathcal{P}$ is a permutation satisfying $\mathcal{P}^T\mathcal{P}=\mathbf{I}$, it follows immediately that this reduced model is precisely:

$$\mathcal{V}_{\mathcal{P}}^T\mathcal{G}_{\mathcal{P}}\mathcal{V}_{\mathcal{P}}=\mathcal{V}^T\mathcal{G}\mathcal{V}=\widehat{\mathcal{G}},\ \ \mathcal{V}_{\mathcal{P}}^T\mathcal{C}_{\mathcal{P}}\mathcal{V}_{\mathcal{P}}=\mathcal{V}^T\mathcal{C}\mathcal{V}=\widehat{\mathcal{C}},$$
$$\mathcal{V}_{\mathcal{P}}^T\mathcal{B}_{\mathcal{P}}=\mathcal{V}^T\mathcal{B}=\widehat{\mathcal{B}}.$$

where (4.31)-(4.39) hold.

Matching behavior at higher frequencies

Although SparseRC is mainly based on matching the first two admittance moments at $s=0$ (which proved sufficient for most experiments of Sect. 4.4), it is possible, when necessary, to additionally improve approximation at higher frequency points.One possibility is to include contributions from the otherwise neglected term $\mathbf{Y}_R^{'}(s)$ of the admittance response (4.7). Let for simplicity $\mathbf{Q}$ be a transformation which reduces $\mathbf{Y}_R^{'}(s)$. One can for instance perform the traditional PRIMA [71] reduction of $\mathbf{Y}_R^{'}(s)$, which constructs for a chosen expansion point s_i the Krylov subspace: $\mathcal{K}_m(\mathbf{A}^{-1}\mathbf{C}_R,\mathbf{A}^{-1}\mathbf{C}_K^{'})=$ span $\left[(\mathbf{A}^{-1}\mathbf{C}_R)^{m-1}\mathbf{A}^{-1}\mathbf{C}_K^{'}\right]$, where $\mathbf{A}=\mathbf{G}_R+s_i\mathbf{C}_R$. If $\mathcal{K}_m\subseteq\mathbf{Q}$ then $\mathbf{Q}$ matches $2m$ multi-port admittance moments of $\mathbf{Y}_R^{'}(s)$ at s_i. As a special case, if $s_i=0$, then $\mathbf{Q}$ matches $2m$ multi-port admittance moments at zero of (4.7), additionally to the 0'th and 1'st which are matched by default. Another option is to include in $\mathbf{Q}$ eigenvectors associated with the dominant eigenvalues of $\mathbf{Y}_R^{'}(s)$, similarly to [56]. Be it either obtained from moment or pole matching (or a combination of both), it is easily verified that the projection $\mathbf{Q}$ which reduces $\mathbf{Y}_R^{'}(s)$ enters the transformation (4.3) as follows:

$$\mathbf{X}_{\mathbf{Q}}\ =\ \left[\begin{array}{c|c}\mathbf{Q} & \mathbf{W}\\ \mathbf{0} & \mathbf{I}\end{array}\right],\ \mathbf{W}=-\mathbf{G}_R^{-1}\mathbf{G}_K \tag{4.46}$$

To form $\mathbf{Q}$ within the partitioned framework, reconsider the reduction of subnet 1 from Sect. 4.3.2. The reducing transformation there is $\mathbf{V}_1$ (4.17), which preserved the first two admittance moments at $s=0$ and eliminated entirely the contribution of internal nodes $\mathbf{x}_{1_R}$. Rather than eliminating $\mathbf{x}_{1_R}$, let $\mathcal{Q}_1$ be the transformation which reduces the internal matrices $\mathcal{G}_{11_R}$ and $\mathcal{C}_{11_R}$. Similarly, during the reduction of subnet 2 (Sect. 4.3.2), let $\mathcal{Q}_2$ be the transformation which reduces $\mathcal{G}_{22_R}$ and $\mathcal{C}_{22_R}$. $\mathcal{Q}_1$ and $\mathcal{Q}_2$ enter the reducing transformation for the recombined network as follows:

$$\mathcal{X}_Q = \begin{bmatrix} \mathcal{Q}_1 & \mathcal{W}_{11} & \mathbf{0} & \mathbf{0} & \mathcal{W}_{13} \\ \mathbf{0} & \mathbf{I}_{S_1} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathcal{Q}_2 & \mathcal{W}_{22} & \mathcal{W}_{23} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I}_{S_2} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I}_3 \end{bmatrix}, \tag{4.47}$$

where (4.18) and (4.40) hold. Note that (4.47) is the extension with $\mathcal{Q}_1$ and $\mathcal{Q}_2$ of the default projection (4.41) which matches the first two multi-port admittance moments at $s = 0$.

In a similar manner, a projection can be constructed which matches directly moments at $s_i \neq 0$ of the entire multi-port admittance (4.7) (see [57] for details, which pertains more to *RLC* reduction). Given that a circuit's offset and slope at DC are precisely the first two admittance moments at $s = 0$ [56], matching these is a natural choice. We emphasize that, in contrast with SparseRC, direct moment matching at $s = 0$ cannot be achieved via PRIMA [71], or SPRIM [27] since the original $\mathcal{G}$ matrix is singular. To summarize, should additional accuracy be necessary when reducing *RC* circuits with SparseRC, it is safer to match the first two admittance moments at $s = 0$ and improve the response with contributions from the internal $\mathbf{Y}_R^{'}(s)$ term as above described. This approach was implemented in the partitioned framework for two examples in Sect. 4.4.1: the *RC* transmission line example where moments of $\mathbf{Y}_R^{'}(s)$ are matched in addition, and the low noise amplifier (**LNA**) where dominant eigenmodes of $\mathbf{Y}_R^{'}(s)$ are additionally matched.

SparseRC matches two moments at $s = 0$ by default. Hence, extra poles/moments can be added, if needed, after the default reduction, by re-arranging the projection $\mathcal{X}_Q$ so that the extra $\mathcal{Q}_i$ columns are formed in a separate reduction run. The decision on whether to add extra poles or moments is difficult to make a-priori, as is the case for any moment-based reduction method. One approach would be to compare the response of the original and default reduced order model for large frequencies. Should significant deviations be observed, then the addition of extra poles/moments is recommended. In PACT [56] poles are added based on a specified error and maximum operating frequency.

Passivity and synthesis

As with PACT [56], the reducing projection $\mathcal{V}$ is a congruence transformation applied on original symmetric, non-negative definite matrices, which gives reduced symmetric, non-negative definite matrices (4.31)-(4.32). Consequently [56], the final reduced model (4.31)-(4.32) is passive and stable, and the reduced netlist obtained from RLCSYN [93] unstamping remains passive irrespective of the values of the resulting *R*, *C* elements. If reduction is performed with the default two-moment matching at $s = 0$ (which was sufficient in all experiments except the two-port *RC* line of Sect. 4.4.1) the projection $\mathcal{V}$

of (4.41) guarantees that the unstamping of $\widehat{\mathcal{G}}$ generates only positive resistors. This is ensured by the special form of $\mathcal{V}$ which performs a Schur-complement operation on the original matrix $\mathcal{G}$ [91] (note that a standard moment matching projection as in PRIMA [71] does not guarantee positive resistors from unstamping, even though the reduced $\widehat{\mathcal{G}}$ is symmetric positive definite). While there may be negative capacitances resulting from unstamping $\widehat{\mathcal{C}}$, they do not violate the passivity of the netlist, nor its direct usability inside a simulator such as Spectre [16]. Furthermore, they do not prejudice the quality of the re-simulation results as confirmed by Sect. 4.4. In fact, as also motivated in SIP [94], dropping negative capacitors from the reduced netlist is in practice a dangerous operation, so all capacitors are safely kept in. In the case of reduction with additional accuracy as in Sect. 4.3.3, the unstamping of $\widehat{\mathcal{G}}$ may generate negative resistors; these again posed no difficulties in the simulations performed (e.g. AC, transient, periodic steady state). PartMOR [70] presents an alternative reduction and synthesis strategy which ensures positive-only elements.

4.3.4 SparseRC algorithm

The SparseRC pseudocode is outlined in Algorithms 1, 2, which describe the most relevant reduction case of matching the first two admittance moments at $s = 0$. To summarize Alg. 1: from the original circuit matrices $\mathcal{G}$, $\mathcal{C}$ and a vector of indices $\mathbf{e}$ denoting the original location of terminals (external nodes), SparseRC outputs the reduced circuit matrices $\widehat{\mathcal{G}}, \widehat{\mathcal{C}}$, and the vector $\widetilde{\mathbf{e}}$ denoting the new terminal locations. As an advanced option, "do_minfill" specifies whether additional fill-reducing orderings should be employed per partition. The graph G defined by the circuit topology [the non-zero pattern (*nzp*) of $\mathcal{G} + \mathcal{C}$] is partitioned into N components. A permutation P (which reorders the circuit matrices in BBD form) is obtained, together with a vector Sep indicating which of the N components is a separator. For each non-separator component C_k, defined by nodes $\mathbf{i}_k$, the corresponding matrix blocks are reduced with EMMP while accounting for the communication of C_k to the remaining components via the separator nodes $\mathbf{i}_{sep}$. All separator components are kept, after having been appropriately updated inside EMMP.

The $\mathcal{G}$, $\mathcal{C}$ matrices supplied at each step to EMMP (line 8 of Alg. 1) are updated in place, and the reduction follows according to Alg. 2. The $\mathbf{i}_k$ index selects from the supplied $\mathcal{G}$, $\mathcal{C}$ the component to be reduced (say, C_k), while $\mathbf{i}_{sep}$ are the indices of separator nodes through which $\mathbf{i}_k$ communicate with the rest of the circuit. If desired, at the entry of EMMP these nodes are reordered with CAMD, as to identify additional internal nodes which may further improve sparsity from reducing C_k (this operation however is only an advanced feature and often unnecessary). Internal and external nodes of component C_k are identified. Internal nodes $\mathbf{i}_R$ will be removed and the selected nodes $\mathbf{i}_S$ will be preserved (i.e. terminals of C_k, corresponding separator nodes, and possibly some additional internal nodes obtained from step 2). The corresponding matrix blocks are identified and the update matrix $\mathbf{W}$ is formed. The blocks corresponding to selected nodes $\mathbf{i}_S$ are updated, while those corresponding to the eliminated $\mathbf{i}_R$ nodes are re-

moved. At output, $\widehat{\mathcal{G}}$, $\widehat{\mathcal{C}}$ are the reduced matrices: internal nodes were eliminated *only* from the component defined by node indices $\mathbf{i}_k$, while nodes corresponding to the other components are untouched. The terminal locations of the reduced model are indexed by $\widetilde{\mathbf{e}}$.

Algorithm 1 $(\widehat{\mathcal{G}}, \widehat{\mathcal{C}}, \widetilde{\mathbf{e}})$ = SparseRC$(\mathcal{G}, \mathcal{C}, \mathbf{e}, \text{do_minfill})$

Given: original $\mathcal{G}, \mathcal{C}$, original vector of terminal indices $\mathbf{e}$, do_minfill $(0/1)$ option for minimum-fill reordering per subnet
Output: reduced $\widehat{\mathcal{G}}, \widehat{\mathcal{C}}$, updated vector of terminal indices $\widetilde{\mathbf{e}}$
1: Let graph $G := \text{nzp}(\mathcal{G} + \mathcal{C})$
2: (P, Sep) =partition(G, N)
3: $\mathcal{G} = \mathcal{G}(P, P), \mathcal{C} = \mathcal{C}(P, P), \mathbf{e} = \mathbf{e}(P)$ ▷ reorder in BBD
4: **for** component $C_k = 1 \dots N$ **do**
5: **if** $C_k \notin Sep$ **then** ▷ C_k is not a separator
6: $\mathbf{i}_k =$ index of nodes for component C_k
7: $\mathbf{i}_{sep} =$ index of separator nodes connecting C_k to components $C_{k+1} \dots C_N$
8: $(\mathcal{G}, \mathcal{C}, \mathbf{e})$ = EMMP$(\mathcal{G}, \mathcal{C}, \mathbf{i}_k, \mathbf{i}_{sep}, \mathbf{e}$, do_minfill) ▷ reduce C_k with EMMP
9: **else** keep separator component C_k
10: **end if**
11: **end for**
12: $\widehat{\mathcal{G}} = \mathcal{G}, \widehat{\mathcal{C}} = \mathcal{C}, \widetilde{\mathbf{e}} = \mathbf{e}$

A few computational remarks: to ensure numerical stability while forming the reduced matrix blocks at step 11 of Alg. 1, we apply rescaling to $\mathcal{C}$ (and/or $\mathcal{G}$). Based on Theorem 4.3.1, it is also ensured that the global moment matching projection which underlies SparseRC inherits the proved [94] full-column-rank properties of the SIP/PACT projection.

On the partitioning strategy

It was seen how preserving internal nodes along with terminals improves the sparsity of the reduced model. Good reduced models are *sparse and small*, i.e. have *minimum fill* and *few preserved internal nodes*. Towards obtaining reduced models with a suitable trade-off between sparsity and dimension, one may naturally ask: (a) what are the optimal partitioning criteria and the number of generated subnets N, and (b) when are additional fill-reducing node reorderings and fill-monitoring actions needed aside from partitioning?

The choice of N Through partitioning, the aim is to minimize the communication among subnets (via a small number of separator nodes) *and* spread the terminals across partitions, as to minimize the fill-in generated from reducing each partition (up to its terminals) and inside the separator blocks. Towards achieving this goal, this chapter relies on the general purpose partitioner nested dissection (*NESDIS*, part of [18]) the choice

Algorithm 2 $(\widehat{\mathcal{G}}, \widehat{\mathcal{C}}, \widetilde{\mathbf{e}})$ = EMMP($\mathcal{G}$, $\mathcal{C}$, $\mathbf{i}_k$, $\mathbf{i}_{sep}$, $\mathbf{e}$, do_minfill)

Given: initial $\mathcal{G}, \mathcal{C}$, corresponding vector of terminal indices $\mathbf{e}$, do_minfill $(0/1)$ option for minimum-fill node reordering

Output: reduced $\widehat{\mathcal{G}}, \widehat{\mathcal{C}}$, corresponding vector of terminals $\widetilde{\mathbf{e}}$

if do_minfill **then** ▷ find additional internal nodes to preserve
  $(\mathbf{i}_k, \mathbf{i}_{sep}, \mathbf{e})$ = reorderCAMD($\mathcal{G}$, $\mathcal{C}$, $\mathbf{i}_k$, $\mathbf{i}_{sep}$, $\mathbf{e}$)
▷ find optimal minimum fill ordering per subnet
end if
$(\mathbf{i}_{int}, \mathbf{i}_{ext})$ = split($\mathbf{i}_k$, $\mathbf{e}$) ▷ split $\mathbf{i}_k$ into internal and external nodes
$\mathbf{i}_R = \mathbf{i}_{int}$ ▷ internal nodes to eliminate
$\mathbf{i}_S = [\mathbf{i}_{ext}, \mathbf{i}_{sep}]$ ▷ selected nodes to keep
$\mathbf{G}_R = \mathcal{G}(\mathbf{i}_R, \mathbf{i}_R)$, $\mathbf{C}_R = \mathcal{C}(\mathbf{i}_R, \mathbf{i}_R)$
$\mathbf{G}_K = \mathcal{G}(\mathbf{i}_R, \mathbf{i}_S)$, $\mathbf{C}_K = \mathcal{C}(\mathbf{i}_R, \mathbf{i}_S)$
$\mathbf{G}_S = \mathcal{G}(\mathbf{i}_S, \mathbf{i}_S)$, $\mathbf{C}_S = \mathcal{C}(\mathbf{i}_S, \mathbf{i}_S)$
$\mathbf{W} = -\mathbf{G}_R^{-1}\mathbf{G}_K$ ▷ construct reducing projection
$\mathcal{G}(\mathbf{i}_S, \mathbf{i}_S) = \mathbf{G}_S - \mathbf{G}_K^T\mathbf{W}$ ▷ update entries for selected nodes
$\mathcal{C}(\mathbf{i}_S, \mathbf{i}_S) = \mathbf{C}_S + \mathbf{C}_K^T\mathbf{W} + \mathbf{W}^T\mathbf{C}_K + \mathbf{W}^T\mathbf{C}_R\mathbf{W}$
$\mathcal{G}(\mathbf{i}_R, \mathbf{i}_R) = \{\,\}$, $\mathcal{C}(\mathbf{i}_R, \mathbf{i}_R) = \{\,\}$ ▷ eliminate $\mathbf{i}_R$ nodes
$\mathcal{G}(\mathbf{i}_R, \mathbf{i}_S) = \{\,\}$, $\mathcal{C}(\mathbf{i}_R, \mathbf{i}_S) = \{\,\}$, $\mathbf{e}(\mathbf{i}_R) = \{\,\}$
$\widehat{\mathcal{G}} = \mathcal{G}$, $\widehat{\mathcal{C}} = \mathcal{C}$, $\widetilde{\mathbf{e}} = \mathbf{e}$

however is by no means exclusive. In [47] and Appendix 4.6.2 for instance the usage of the hypergraph partitioner Mondriaan [96] is documented. NESDIS partitions a graph so that communication among subnets is minimized, however cannot control explicitly the distribution of terminals across parts. With NESDIS, terminals get spread indirectly, as a consequence of partitioning. While estimating an optimal N is an open problem which must simultaneously account for multiple factors (number of terminals, internal nodes, elements, potential fill-in), we provide some guidelines as to quickly determine a satisfactory value to be used with NESDIS. For our experiments, N was mostly determined immediately by inspecting the ratio of terminals to internal nodes in the original graph, $\frac{p}{n}$. For netlists with small $\frac{p}{n}$ [for instance $\frac{p}{n} < \frac{1}{10}$], a coarse partitioning is already sufficient to achieve few terminals per subnet and ensure sparsity (certainly, as long as the resulting number of nodes per subnet enables the computation of the corresponding block projection). Such circuits are the ideal candidates for SparseRC reduction based on NESDIS partitioning alone, without extra fill-reducing ordering actions.

Additional fill-monitoring actions For circuits with large $\frac{p}{n}$ ratios though, finer NESDIS partitions are needed to achieve a small enough number of terminals per subnet (see the **Filter** net in Sect. 4.4.2). In this case, additional fill-reducing orderings and minimum-fill tracking actions may further improve the sparsity attained from partitioning, at the cost of preserving more internal nodes. In Sect. 4.4 examples illustrate the sparsity, dimensionality and computational implications of the partitioning fineness and, where needed, of additional fill-monitoring actions.

Clarifying remarks The functionality of SparseRC is not tied strictly to the partitioner used or the choice of N. It is assumed that the original circuits (graphs) are sparse. The sparsity of the original circuit will dictate the partitioning performance and consequently the dimension and sparsity level of the reduced circuit. Precise judgements on the optimal N or the necessity of additional fill-monitoring operations cannot be made a priori. These could be resolved by the following *multi-terminal graph optimization problem* [47]: For a multi-terminal network $G = (V, E)$, of which $P \subset V$ are terminals, $|P| = p$, find an N-way partitioning with the objective of minimizing the number of separator nodes subject to the following constraints: (a) the separator nodes and terminals are balanced across the N parts, and (b) eliminating the internal nodes from each subnet introduces minimum fill in the parts determined by terminals and separator nodes. Chapter 7 analyzes the multi-terminal partitioning problem in more detail. As concerns the approximation quality, the SparseRC model will always be at least as good as a PACT [56] reduced model. Due to Theorem 4.3.1 SparseRC guarantees not only local but also global moment matching for the recombined network, *irrespective of* N or the partitioner used.

Computational complexity

The computational complexity of SparseRC is dominated by the cost of computing $\mathbf{W}$ inside EMMP (line 9 in Alg. 2), for each of the $N_{max} < N$ non-separator components. With n_{max} denoting the maximum number of internal nodes for a component (i.e. the maximum size of block $\mathbf{G}_R$), and m_{max} the maximum size of $\mathbf{G}_S$, the cost of one EMMP operation is at most $\mathrm{O}(n_{max}^{\alpha} m_{max})$, with $1 < \alpha \leq 2$ [75]. When n and p are large and the circuit is partitioned, one aims at $n_{max} \ll n$ and $m_{max} \ll p$ [note that $m_{max} = p_{max} + s_{max}$, with p_{max} denoting the maximum number of terminals per component (i.e. length of $\mathbf{i}_{ext}$) and s_{max} the maximum number of separator nodes connecting a component C_k to components $C_{k+1} \ldots C_N$ (i.e. length of $\mathbf{i}_{sep}$)]. Therefore, especially for netlists with many internal nodes n and many terminals p, the total cost $\mathrm{O}(N_{max}(n_{max}^{\alpha} m_{max}))$ of SparseRC is much smaller than the $\mathrm{O}(n^{\alpha} p)$ cost of constructing (if at all feasible) a PACT reducing projection directly from the original, unpartitioned matrices. The results in Sect. 4.4.1, Table 4.1 confirm this through netlists 6.**RX** and 7.**PLL** which contain more than 300000 internal nodes and 4000 terminals. For a graph $G = (V, E)$ with $|V| = n + p$ nodes and $|E|$ edges, the cost of NESDIS partitioning (being based on METIS [53]) is $O(|E|)$ hence cheap to perform for sparse graphs (here, $|V|$ is the number of circuit nodes and $|E|$ the number of resistor and capacitors). Should additional reorderings be employed per subnet, the cost of CAMD is $O(|V||E|)$ [1, 2, 38], also a fast operation.

The cost for the advanced option of tracking fill-in is more expensive, especially for networks with nodes and elements exceeding thousands. The operation becomes a partial Gaussian elimination up to terminals, which may reach $O(|V|^3)$ in the worst case. Nonetheless, such fill-monitoring operations are only an optional, advanced feature of SparseRC which was rarely needed in our experiments.

With ingredients of SparseRC in hand, we summarize its properties in light of the reduction criteria defined in Sect. 4.2.1. SparseRC meets the accuracy (a), passivity (b), and terminal re-connectivity (d), requirements while reducing multi-terminal RC circuits via a block-wise EMMP reducing projection. The efficiency (e) of SparseRC is ensured via a partitioning-based implementation, which reduces much smaller subnets (also with fewer terminals) individually while maintaining the accuracy, passivity and re-connectivity requirements of the entire circuit. The sparsity (c) of the SparseRC reduced model is enhanced by preserving a subset of internal nodes which are identified automatically from partitioning and where necessary, from additional fill-reducing orderings. The performance of SparseRC in practice is shown by the numerical and simulations results presented next.

4.4 Numerical results and circuit simulations

Several parasitic extracted *RC* circuits from industrial applications are reduced with SparseRC. For each circuit, the terminals are nodes to which non-linear devices (such as diodes or transistors) are connected. During a parsing phase, the multi-terminal *RC* circuit is stamped into the MNA form (4.12) and reduced with SparseRC. The reduced model (4.31)-(4.32) is synthesized with RLCSYN [93] into its netlist description. As connectivity via the external nodes is preserved with SparseRC, no voltage/current controlled sources are generated during synthesis. The non-linear devices are directly re-coupled via the terminal nodes to the reduced parasitics. The reduced circuit is re-simulated with Spectre and its performance is compared to the original simulation.

Most of the circuits are reduced with the default options of SparseRC: matching only the 0'th and 1'st order multi-port admittance moments at $s = 0$ and also *without* employing additional fill-tracking options. For some examples, the functionality of advanced options within SparseRC is demonstrated, such as: additional moment or pole matching, or additional fill-monitoring actions per subnet. All results presented next are based on partitioning with nested dissection (for additional experiments with Mondriaan hypergraph partitioning see Appendix 4.6.2).

SparseRC was implemented in Matlab (version R2007a); all reduction experiments were performed on a Linux (Ubuntu) machine with 3.9GRAM main memory and two Intel(R) Core(TM)2 Duo, 2.4GHz, CPUs. All Spectre simulations were run on a Linux (Redhat) machine with 16GRAM main memory and six Intel(R) Xeon(R) X5460, 3.1GHz, CPUs.

Table 4.1: *Reduction for various netlists with SparseRC vs. PACT (SparseRC with no partitioning). Sim. time: Spectre [16] netlist simulation time, Total red. time: partitioning plus reduction time*

Net	Type	n_i	#R	#C	Sim. time (s)	Total red. time (s)	Error (RMS)	Type	n_i	#R	#C	Sim. time (s)	Total red. time (s)	Error (RMS)
1. Transmission line (**TL**) $p = 22$	Orig.	3231	5892	3065	0.51	-	-	Orig.	3231	5892	3065	0.51	-	-
	SparseRC	21	165	592	0.01	0.39	2.5e–4	PACT	0	40	231	0.01	0.1	6.1e–4
	Red. rate	99.35%	97.20%	80.69%	51 X	-		Red. rate	100%	99.32%	92.46%	51 X	-	
2. Low noise amplifier (**LNA**) $p = 79$	Orig.	29806	53285	12025	1525	-		Orig.	29806	53285	12025	1525	-	
	SpRC-dp	50	308	2608	13	53.4	4e–4	PACT	0	111	1080	9	0.92	1e–3
	Red. rate	99.8%	99.4%	78.3%	117 X	-		Red. rate	100%	99.79%	91.02%	169 X	-	
3. Mixer_3 (**MX3**) $p = 110$	Orig.	757	1393	2353	1900	-		Orig.	757	1393	2353	1900	-	
	SpRC-mf	40	137	810	754	1.09	1.6e–7	PACT	0	111	1117	831	0.03	9.4e–7
	Red. rate	94.72%	90.17%	65.58%	2.5 X	-		Red. rate	100%	92.03%	52.53%	2.3 X	-	
4. Interconnect structure (**IS**) $p = 646$	Orig.	16216	26413	173277	1730	-		Orig.	16216	26413	173277	1730	-	
	SpRC	208	10228	72748	18	3.49	3.5e–5	PACT	17	6844	66758	17	3.56	5.6e–4
	Red. rate	98.72%	61.28%	58.02%	96 X	-		Red. rate	99.9%	74.09%	61.47%	101 X	-	
5. Mixer_7 (**MX7**) $p = 66$	Orig.	67	119	194	7.15	-		Orig.	67	119	194	7.15	-	
	SpRC-mf	11	59	187	5.51	0.32	1.2e–8	PACT	0	56	308	9.12	0.01	7.1e–8
	Red. rate	83.58%	50.42%	3.61%	1.3 X	-		Red. rate	100%	52.94%	–58.76%	0.8 X	-	
6. Receiver (**RX**) $p = 15171$	Orig.	788081	1416454	1961224	NA	-		Orig.	788081	1416454	1961224	NA	-	
	SpRC	6719	95162	845699	520	589.4	NA	PACT	NA	NA	NA	NA	NA	NA
	Red. rate	99.15%	93.28%	56.88%	∞	-		Red. rate	NA	NA	NA	NA	-	
7. Phase-locked loop (**PLL**) $p = 4041$	Orig.	377433	593786	555553	NA	-		Orig.	377433	593786	555553	NA	-	
	SpRC	3905	46499	312351	3710	151.54	NA	PACT	NA	NA	NA	NA	NA	NA
	Red.	98.97%	92.17%	43.78%	∞	-		Red. rate	NA	NA	NA	NA	-	
8. **Filter** $p = 5852$	Orig.	32140	47718	123696	1140	-		Orig.	32140	47718	123696	1140	-	
	SpRC	6882	31995	155011	700	185.06	1.4e–5	PACT	0	414500	5927790	$>24h$	68.5	NA
	Red. rate	78.59%	32.95%	−25.32%	1.6 X	-		Red. rate	100%	−768.64%	−4692%	NA	-	

4.4.1 General results

Table 4.1 collects the main SparseRC reduction results for various multi-terminal netlists obtained from real chip designs, and compares the results obtained with PACT[7]. Each block row consists of a netlist example with the corresponding number of terminals p (the same before and after reduction). For each netlist, the sparsity before and after reduction are recorded here as the number of circuit elements, i.e. #R, #C. The reduction rate (*Red. rate*) shows the percentage reduction for the corresponding column quantity. For instance the percentage reduction in internal nodes n_i is: $P_n = \frac{100(n_{i_{Orig.}} - n_{i_{SpRC}})}{n_{i_{Orig.}}}$, similarly for #$R$, #$C$. The *Red. rate* in simulation time is computed as a speed-up factor: $\frac{\text{Sim. time}_{Orig}}{\text{Sim. time}_{SpRC}}$ (similarly for PACT). The approximation error is measured as the root-mean-square (RMS) value of the difference between the signals of the original and the reduced circuit. Several simulations are performed, including: AC, noise, transient, (quasi) periodic steady state, (quasi) s-parameter, usually a combination of these for each circuit depending on the underlying application. Due to space limitations we can only present some of the simulation figures, however the simulation timings recorded in the tables represent the sum of all analysis types performed for one circuit. The error RMS value is recorded for one analysis only, but is representative of all analysis types performed for that circuit.

For most examples, excellent reduction rates (above 80%) in the number of internal nodes n_i were obtained. The n_i internal nodes of the reduced model are precisely the internal nodes which, if otherwise eliminated, would have introduced too much fill-in. They are the separator nodes identified automatically from partitioning (plus, where suitable, some additional internal nodes identified from fill monitoring operations). With n_i thus preserved, very good reduction rates were obtained in the number of circuit elements: mostly above 60% reduction in resistors and above 50% for capacitors. The effect of reducing internal nodes as well as the number of circuit elements is revealed by significant speed-ups (mostly above $2X$) attained when simulating the reduced circuits instead of the original. Even more, for the largest examples (netlists **RX, PLL**) simulation was only possible after reduction, as the original simulations failed due to insufficient CPU and memory resources. In addition, the reduction times recorded in Table 4.1 show that these reduced netlists were obtained efficiently.

Reduction without partitioning

For comparison, results for SparseRC without partitioning (essentially, PACT) are also recorded in Table 4.1. PACT amounts to running the SparseRC Algorithm 1 of Sect. 4.3.4 with $N=1$ in line 2. The results reveal the strength of SparseRC *especially* when reducing challenging circuits with very large node and terminal numbers (e.g., nets 6, 7, 8).

[7] For simplicity, "PACT" is used here only as a short term to denote SparseRC reduction without partitioning; the full PACT methodology [56] also includes more advanced analysis such as pole matching.

First, the computational advantages of partitioning for very large netlists are revealed through examples **PLL** and **RX**, for which an unpartitioned PACT projection could not even be computed. For the smaller examples, the PACT reduction times are smaller than the SparseRC ones, indicating that partitioning is not necessary. Second, partitioning and the preservation of separator nodes improves the sparsity of the reduced models. This is confirmed by the **Filter** and **MX7**, where the unpartitioned approach resulted in dense reduced netlists which were slower to simulate than the originals (the effect would be the same for **PLL** and **RX**).

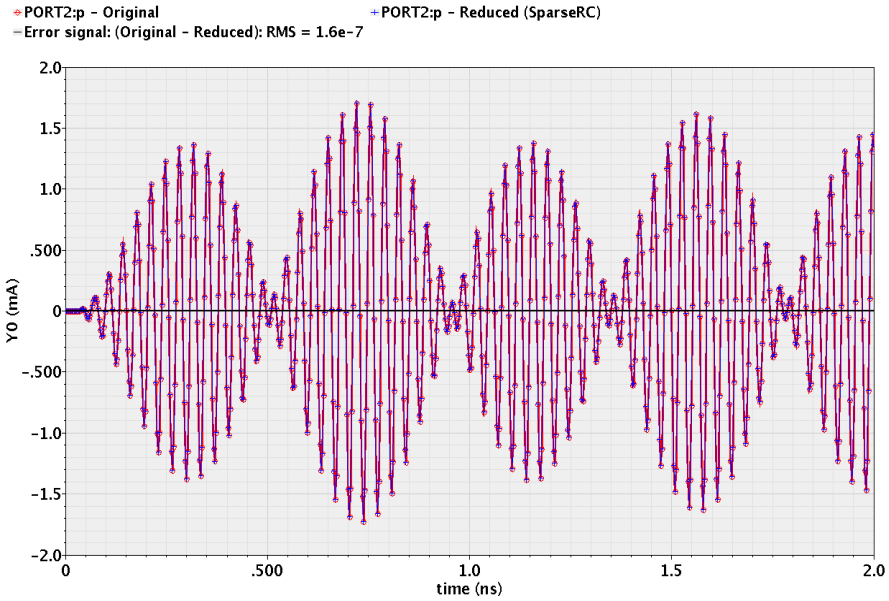


Figure 4.4: ***MX3****. Transient simulation of the original (red) and SparseRC (blue) overlap. The error signal (black) has an RMS value of* $1.6 \cdot 10^{-7} A$.

Next, selected simulation results are presented. The **MX3** net comes from a mixer circuit. Here, the SparseRC model was obtained by further re-ordering the partitioned circuit matrices (obtained via NESDIS) with CAMD and by keeping track of fill-in during the block-wise reduction process. The transient simulation in Fig. 4.4 shows that the original and SparseRC curves are indistinguishable.

Improving accuracy at higher frequencies

Two examples in particular demonstrate possibilities for improving the approximation accuracy beyond matching the 0'th and 1'st moment at $s = 0$, as described in Sect. 4.3.3:

The low noise amplifier (LNA) Net **LNA** from Table 4.1 is part of a low noise amplifier circuit (C45 technology), and was reduced in 53.4 seconds to a SparseRC-dp[8] model which was 99% faster to simulate. SparseRC-dp denotes a reduced model obtained from a partition into $N = 3$ components where, for each component, the two default moments at $s = 0$ are matched plus $\sim$ eight dominant poles of the internal contribution $\mathbf{Y}_R^{'}$. These dominant poles were computed with the subspace accelerated dominant pole method [79]. In Fig. 4.5 the simulation results are shown of the noise analysis for the original, SparseRC-dp and PACT models. The effect of improving the SparseRC response with the additional dominant poles is visible in Fig. 4.5. This was also confirmed in transient simulation, which gave an RMS error of $4{\cdot}10^{-4}$ for SparseRC-dp, smaller than $1{\cdot}10^{-3}$ for PACT (see Table 4.1).

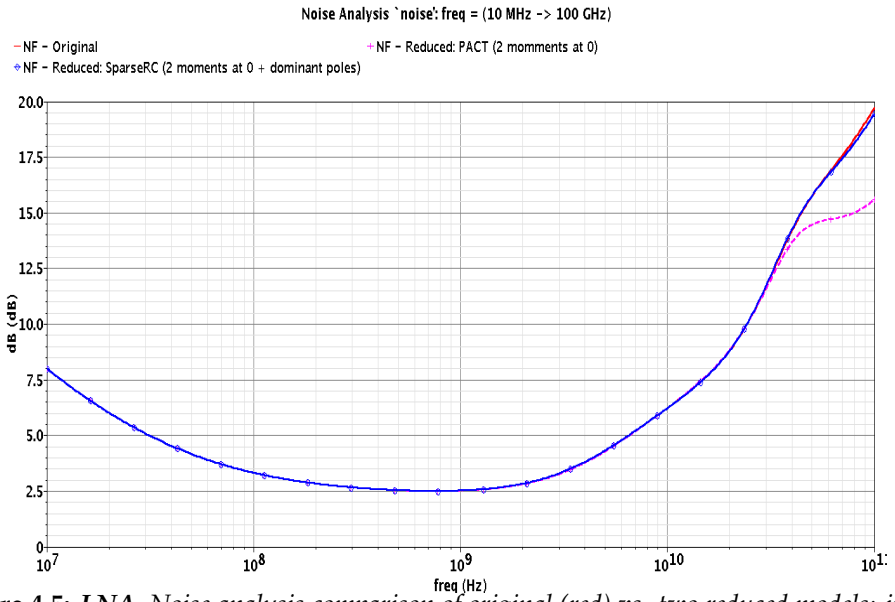


Figure 4.5: ***LNA****. Noise analysis comparison of original (red) vs. two reduced models: in blue, SparseRC-dp (moment patching at $s = 0$ with additional dominant poles), and in pink, PACT with default moment matching at $s = 0$.*

Two port *RC* **transmission line** A uniform *RC* transmission line with two terminals (one node at the beginning, and one node at the end of the line) is considered, with a cascade of 10000 *R*-*C* sections. The circuit was partitioned into $N = 2$ subnets (and one separator block). Two reduced SparseRC models were computed: without and with additional matching at higher moments. The latter model was computed with the projection (4.47), which matched a combination of moments of the $\mathbf{Y}_R^{'}$ term per subnet: two moments at 0, one at $s = 10^{10}$ and one at $s = 10^{14}$ was chosen. The bode magnitude plot of the frequency response for the original and the two reduced models is shown in

[8] "dp" is short for "dominant pole"

Fig. 4.6. The behavior at higher frequencies is indeed approximated more accurately for the reduced model which preserves additional moments.

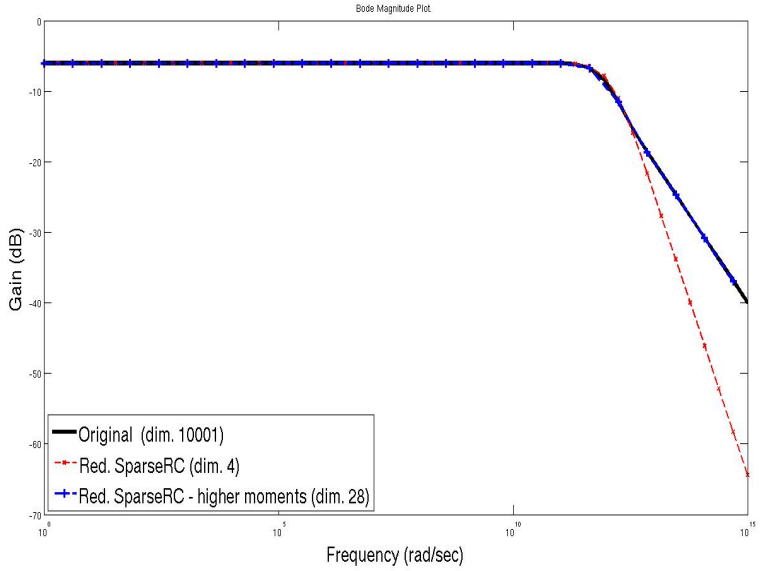


Figure 4.6: ***RCtline****: Bode plot for original and two reduced SparseRC models: by matching the first two admittance moments at $s = 0$ only (red) and by matching in addition higher moments from the $\mathbf{Y}_R^{'}$ term per subnet (blue).*

4.4.2 Advanced comparisons

Nets **PLL** (phase-locked-loop) and **Filter**, two of the largest and most challenging netlists from Table 4.1 are analyzed in detail in Table 4.2. The purpose of the analysis is three-fold: (1) the advantages of SparseRC over existing methodologies are revealed, (2) the effects of various partitioning sizes and of additional reorderings are shown, and (3) possible limitations and improvement directions for SparseRC are identified.

The PLL

Performing an original simulation was unfeasible for this circuit, as it failed due to insufficient CPU and memory resources even on a larger machine, but reduction makes the simulation possible. The original $\mathcal{G}$ and $\mathcal{C}$ matrices, reordered and partitioned in BBD form are shown in Fig. 4.7. The borders are clearly visible, collecting the separator nodes that will be preserved along with the terminals. The reduced matrices retain the BBD structure and are sparse, as seen in Fig. 4.8.

Table 4.2: *Advanced comparisons for very large examples. N is the number of subnets, AvgN size is the average size of a subnet, AvgS size/Avg-p size are the average number of separator nodes/terminals per subnet, Avg. red. time is the average reduction time per subnet*

Net	Type	n_i	P_{n_i} (%)	#R	P_R (%)	#C	P_C (%)	Sim. time (s)	Sim. speed-up (X)	Red. time (s)	Partition time (s)	N #part.	AvgN size	AvgS size	Avg-p size	Avg. red. time (s)
7. **PLL** $p = 4041$	Orig.	377433	-	593786	-	555553	-	NA	-	-	-	-	-	-	-	-
	SpRC$_c$	3905	98.97	46499	92.17	312351	43.78	689	∞	116	36	255	2949	32	29	0.86
	SpRC$_f$	12861	96.59	56582	90.47	243350	56.20	656	∞	1199	394	2423	304	11	3	0.93
	PartMOR	5392	98.57	138628	76.6	209835	62.22	836	∞	1410	559	229	–	–	–	–
	PACT	NA	NA	NA	NA	NA	NA	NA	NA	NA	-	-	-	-	-	-
	SIP-mf	NA	NA	NA	NA	NA	NA	NA	NA	$> 24h$	-	-	-	-	-	-
8. **Filter** $p = 5852$	Orig.	32140	-	47718	-	123696	-	1140	-	-	-	-	-	-	-	-
	SpRC	6882	78.59	31995	32.95	155011	−25.32	700	1.63 X	163	22.60	2065	29	8	4	0.15
	SpRC-mf	16729	47.95	32760	31.35	116272	6.00	783	1.46 X	1237	19.44	2065	29	8	4	1.19
	PACT	0	100	414500	−768.64	5927790	−4692	$> 24h$	NA	69	-	-	-	-	-	-
	SIP-mf	30935	3.75	46399	2.76	123135	0.45	892	1.28 X	11648	-	-	-	-	-	-

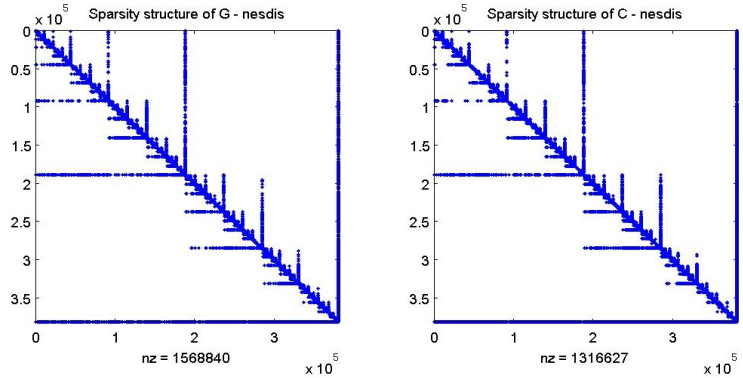


Figure 4.7: ***PLL****: re-ordered $\mathcal{G}$ and $\mathcal{C}$ in BBD form after NESDIS partitioning (dimension $n + p = 381474$ nodes).*

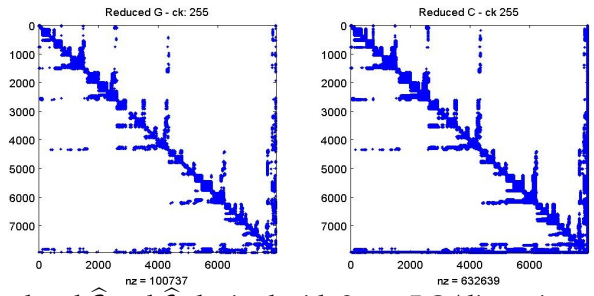


Figure 4.8: ***PLL****: reduced $\widehat{\mathcal{G}}$ and $\widehat{\mathcal{C}}$ obtained with SparseRC (dimension $n + p = 7946$ nodes). The BBD structure is retained and the matrices remain sparse.*

Two SparseRC reduced models were computed: SpRC_c and SpRC_f, based on a coarse and fine NESDIS partitioning respectively, with the relevant statistics shown in Table 4.2. Both reduced models achieved excellent reduction rates in internal nodes and circuit elements, and were fast to simulate. After a coarse partitioning, the reduction time was smaller than after the fine partitioning due to less computational overhead in forming the reduced matrices per subnet. The SpRC_f reduced model however was faster to simulate than SpRC_c, possibly due to the fact that, although larger in numbers of nodes than SpRC_c, SpRC_f has fewer circuit elements. Determining the appropriate balance between preserved internal nodes n_i and sparsity, and its influence on simulation time remains to be further studied. A direct PACT reduction is immediately dismissed, due to the prohibitive computational and density considerations. An SIP-based reduced model was attempted, but the fill-in monitoring actions were too expensive ($>$24 hours).

Comparison with PartMOR For the **PLL**, statistics of a reduced PartMOR model are also included, thanks to the authors of [70]. Compared to PartMOR, SpRC_c has fewer internal nodes and circuit elements. The SparseRC models were faster to simulate, and also obtained in a shorter partitioning and reduction time. Although there is no original simulation to compare the reductions against, Fig. 4.9 shows an AC simulation wave-

form for the two SparseRC models, and PartMOR. SpRC$_c$ and SpRC$_f$ overlap, confirming that the accuracy of SparseRC is robust to changes in the partition strategy, due to guaranteed local *and* global moment matching. The reduced PartMOR model was determined by local matching of the 0'th and 1'st moments at DC, however with no guarantee of matched moments for the recombined network (personal communication with PartMOR [70] authors, March 14, 2011). Thus, since the accuracy of PartMOR is dependent on the number of partitions, finer partitioning would likely be needed to reach what we infer are the correct waveforms produced by SparseRC. While PartMOR offers advantages in other respects, such as guaranteed positive elements, these results motivate a more thorough investigation into combining the strengths of both methods in the future.

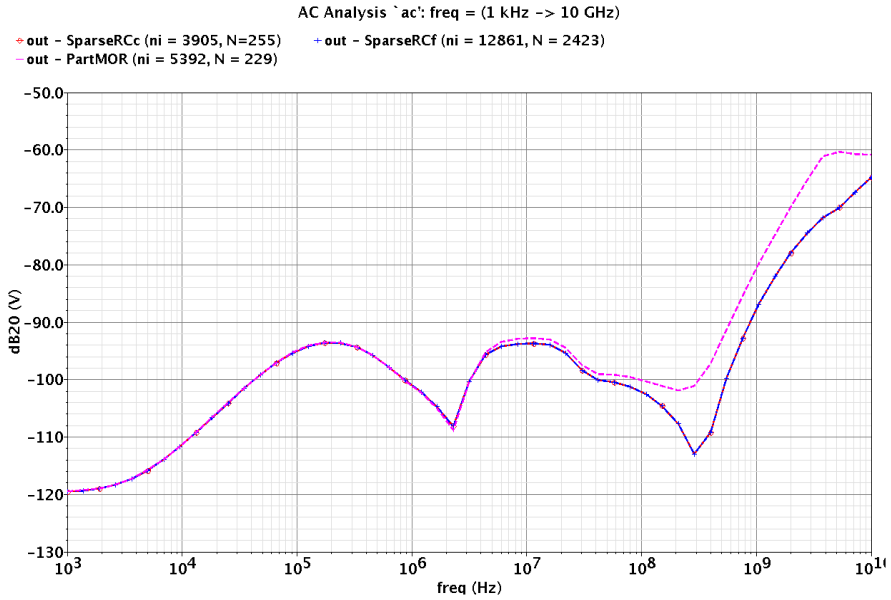


Figure 4.9: ***PLL****. AC analysis of reduced models: SpRC$_c$ (red) and SpRC$_f$ (blue) are overlapping as expected. PartMOR (magenta) deviates slightly.*

The Filter

This netlist is more challenging due to its large ratio $\frac{p}{n_i} > 10^{-1}$. Two SparseRC reduced models were computed: SpRC, based on NESDIS partitioning alone, and SpRC-mf[9], where after NESDIS partitioning a CAMD based re-ordering was applied on each subnet and additional internal nodes were preserved via fill-in monitoring operations. In both cases a fine partitioning into $N = 2065$ components was needed to distribute ter-

[9] "mf" stands from "minimum fill"

minals into Avg-p=4 terminals per component. Although the SpRC-mf has much fewer circuit elements than SpRC, it takes longer to simulate, due to the presence of many internal nodes n_i. The fill-monitoring operations inside SpRC-mf also make the reduction time significantly longer than for SpRC. The PACT reduced model is the smallest in dimension (has no preserved internal nodes) but extremely dense and useless in simulation (the re-simulation was stopped after 24 hours). An SIP reduced model was also computed: CAMD reordering and fill-monitoring actions were applied on the original netlist's graph to determine the reduced model with minimum fill. The result is shown in Fig. 4.11, where the minimum fill point is identified after the elimination of the first 1200 internal nodes. Therefore the optimal SIP reduced circuit achieves a much smaller reduction in internal nodes and elements than SparseRC, and is slower to simulate. Also, the SIP reduction time was much larger than SparseRC. The AC analysis comparison of SparseRC and SIP match with the original, as shown in Figure 4.10. This further strengthens the advantages of partitioning: aside from enhancing sparsity by preserving separator nodes, it also makes fill-monitoring actions cheaper to perform. In summary, SparseRC achieves the best trade-off in terms of accuracy, dimension, sparsity, reduction time and re-simulation time.

Finally, the **Filter** example reveals several directions for improving the SparseRC methodology. It indicates that other reduction transformations and/or further sparsification methods may be appropriate for circuits with many more capacitors than resistors. One option could be to postprocess the reduced system by dropping negative capacitances as in TICER [84], at the price of losing the two moment matching property. It was also seen that circuits with large $\frac{p}{n}$ ratios are the most difficult to partition and reduce with a satisfactory sparsity level. This could be resolved with the help of partitioners which could directly control the distribution of terminals, and remains for further research [47].

4.5 Concluding remarks

SparseRC is presented, a robust and efficient reduction strategy for large *RC* circuits with many terminals. It efficiently reduces testcases where traditional model reduction fails, due to either the large dimension of the problem at hand, or the density of the final reduced circuit. Developed on the divide and conquer principle, SparseRC uses graph-partitioning and fill-reducing node reorderings to separate a network into minimally connected components. These are reduced individually with an appropriate update of the interconnections among them. This guarantees that two multi-port admittance moments are matched for the entire net irrespective of the partitioning size or strategy. SparseRC reduced circuits contain fewer nodes and circuit elements compared to conventional MOR results. As challenging industrial testcases show, significant speed-ups are obtained when re-simulating SparseRC reduced models in place of the original circuits.

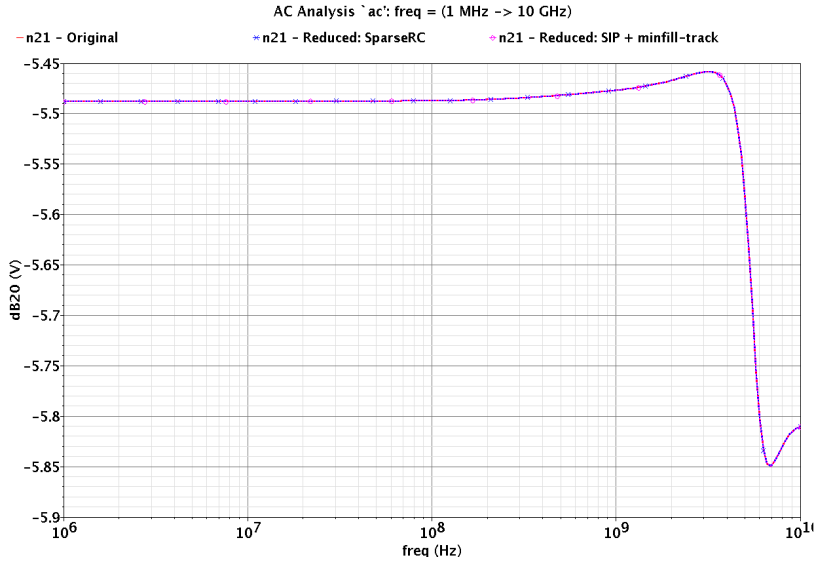


Figure 4.10: ***Filter****. AC analysis of original (red), reduced SparseRC model (blue), and reduced SIP model with minimum fill-track (magenta) match perfectly.*

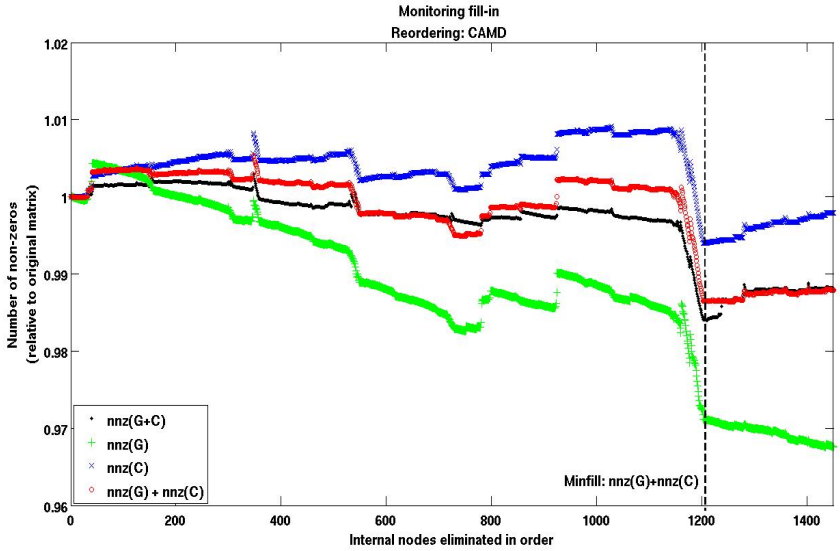


Figure 4.11: ***Filter****. Determining the dimension of the SIP reduced model from CAMD reordering and node-wise elimination. The Minimum fill point is reached after eliminating the first* 1200 *of the* 32140 *internal nodes (for clarity only the first* 1400 *internal nodes are shown).*

4.6 Appendix

4.6.1 Reflection on the partition-based RC reduction alternatives

Two reduction approaches for RC networks were proposed in this thesis: SparseRC of this chapter and the one based on the strongly connected components of $\mathcal{G}$ from Chapter 3 [we refer to it as SCC($\mathcal{G}$)-based here]. At this point, it is appropriate to reconsider for which types of circuits the two alternatives are most suitable.

The first indicator is the number of terminals p of the circuit: if p is within a few hundreds (what we call "small"), then the SCC($\mathcal{G}$)-based reduction is usually sufficient to achieve good reduction rates in both the number of internal nodes and the number of circuit elements. Recall from Theorem 3.3.1 that reduction based on SCC($\mathcal{G}$) partitioning is mathematically equivalent to that of eliminating all internal nodes up-front. Hence, when p is small, the fill generated in the $p \times p$ reduced matrices is not a major concern. Nevertheless, when the circuit has a very large number of internal nodes (exceeding hundreds of thousands), the SCC($\mathcal{G}$) partitioning brings computational benefits as smaller subnets are reduced individually. This partitioning also gives a natural solution for computing path resistances between network terminals (see Sect. 3.3.3). In short, the SCC($\mathcal{G}$)-based reduction is recommended for circuits with a very large number of internal nodes, and a small number of terminals.

When p exceeds thousands, the fill generated in the $p \times p$ block from eliminating all internal nodes becomes non-negligible. Thus, fill-creating nodes must be identified and kept in the reduced model so that its sparsity is improved. This is achieved with SparseRC based on the partitioning of the entire R, C topology (rather than on the R-topology only as is done with the SCC($\mathcal{G}$)-based reduction). So, especially when the number of internal nodes is also large (hundreds of thousands), SparseRC is the method of choice. One limitation of SparseRC is the reduction of networks with many more capacitors than resistors. This could be improved with the help of better partitioners (see Chapter 7) and a reducing projection based on the $\mathcal{C}$ matrix rather than the $\mathcal{G}$ matrix (this would place more emphasis on capturing the behavior at higher frequencies rather than on low frequencies).

The second indicator is the ratio of terminals p to internal nodes n_i in the network. In general, as for most reduction methods, if $\frac{p}{n_i}$ is small (less than $\frac{1}{10}$), good reduction rates in internal nodes can be obtained. Hence, if $\frac{p}{n_i}$ is small, the SCC($\mathcal{G}$)-based reduction is recommended when p is small, while SparseRC is recommended when p is large. On the other hand when $\frac{p}{n_i}$ is high, only few internal nodes can be eliminated. This is a challenge both for the SCC($\mathcal{G}$)-based reduction and for SparseRC. In this case, more advanced partitioners would help to find a more appropriate balance between the number of eliminated internal nodes and sparsity. Further considerations on partitioning are given in Chapter 7.

4.6.2 Additional experiments

Additional RC experiments

Additional SparseRC reduction experiments, based on NESDIS graph partitioning, were performed on the multi-terminal netlists in Table 4.3. The expectations regarding the improvements in sparsity and resimulation time, and the excellent approximation quality, are again confirmed by the SparseRC results. For the **SymmRC1** circuit, challenging due to the presence of symmetries in the original design, the resimulation plots are shown in Fig. 4.12. The SparseRC reduced circuit captures the symmetry and matches the response of the original circuit.

Table 4.3: *Reduction for various netlists with SparseRC. Sim. time: Spectre [16] netlist simulation time, Total red. time: partitioning plus reduction time*

Net	Type	n_i	#R	#C	Sim. time (min)	Total red. time (s)	N # parts
9. Symmetric	Orig.	15965	24636	94840	52	-	-
RC (**SymmRC1**)	SparseRC	1463	13868	152445	19	6.2	37
$p = 1383$	Red. rate	90.84%	43.71%	-60.74%	2.7 X	-	-
10. Symmetric	Orig.	12634	22862	119351	-	-	
RC (**SymmRC2**)	SparseRC	805	5503	82170	-	4	15
$p = 732$	Red. rate	93.63%	75.93%	31.15%	-	-	
11. Shift register	Orig.	29748	43550	95521	$2h\,36min$	-	
(**Sband**)	SpRC-mf	9837	19685	88998	$1h12min$	609	157
$p = 5003$	Red. rate	66.93%	54.80%	6.83%	2.2 X	-	

Using Mondriaan hypergraph partitioning

To demonstrate that SparseRC can accommodate other partitioning algorithms than nested dissection, Tables 4.4, 4.5, and 4.6 collect the reduction results based on Mondriaan hypergraph partitioning [96]. With Mondriaan the resulting subnets are allowed to differ in size (by specifying an *Imbalance* factor), while with nested dissection the decomposition gives equal-sized partitions. The higher the *Imbalance* (the maximum value is 1), the more are the subnets generated by Mondriaan allowed to differ in size. This feature allows to experiment with different partition sizes and their impact on sparsity. In the following tables, each block row corresponds to the experiments performed for one netlist. "SpRC$_B$" corresponds to the reduction statistics obtained after a balanced decomposition, and "SpRC$_I$" after an imbalanced one. The quantities recorded in each column are the same as those of Table 4.2, with the difference that column "Max-N size" denotes the size of the largest subnet resulting from partitioning, and "Max-p size" the number of terminals falling in the maximum subnet. There are no significant dif-

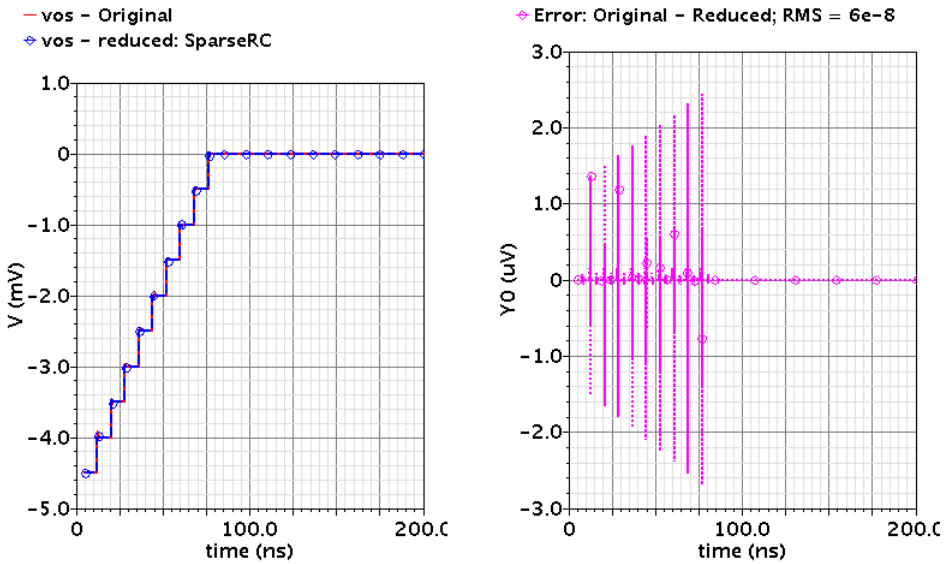


Figure 4.12: ***SymmRC1****. Transient analysis of the original (red) and the SparseRC reduced (blue) circuit shows perfect match. The RMS error is also small,* $6 \cdot 10^{-8} V$.

ferences in sparsity after reduction based on Mondriaan partitioning compared to the experiments based on nested dissection (Tables 4.1 and 4.2), however one experiment revealed an interesting insight. In particular, for the **IS** netlist, the imbalanced partition (SpRC_I) achieved significantly better reduction rates in the number of elements than the balanced partition, and also better than with nested dissection (see Table 4.1). This confirms that enforcing balanced decompositions may not be optimal for the multi-terminal reduction problem, and that more advanced partitioning criteria could significantly improve sparsity rates (this topic is addressed separately in Chapter 5). Partitioning time with Mondriaan was slower than with nested dissection especially for the large netlists (see the **PLL** and **Filter** examples); this is expected as Mondriaan is a hypergraph partitioner, while nested dissection is based on the Metis graph partitioner.

Table 4.4: *Reduction statistics with Mondriaan partitioning: part* 1

Net	Type	n_i	P_{n_i} (%)	#R	P_R (%)	#C	P_C (%)	Red. time (s)	Partition time (s)	N #part.	Imbal.	Max-N size	Max-p size
	Orig.	3231	-	5892	-	3065	-	-	-	-	-	-	-
1. **TL**	SpRC_B	12	99.63	64	98.91	401	86.92	0.11	0.80	3	0.03	1667	13
$p = 22$	SpRC_I	17	99.47	132	97.76	592	80.69	0.12	0.79	3	0.90	3058	17
	PACT	0	100.00	40	99.32	231	92.46	0.08	-	-	-	-	-
	Orig.	29806	-	53285	-	12025	-	-	-	-	-	-	-
2. **LNA**	SpRC_B	263	99.12	3507	93.42	6804	43.42	2.69	3.11	3	0.03	15591	11
$p = 79$	SpRC_I	23	99.92	164	99.69	1400	88.36	1.15	2.95	3	0.90	28464	63
	PACT	0	100.00	111	99.79	1080	91.02	1.10	-	-	-	-	-
	Orig.	757	-	1393	-	2353	-	-	-	-	-	-	-
3. $\mathbf{MX}_3$	SpRC_B	44	94.19	150	89.23	954	59.46	0.07	0.71	15	0.03	115	6
$p = 110$	SpRC_I	24	96.83	138	90.09	1050	55.38	0.07	0.75	15	0.90	168	6
	PACT	0	100.00	111	92.03	1117	52.53	0.02	-	-	-	-	-
	Orig.	16216	-	26413	-	173277	-	-	-	-	-	-	-
4. **IS**	SpRC_B	334	97.94	10374	60.72	77271	55.41	3.12	5.63	3	0.03	8321	331
$p = 22$	SpRC_I	111	99.32	5151	80.50	39006	77.49	2.48	6.43	3	0.90	14206	503
	PACT	17	99.90	6844	74.09	66758	61.47	4.47	-	-	-	-	-
	Orig.	67	-	119	-	194	-	-	-	-	-	-	-
5. $\mathbf{MX}_7$	SpRC_B	16	76.12	66	44.54	197	-1.55	0.04	0.29	15	0.03	19	12
$p = 66$	SpRC_I	8	88.06	59	50.42	188	3.09	0.04	0.29	15	0.90	32	16
	PACT	0	100.00	56	52.94	308	-58.76	0.00	-	-	-	-	-

Table 4.5: *Reduction statistics with Mondriaan partitioning: part* 2

Net	Type	n_i	P_{n_i} (%)	#R	P_R (%)	#C	P_C (%)	Red. time (s)	Partition time (s)	N #part.	Imbal.	Max-N size	Max-p size
7. **PLL** $p = 4041$	Orig.	377433	-	593786	-	555553	-	-	-	-	-	-	-
	SpRC_I	3984	98.94	57037	90.39	587112	−5.68	184.89	4303.00	63	0.90	24702	79
	SpRC_B	8220	97.82	62312	89.51	405331	27.04	82.18	2280.29	127	0.03	7840	8
	PACT	-	-	-	-	-	-	NA	-	-	-	-	-
8. **Filter** $p = 5852$	Orig.	32140	-	47718	-	123696	-	-	-	-	-	-	-
	SpRC_B	4955	84.58	34833	27.00	171211	−38.41	41.09	27.32	511	0.03	639	91
	SpRC_I	3971	87.64	35586	25.42	179612	−45.20	82.16	29.50	511	0.90	390	26
	SpRC_B	6869	78.63	31844	33.27	153313	−23.94	79.13	34.23	1023	0.03	678	97
	SpRC_I	5603	82.57	34920	26.82	165926	−34.14	69.30	35.61	1023	0.90	400	74
	PACT	0	100.00	414500	−768.64	5927790	−4692.22	67.39	-	-	-	-	-

Table 4.6: *Reduction statistics with Mondriaan partitioning: part* 3

Net	Type	n_i	P_{n_i} (%)	#R	P_R (%)	#C	P_C (%)	Red. time (s)	Partition time (s)	N #part.	Imbal.	Max-N size	Max-p size
	Orig.	27962	-	50102	-	9773	-	-	-	-	-	-	-
9. $\mathbf{LNA}_3$	SpRC_B	348	98.76	5884	88.26	16863	−72.55	3.37	3.61	3	0.03	14736	7
$p = 79$	SpRC_I	25	99.91	173	99.65	1333	86.36	1.09	3.29	3	0.90	26757	60
	PACT	0	100.00	111	99.78	1065	89.10	0.95	-	-	-	-	-
	Orig.	33818	-	60569	-	12132	-	-	-	-	-	-	-
10. $\mathbf{LNA}_2$	SpRC_B	296	99.12	5570	90.80	16113	−32.81	3.47	3.76	3	0.03	17814	7
$p = 75$	SpRC_I	36	99.89	222	99.63	1466	87.92	1.32	3.58	3	0.90	31977	52
	PACT	0	100.00	93	99.85	944	92.22	1.12	-	-	-	-	-
	Orig.	480	-	721	-	3132	-	-	-	-	-	-	-
11. **IF2cHB**	SpRC_B	149	68.96	424	41.19	3417	−9.10	0.10	0.56	31	0.03	53	15
$p = 252$	SpRC_I	140	70.83	402	44.24	3458	−10.41	0.14	0.59	31	0.90	90	34
	PACT	8	98.33	443	38.56	7457	−138.09	0.05	-	-	-	-	-
	Orig.	606	-	1116	-	1432	-	-	-	-	-	-	-
12. **PPFHB**	SpRC_B	234	61.39	975	12.63	1851	−29.26	0.31	0.68	127	0.03	31	5
$p = 420$	SpRC_I	242	60.07	952	14.70	1838	−28.35	0.48	1.03	255	0.90	22	22
	PACT	0	100.00	7429	−565.68	19926	−1291.48	0.09	-	-	-	-	-
	Orig.	217	-	362	-	1068	-	-	-	-	-	-	-
13. **TIAHB**	SpRC_B	95	56.22	265	26.80	1261	−18.07	0.09	0.37	31	0.03	44	8
$p = 142$	SpRC_I	75	65.44	284	21.55	1395	−30.62	0.08	0.39	31	0.90	40	11
	PACT	1	99.54	489	−35.08	3515	−229.12	0.03	-	-	-	-	-

Chapter 5

Reduction of multi-terminal RLC networks

The reduction of multi-terminal *RLC* circuits is analyzed in the context of partitioning. Comparisons between reduction in first vs. second-order form are provided, and their potential for implementation in a partitioning setup is discussed. Based on the second-order form, the BBD-based reduction of *RLC* networks is derived, and a post-processing procedure is proposed which allows the reduced model to be synthesized and re-used successfully with a circuit simulator. Advantages and limitations within this framework are identified, and directions for future research are provided.

5.1 Introduction

Taking the methodology of Chapter 4 one step further, this chapter investigates the extension of partition-based reduction to large *RLC* networks. As for *RC* networks, the aim is to reduce large, multi-terminal *RLC* netlists efficiently, so that their re-use in a simulation setup is possible and more efficient than an original simulation, this at little accuracy loss. Achieving these goals simultaneously is more difficult for *RLC* than for *RC* networks, as this chapter reveals. The content here-in thus provides a skeleton for partition-based *RLC* reduction which identifies the main partition-reduction steps, the limitations encountered and the possibilities to overcome them.

Among the popular approaches to multi-terminal *RLC* reduction are PRIMA [71] and its structure preserving followers SPRIM [27], and SAPOR [66]. These methods construct the reducing projection as a (block)-Krylov subspace, and preserve the passiv-

ity of the circuit via congruence transforms. The structure preserving versions allow for voltage and current variables to be reduced separately, among the main goals being to (a) preserve terminal connectivity and (b) obtain a reduced model which can be cast back into an *RLC* netlist. While these methods may work well for relatively small-sized networks, their performance is limited when applied to large networks with many terminals. While on one hand, computing the Krylov subspaces for the original system is either too expensive or impossible, on the other the resulting matrices are prohibitively dense. Furthermore, the synthesis envisioned after an SPRIM or SAPOR projection is not as straightforward as it seems, due to the fact that the reduced models may contain singularities which in turn cause simulation failures. Here, a framework for multi-terminal *RLC* reduction is derived, with emphasis on achieving netlists that are re-usable in simulations. As a continuation of Chapter 4 which developed sparse *RC* reduction based on the PACT method [56], the starting point here is the extension of PACT to *RLC* circuits, namely [57]. In particular the potential of [56] in a partition-based context is studied. In Sect. 5.2 the setup for unpartitioned reduction is described based on two system formulations: in the first- and second-order form respectively. For reasons there explained, the second-order form is chosen for partition-based reduction, which is presented in Sect. 5.3. A demonstrative numerical example is provided in Sect. 5.4, and Sect. 5.5 concludes.

5.2 Reduction without partitioning

Consider the impedance-based Modified Nodal Analysis (MNA) [37] representation of an $RLC(k)$ circuit:

$$\left(\begin{bmatrix} \mathcal{G} & \mathcal{E}^T \\ -\mathcal{E} & \mathbf{0} \end{bmatrix} + s \begin{bmatrix} \mathcal{C} & \mathbf{0} \\ \mathbf{0} & \mathcal{L} \end{bmatrix} \right) \begin{bmatrix} \mathbf{v} \\ \mathbf{i}_L \end{bmatrix} = \begin{bmatrix} \mathcal{B} \\ \mathbf{0} \end{bmatrix} \mathbf{u}(s), \tag{5.1}$$

where $\mathbf{v} \in \mathbb{R}^{n_v}$ are the voltage unknowns (voltage drops measured at all nodes in the circuit), $\mathbf{i}_L \in \mathbb{R}^{n_L}$ are the currents through inductors, $\mathcal{G} \in \mathbb{R}^{(n_v \times n_v)}$ is the conductance matrix, $\mathcal{C} \in \mathbb{R}^{(n_v \times n_v)}$ is the capacitance matrix, $\mathcal{L} \in \mathbb{R}^{(n_L \times n_L)}$ is the inductance matrix (a diagonal with the inductor values if there are no mutual inductances, otherwise the mutuals appear as off-diagonal entries). $\mathcal{E} \in \mathbb{R}^{(n_v \times n_L)}$ is the incidence matrix which determines the topological connections for the inductors. The inputs are $\mathbf{u} \in \mathbb{R}^p$, the current injections into the p terminals of the circuit, and $\mathcal{B} \in \mathbb{R}^{(n_v \times p)}$ is the incidence matrix of current injections (the i'th column of $\mathcal{B}$ is the i'th unit vector for each input $i = 1 \dots p$). Let also n be the number of internal nodes, so that $n_v = n + p$.

The starting point for multi-terminal *RLC* reduction is [57], which reduces (5.1) in first-order form using so called *split congruence transforms*. The *splitting* projects separately the voltage from the current unknowns, essentially preserving the structure of the *RLC* circuit, and is necessary to ensure that passivity is preserved [55, 57]. The same concept of splitting the reducing projection is seen in SPRIM [27], the Krylov framework [8] and

RLCSYN [93]. In Sect. 5.2.1 we present the main derivations for reduction in the first-order form based on split congruence transforms, according to [57]. Then, in Sect. 5.2.2, the new reduction in second-order form is presented which, as will be seen, does no longer require split transformations.

5.2.1 First-order form

In (5.1), let the unknown voltages in (5.16) be split into *selected nodes* $\mathbf{v}_S$ to be preserved (p terminals and if desired additional m internal nodes), and internal nodes to be eliminated $\mathbf{v}_R$. This dictates the following structure of (5.1):

$$\left(\left[\begin{array}{c|cc} \mathcal{G}_S & \mathcal{G}_K^T & \mathcal{E}_S^T \\ \hline \mathcal{G}_K & \mathcal{G}_R & \mathcal{E}_R^T \\ -\mathcal{E}_S & -\mathcal{E}_R & \mathbf{0} \end{array}\right] + s\left[\begin{array}{c|cc} \mathcal{C}_S & \mathcal{C}_K^T & \mathbf{0} \\ \hline \mathcal{C}_K & \mathcal{C}_R & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathcal{L} \end{array}\right]\right)\left[\begin{array}{c} \mathbf{v}_S \\ \hline \mathbf{v}_R \\ \mathbf{i}_L \end{array}\right] = \left[\begin{array}{c} \mathcal{B}_S \\ \hline \mathbf{0} \\ \mathbf{0} \end{array}\right]\mathbf{u}(s) \tag{5.2}$$

We make the analogy between the form (5.2) and the congruence-based framework of [57]. Notice that the bottom right corner block of $\mathcal{G}$ in (5.2) is unsymmetric. The derivations of [57] are based on congruence transformations applied to symmetric matrices, hence a sign change is required for the last equation of (5.2), which gives:

$$\left(\left[\begin{array}{c|cc} \mathcal{G}_S & \mathcal{G}_K^T & \mathcal{E}_S^T \\ \hline \mathcal{G}_K & \mathcal{G}_R & \mathcal{E}_R^T \\ \mathcal{E}_S & \mathcal{E}_R & \mathbf{0} \end{array}\right] + s\left[\begin{array}{c|cc} \mathcal{C}_S & \mathcal{C}_K^T & \mathbf{0} \\ \hline \mathcal{C}_K & \mathcal{C}_R & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & -\mathcal{L} \end{array}\right]\right)\left[\begin{array}{c} \mathbf{v}_S \\ \hline \mathbf{v}_R \\ -\mathbf{i}_L \end{array}\right] = \left[\begin{array}{c} \mathcal{B}_S \\ \hline \mathbf{0} \\ \mathbf{0} \end{array}\right]\mathbf{u}(s) \quad \Leftrightarrow \tag{5.3}$$

$$\left(\left[\begin{array}{c|c} \mathbf{G}_P & \mathbf{G}_C^T \\ \hline \mathbf{G}_C & \mathbf{G}_I \end{array}\right] + s\left[\begin{array}{c|c} \mathbf{C}_P & \mathbf{C}_C^T \\ \hline \mathbf{C}_C & \mathbf{C}_I \end{array}\right]\right)\left[\begin{array}{c} \mathbf{x}_P \\ \hline \mathbf{X}_I \end{array}\right] = \left[\begin{array}{c} \mathbf{b}_P \\ \hline \mathbf{0} \end{array}\right]\mathbf{u}(s) \tag{5.4}$$

The matrix blocks pertaining to (5.3) are redefined as in (5.4), as to directly identify the analogy with [57, equation (12)]. The internal unknowns then become $\mathbf{X}_I = \begin{bmatrix} \mathbf{v}_i \\ \mathbf{i}_L \end{bmatrix}$, and contain both voltage and current variables. This will later require a structure preserving reduction (also called *split*), as to guarantee passivity preservation [55]. In [57] the following congruence transformation:

$$\mathbf{X} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{W} & \mathbf{I} \end{bmatrix}, \text{ where } \mathbf{W} = -\mathbf{C}_I^{-1}\mathbf{C}_C \tag{5.5}$$

is applied to (5.4) as to zero-out $\mathbf{C}_C$:

$$\left(\left[\begin{array}{c|c} \mathbf{G}_P' & \mathbf{G}_C'^T \\ \hline \mathbf{G}_C' & \mathbf{G}_I \end{array}\right] + s\left[\begin{array}{c|c} \mathbf{C}_P' & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{C}_I \end{array}\right]\right)\left[\begin{array}{c} \mathbf{x}_P' \\ \hline \mathbf{X}_I \end{array}\right] = \left[\begin{array}{c} \mathbf{b}_P \\ \hline \mathbf{0} \end{array}\right]\mathbf{u}(s) \tag{5.6}$$

with:

$$\mathbf{G}_P^{'} = \mathbf{G}_P + \mathbf{W}^T\mathbf{G}_I\mathbf{W} + \mathbf{W}^T\mathbf{G}_C + \mathbf{G}_C^T\mathbf{W}, \;\; \mathbf{G}_C^{'} = \mathbf{G}_C + \mathbf{G}_I\mathbf{W} \tag{5.7}$$

$$\mathbf{C}_P^{'} = \mathbf{C}_P + \mathbf{C}_C^T\mathbf{C}_I^{-1}\mathbf{C}_C. \tag{5.8}$$

At this point, it is worth pausing to answer the following questions: why is the transformation to zero-out the capacitance connection matrix $\mathbf{C}_C$ needed and why not zero-out $\mathbf{G}_C$, as was done for *RC* circuits in Chapter 4? There are two main reasons: passivity preservation and moment matching. First [57, Theorems 1, 2] show that, for *RLC* reduction to preserve passivity, the underlying projection must be split, as to preserve the structure dictated by the voltage and current variables [see $\mathbf{v}_R$, $\mathbf{i}_L$ in (5.3)]. As explained in [55, Sect. 6.3], a transformation of the form $-\mathbf{G}_I^{-1}\mathbf{G}_C$ which would zero-out $\mathbf{G}_C$ from (5.4) is *not naturally split*, due to the presence of the $\mathcal{E}_R$ blocks in $\mathbf{G}_I$. However, since $\mathbf{C}_I$ is block diagonal, the transformation (5.5) to zero-out $\mathbf{C}_C$ *is naturally split*, and maintains the necessary structure for passivity preserving reduction. Furthermore, based on the assumption that the model has the form (5.6), namely that the capacitive connection blocks are zero, [57, Theorem 4] shows that reducing the internal blocks of (5.6) via a structure-preserving (split) Krylov projection matches moments of the original model system (5.4). This is outlined next.

The multiport admittance response of (5.6) describes the circuit behavior at the ports, and is obtained by eliminating the $\mathbf{X}_I$ variables from the second equation of (5.6) and replacing them in the first. This is expressed as follows:

$$\mathbf{Y}^{'}(s)\mathbf{x}_P(s) = \mathbf{b}_p\mathbf{u}(s), \quad \text{where :} \tag{5.9}$$

$$\mathbf{Y}^{'}(s) = \underbrace{\left(\mathbf{G}_P^{'} + s\mathbf{C}_P^{'}\right)}_{\mathbf{Y}_P^{'}(s)} \underbrace{-\mathbf{G}_C^{'\,T}\left(\mathbf{G}_I + s\mathbf{C}_I\right)^{-1}\mathbf{G}_C^{'}}_{\mathbf{Y}_I^{'}(s)} \tag{5.10}$$

At this point, [57, Theorem 4] enables the construction of a reduced model which matches multi-port admittance moments of the original system (5.6) via a congruence transform:

$$\begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{V}^T \end{bmatrix} \left(\left[\begin{array}{c|c} \mathbf{G}_P^{'} & \mathbf{G}_C^{'\,T} \\ \hline \mathbf{G}_C^{'} & \mathbf{G}_I \end{array}\right] + s \left[\begin{array}{c|c} \mathbf{C}_P^{'} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{C}_I \end{array}\right] \right) \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{V} \end{bmatrix}, \tag{5.11}$$

which reduces (5.6) to:

$$\left(\left[\begin{array}{c|c} \mathbf{G}_P^{'} & \widehat{\mathbf{G}}_C^T \\ \hline \widehat{\mathbf{G}}_C & \widehat{\mathbf{G}}_I \end{array}\right] + s \left[\begin{array}{c|c} \mathbf{C}_P^{'} & \mathbf{0} \\ \hline \mathbf{0} & \widehat{\mathbf{C}}_I \end{array}\right] \right) \left[\begin{array}{c} \mathbf{x}_P^{'} \\ \hline \widehat{\mathbf{X}}_I \end{array}\right] = \left[\begin{array}{c} \mathbf{b}_P \\ \hline \mathbf{0} \end{array}\right] \mathbf{u}(s), \tag{5.12}$$

In [57, Theorem 4], it is proven that the reduced (5.12) matches $2m$ moments of (5.6) at s_0 if $\mathbf{A} = \mathbf{G}_I + s_0\mathbf{C}_I$ is non-singular, $\mathbf{G}_I$ and $\mathbf{C}_I$ are symmetric and $\mathbf{V}$ is a subspace of

linearly independent columns which satisfies:

$$\text{span}\left[\left(\mathbf{A}^{-1}\mathbf{C}_I\right)^i \mathbf{A}^{-1}\mathbf{G}_C^{'}\right] \subseteq \mathbf{V},\ \ i = 0 \ldots m-1. \tag{5.13}$$

From (5.12), the block matrices corresponding to port nodes $\mathbf{x}_P^{'}$ are preserved, and the internal matrices are reduced to: $\widehat{\mathbf{G}}_I = \mathbf{V}^T\mathbf{G}_I\mathbf{V}$, $\widehat{\mathbf{G}}_C = \mathbf{V}^T\mathbf{G}_C^{'}$, $\widehat{\mathbf{C}}_I = \mathbf{V}^T\mathbf{C}_I\mathbf{V}$. The reduced multi-port admittance response which characterizes (5.12) is then:

$$\widehat{\mathbf{Y}}(s) = \underbrace{(\mathbf{G}_P^{'} + s\mathbf{C}_P^{'})}_{\mathbf{Y}_P^{'}(s)} - \underbrace{\widehat{\mathbf{G}}_C^T(\widehat{\mathbf{G}}_I + s\widehat{\mathbf{C}}_I)^{-1}\widehat{\mathbf{G}}_C}_{\widehat{\mathbf{Y}}_I(s)}. \tag{5.14}$$

Two important conditions must be satisfied as to ensure moment matching and passivity preservation: (a) $\mathbf{G}_I$ and $\mathbf{C}_I$ must be symmetric and (b) $\mathbf{V}$ must be split according to structure of the internal matrices from (5.3). In this sense, the $\mathbf{V}$ which satisfies (5.13) with equality is nothing but an SPRIM [27] projection of the internal admittance response $\mathbf{Y}_I^{'}(s)$ from (5.10). The splitting of $\mathbf{V}$ allows the reduced model (5.12) to be expressed as a counterpart of (5.3):

$$\left(\left[\begin{array}{c|cc} \mathcal{G}_S^{'} & \widehat{\mathcal{G}}_K^T & \widehat{\mathcal{E}}_S^T \\ \hline \widehat{\mathcal{G}}_K & \widehat{\mathcal{G}}_R & \widehat{\mathcal{E}}_R^T \\ \widehat{\mathcal{E}}_S & \widehat{\mathcal{E}}_R & \mathbf{0} \end{array}\right] + s\left[\begin{array}{c|cc} \mathcal{C}_S^{'} & \mathbf{0} & \mathbf{0} \\ \hline \mathbf{0} & \widehat{\mathcal{C}}_R & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & -\widehat{\mathcal{L}} \end{array}\right]\right)\left[\begin{array}{c} \mathbf{v}_S \\ \hline \widehat{\mathbf{v}}_R \\ -\widehat{\mathbf{i}}_L \end{array}\right] = \left[\begin{array}{c} \mathcal{B}_S \\ \hline \mathbf{0} \\ \mathbf{0} \end{array}\right]\mathbf{u}(s) \tag{5.15}$$

System (5.15) being in a symmetric form can be unstamped into an *RC* equivalent netlist (where negative *R* and *C* values are also generated).

Difficulties with the first-order form

Several difficulties have been encountered in practice with the first-order reduction setup.

1. In practice, it was observed that the reduced block $\left[\begin{array}{c} \widehat{\mathcal{E}}_S^T \\ \widehat{\mathcal{E}}_R^T \end{array}\right]$ in (5.15) does not always have linearly independent columns. This in turn caused failures in computing the DC solution during the the re-simulation of the synthesized reduced model. Hence a post-processing step would be required to ensure linear independence, this without affecting the structure of the input/output matrix. We will see how this is easily achieved from the second-order form.
2. If partitioning is employed, the internal matrix $\mathbf{G}_I = \left[\begin{array}{cc} \mathcal{G}_R & \mathcal{E}_R^T \\ \mathcal{E}_R & \mathbf{0} \end{array}\right]$ resulting per sub-block may become singular (even when for the unpartitioned system it is not

singular). This happens for instance when nodes to which inductors are connected are promoted as separators (these nodes become terminals and, when shorted together, may cause inductor loops [57]). It can also happen that from partitioning, the $\mathcal{E}_R^T$ block per subnet has columns of zero, which is undesirable. All these structural side-effects from partitioning make it thus difficult to construct a Krylov subspace (5.13) per subnet, especially in the context of matching moments at DC. A more detailed discussion on this issue follows Sect. 5.3.1.

Next, a framework is derived for multi-terminal reduction in the second-order form, through which item 1. finds a solution, while item 2. can be resolved in certain scenarios.

5.2.2 Second-order form

By eliminating the current variables $\mathbf{i}_L$ from the second equation of (5.1) and replacing them in the first, it is possible to express (5.1) in second-order form:

$$\left(\mathcal{G} + s\mathcal{C} + \frac{1}{s}\Gamma\right)\mathbf{v} = \mathcal{B}, \quad \text{with} \quad \Gamma = \mathcal{E}^T\mathcal{L}^{-1}\mathcal{E}. \tag{5.16}$$

The second-order form (5.16) has the following main advantages: (a) congruence transforms on (5.16) automatically preserve passivity without need of splitting, as all unknowns are voltages (b) it defines the circuit graph topology in an intuitive way, giving a natural avenue for partition-based reduction and (c) some of the numerical limitations encountered from partitioning in the first-order form can be avoided.

In addition to the resistive and capacitive topology determined by $\mathcal{G}$ and $\mathcal{C}$ respectively (as was the case for the *RC* circuits), Γ dictates the topology of inductors. Hence, the graph associated with the *RLC* circuit is given by the non-zero pattern $G := nzp(\mathcal{G} + \mathcal{C} + \Gamma)$. The following derivations are based on the second-order form. As usual, the unknown voltages in (5.16) are split into *selected nodes* $\mathbf{v}_S$ to be preserved (p terminals and if desired m additional internal nodes[1]), and internal nodes to be eliminated $\mathbf{v}_R$, revealing the following structure:

$$\left(\begin{bmatrix} \mathcal{G}_R & \mathcal{G}_K \\ \mathcal{G}_K^T & \mathcal{G}_S \end{bmatrix} + s\begin{bmatrix} \mathcal{C}_R & \mathcal{C}_K \\ \mathcal{C}_K^T & \mathcal{C}_S \end{bmatrix} + \frac{1}{s}\begin{bmatrix} \Gamma_R & \Gamma_K \\ \Gamma_K^T & \Gamma_S \end{bmatrix}\right)\begin{bmatrix} \mathbf{v}_R \\ \mathbf{v}_S \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathcal{B}_S \end{bmatrix}\mathbf{u}(s). \tag{5.17}$$

For the same moment matching considerations of Sect. 5.2.1, a congruence transformation is applied first which zeroes-out the $\mathcal{C}_K$ connections (this transformation can be

[1] These are for instance the separator nodes that are revealed from partitioning, as in Chapter 4

easily shown as analogous to the first-order case of zeroing-out $\mathbf{C}_C$). Let:

$$\mathcal{X} \;=\; \left[\begin{array}{c|c} \mathbf{I} & \mathcal{W} \\ \mathbf{0} & \mathbf{I} \end{array}\right],\; \mathcal{W} = -\mathcal{C}_R^{-1}\mathcal{C}_K. \tag{5.18}$$

$\mathcal{X} \in \mathbb{R}^{n_v \times n_v}$ is only a transformation (*no reduction*) of (5.17) into:

$$\mathcal{G}^{'} = \mathcal{X}^T\mathcal{G}\mathcal{X},\; \mathcal{C}^{'} = \mathcal{X}^T\mathcal{C}\mathcal{X},\; \Gamma^{'} = \mathcal{X}^T\Gamma\mathcal{X},\; \mathcal{B}^{'} \;=\; \mathcal{X}^T\mathcal{B} \;\Leftrightarrow \tag{5.19}$$

$$\left(\left[\begin{array}{cc} \mathcal{G}_R & \mathcal{G}_K^{'} \\ \mathcal{G}_K^{'T} & \mathcal{G}_S^{'} \end{array}\right] + s\left[\begin{array}{cc} \mathcal{C}_R & \mathbf{0} \\ \mathbf{0}^T & \mathcal{C}_S^{'} \end{array}\right] + \frac{1}{s}\left[\begin{array}{cc} \Gamma_R & \Gamma_K^{'} \\ \Gamma_K^{'T} & \Gamma_S^{'} \end{array}\right]\right)\left[\begin{array}{c} \mathbf{v}_R \\ \mathbf{v}_S^{'} \end{array}\right] \;=\; \left[\begin{array}{c} \mathbf{0} \\ \mathcal{B}_S \end{array}\right]\mathbf{u}(s), \tag{5.20}$$

where:

$$\mathcal{G}_S^{'} \;=\; \mathcal{G}_S + \mathcal{W}^T\mathcal{G}_R\mathcal{W} + \mathcal{W}^T\mathcal{G}_K + \mathcal{G}_K^T\mathcal{W},\quad \mathcal{G}_K^{'} = \mathcal{G}_K + \mathcal{G}_R\mathcal{W} \tag{5.21}$$

$$\Gamma_S^{'} \;=\; \Gamma_S + \mathcal{W}^T\Gamma_R\mathcal{W} + \mathcal{W}^T\Gamma_K + \Gamma_K^T\mathcal{W},\quad \Gamma_K = \Gamma_K + \Gamma_R\mathcal{W} \tag{5.22}$$

$$\mathcal{C}_S^{'} \;=\; \mathcal{C}_S - \mathcal{C}_K^T\mathcal{C}_R^{-1}\mathcal{C}_K,\quad \mathcal{C}_K^{'} = \mathcal{C}_K + \mathcal{C}_R\mathcal{W} = \mathbf{0}. \tag{5.23}$$

As with the first-order form let $\mathcal{Y}^{'}(s)$ be the multi-port admittance response of (5.20), obtained by eliminating the $\mathbf{v}_R$ unknowns:

$$\mathcal{Y}^{'}(s) \;=\; \underbrace{\left(\mathcal{G}_S^{'} + s\mathcal{C}_S^{'} + \frac{1}{s}\Gamma_S^{'}\right)}_{\mathcal{Y}_S^{'}(s)} - \underbrace{\left(\mathcal{G}_K^{'} + \frac{1}{s}\Gamma_K^{'}\right)^T\left(\mathcal{G}_R + s\mathcal{C}_R + \frac{1}{s}\Gamma_R\right)^{-1}\left(\mathcal{G}_K^{'} + \frac{1}{s}\Gamma_K^{'}\right)}_{\mathcal{Y}_R^{'}(s)} \tag{5.24}$$

Towards understanding how (5.20) can be reduced by moment matching, we make the following important analogy between the first and second-order formulation:

Lemma 5.2.1 *The multiport admittance $\mathcal{Y}^{'}(s)$ given by (5.24) which characterizes the second-order system (5.20) is equal to $\mathbf{Y}^{'}(s)$ given by (5.10) which pertains to the first-order form (5.6).*

Proof 5.2.1 *Without detailing all the derivations, based on the assignments (5.3)-(5.4), the form of (5.7)-(5.8), and of (5.21)-(5.23), it can be shown that:*

$$\mathbf{Y}_P^{'}(s) \;=\; \mathcal{G}_S^{'} + s\mathcal{C}_S^{'} \tag{5.25}$$

$$\mathbf{Y}_I^{'}(s) \;=\; -\frac{1}{s}\Gamma_S^{'} + \underbrace{\left(\mathcal{G}_K^{'} + \frac{1}{s}\Gamma_K^{'}\right)^T\left(\mathcal{G}_R + s\mathcal{C}_R + \frac{1}{s}\Gamma_R\right)^{-1}\left(\mathcal{G}_K^{'} + \frac{1}{s}\Gamma_K^{'}\right)}_{\mathcal{Y}_R^{'}(s)} \tag{5.26}$$

$$\Rightarrow \mathbf{Y}^{'}(s) \;=\; \mathbf{Y}_P^{'}(s) - \mathbf{Y}_I^{'}(s) = \mathcal{Y}_S^{'}(s) - \mathcal{Y}_R^{'}(s) = \mathcal{Y}^{'}(s) \qquad \blacksquare$$

Lemma 5.2.1 allows reduction to be performed in the second-order form, similarly to

that in the first-order form. This amounts to reducing the $\mathcal{Y}_R^{'}(s)$ term of (5.24). Let for the moment $\mathcal{Q} \in \mathbb{R}^{n_v \times k}$ be a transformation which reduces the matrix blocks from (5.17) corresponding to the internal nodes $\mathbf{v}_R$. As in the first-order case, the following congruence transform applied to (5.20):

$$\left[\begin{array}{cc} \mathcal{Q}^T & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{array}\right] \left(\left[\begin{array}{cc} \mathcal{G}_R & \mathcal{G}_K^{'} \\ \mathcal{G}_K^{'T} & \mathcal{G}_S^{'} \end{array}\right] + s \left[\begin{array}{cc} \mathcal{C}_R & \mathbf{0} \\ \mathbf{0}^T & \mathcal{C}_S^{'} \end{array}\right] + \frac{1}{s} \left[\begin{array}{cc} \Gamma_R & \Gamma_K^{'} \\ \Gamma_K^{'T} & \Gamma_S^{'} \end{array}\right] \right) \left[\begin{array}{cc} \mathcal{Q} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{array}\right] \tag{5.27}$$

reduces (5.20) to:

$$\left(\left[\begin{array}{cc} \widehat{\mathcal{G}}_R & \widehat{\mathcal{G}}_K \\ \widehat{\mathcal{G}}_K^T & \mathcal{G}_S^{'} \end{array}\right] + s \left[\begin{array}{cc} \widehat{\mathcal{C}}_R & \mathbf{0} \\ \mathbf{0}^T & \mathcal{C}_S^{'} \end{array}\right] + \frac{1}{s} \left[\begin{array}{cc} \widehat{\Gamma}_R & \widehat{\Gamma}_K \\ \widehat{\Gamma}_K^T & \Gamma_S^{'} \end{array}\right] \right) \left[\begin{array}{c} \widehat{\mathbf{v}}_R \\ \mathbf{v}_S^{'} \end{array}\right] = \left[\begin{array}{c} \mathbf{0} \\ \mathcal{B}_S \end{array}\right] \mathbf{u}(s), \tag{5.28}$$

where (5.21)-(5.23) hold and:

$$\widehat{\mathcal{G}}_R = \mathcal{Q}^T \mathcal{G}_R \mathcal{Q},\ \widehat{\mathcal{G}}_K = \mathcal{Q}^T \mathcal{G}_K^{'},\ \widehat{\Gamma}_R = \mathcal{Q}^T \Gamma_R \mathcal{Q},\ \widehat{\Gamma}_K = \mathcal{Q}^T \Gamma_K^{'},\ \widehat{\mathcal{C}}_R = \mathcal{Q}^T \mathcal{C}_R \mathcal{Q}. \tag{5.29}$$

The multi-port admittance function for the reduce model (5.28) is similarly expressed by eliminating the $\widehat{\mathbf{v}}_R$ unknowns:

$$\widehat{\mathcal{Y}}(s) = \left(\mathcal{G}_S^{'} + s\mathcal{C}_S^{'} + \frac{1}{s}\Gamma_S^{'}\right) - \left(\widehat{\mathcal{G}}_K + \frac{1}{s}\widehat{\Gamma}_K\right)^T \left(\widehat{\mathcal{G}}_R + s\widehat{\mathcal{C}}_R + \frac{1}{s}\widehat{\Gamma}_R\right)^{-1} \left(\widehat{\mathcal{G}}_K + \frac{1}{s}\widehat{\Gamma}_K\right) = \tag{5.30}$$

$$= \mathcal{Y}_S^{'}(s) - \widehat{\mathcal{Y}}_R(s). \tag{5.31}$$

As regards the appropriate construction of $\mathcal{Q}$ from (5.27), one possibility is to obtain it via a SAPOR [66] reduction of $\mathcal{Y}_R^{'}(s)$ (similarly to how $\mathbf{V}$ is constructed as an SPRIM projection of $\mathbf{Y}_I^{'}(s)$ in the first-order form of Sect. 5.2.1). Via SAPOR, a linearization of $\mathcal{Y}_R^{'}(s)$ is needed to form an associated Krylov subspace (for a given expansion point s_0), from which the reducing projection $\mathcal{Q}$ is obtained (details on the construction of the SAPOR projection $\mathcal{Q}$ are given in [66]). One apparent hurdle for SAPOR is the frequency dependency s inside the input-output term $\mathcal{G}_K^{'} + \frac{1}{s}\Gamma_K^{'}$. Nevertheless, it is shown next that the linearization proposed in [66] eliminates the dependency on s when computing the actual Krylov subspace, so that the usual SAPOR reduction applies.

Linearization and SAPOR

Following [66] we outline the linearization of $\mathcal{Y}_R^{'}(s)$ from (5.24). This then leads to the formation of the Krylov subspace from which the reducing projection $\mathcal{Q}$ is extracted. Consider $\mathcal{Y}_R^{'}(s)$ separately from (5.24). $\mathcal{Y}_R^{'}(s)$ then represents the transfer function of

the following internal system:

$$\begin{cases} \left(\mathcal{G}_R+s\mathcal{C}_R+\frac{1}{s}\Gamma_R\right)\mathbf{v}_R(s) &= \left(\mathcal{G}_K^{'}+\frac{1}{s}\Gamma_K^{'}\right)\mathbf{u}_R(s) \\ \mathbf{y}_R(s) &= \left(\mathcal{G}_K^{'}+\frac{1}{s}\Gamma_K^{'}\right)^T\mathbf{v}_R(s) \end{cases} \tag{5.32}$$

where $\mathbf{u}_R(s)$ are the inputs of the internal system and $\mathbf{y}_R(s)$ are the outputs. Multiplying (5.32) with s, one obtains:

$$\left(s^2\mathcal{C}_R+s\mathcal{G}_R+\Gamma_R\right)\mathbf{v}_R(s) = \left(s\mathcal{G}_K^{'}+\Gamma_K^{'}\right)\mathbf{u}_R(s). \tag{5.33}$$

Shifting (5.33) by setting $s = s_0 + \sigma$, one obtains the shifted system:

$$(\sigma^2\mathcal{C}_R + \sigma\mathcal{D}_R + \mathcal{K}_R)\mathbf{v}_R(\sigma) = (\mathbf{B}_0 + \sigma\mathbf{B}_1)\mathbf{u}_R(\sigma), \tag{5.34}$$

where:

$$\mathcal{D}_R = \mathcal{G}_R + 2s_0\mathcal{C}_R,\ \mathcal{K}_R = s_0^2\mathcal{C}_R + s_0\mathcal{G}_R + \Gamma_R,\ \mathbf{B}_0 = s_0\mathcal{G}_K^{'} + \Gamma_K^{'},\ \mathbf{B}_1 = \mathcal{G}_K^{'}, \tag{5.35}$$

and s_0 is such that $\mathcal{K}_R$ is non-singular.

System (5.34) is linearized as follows. Let the intermediate variable $\mathbf{z}(\sigma)$ be such that:

$$\sigma\mathcal{C}_R\mathbf{v}_R(\sigma) + \mathbf{z}(\sigma) = \mathbf{B}_1\mathbf{u}_R(\sigma) \tag{5.36}$$

Replacing (5.36) in (5.34) results in:

$$\begin{aligned} -\sigma\mathbf{z}(\sigma) + (\sigma\mathcal{D}_R + \mathcal{K}_R)\mathbf{v}_R(\sigma) &= \mathbf{B}_0\mathbf{u}_R(\sigma) \Leftrightarrow \\ \left(\mathbf{I} + \sigma\mathcal{K}_R^{-1}\mathcal{D}_R\right)\mathbf{v}_R(\sigma) - \sigma\mathcal{K}_R^{-1}\mathbf{z}(\sigma) &= \mathcal{K}_R^{-1}\mathbf{B}_0\mathbf{u}_R(\sigma). \end{aligned} \tag{5.37}$$

Combining (5.37) with (5.36) leads to the following linearized system:

$$\underbrace{\left(\begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} - \sigma \begin{bmatrix} -\mathcal{K}_R^{-1}\mathcal{D}_R & \mathcal{K}_R^{-1} \\ -\mathcal{C}_R & \mathbf{0} \end{bmatrix}\right)}_{:=\mathcal{I}-\sigma\mathcal{T}} \begin{bmatrix} \mathbf{v}_R(\sigma) \\ \mathbf{z}(\sigma) \end{bmatrix} = \begin{bmatrix} \mathcal{K}_R^{-1}\mathbf{B}_0 \\ \mathbf{B}_1 \end{bmatrix}\mathbf{u}_R(\sigma). \tag{5.38}$$

Notice that the input matrix of the linearized system (5.38) has no dependency on the frequency σ [$\mathbf{B}_0$ and $\mathbf{B}_1$ as in (5.35) are independent from σ]. Based on (5.38), the Block SAPOR method [66] proposes to compute an orthonormal basis $\mathcal{Q}$ which spans a desired number of block moments of $\mathbf{v}_R(\sigma)$. This is then used to project the internal system (5.32):

$$\left(\mathcal{Q}^T\mathcal{G}_R\mathcal{Q}+s\mathcal{Q}^T\mathcal{C}_R\mathcal{Q}+\frac{1}{s}\mathcal{Q}^T\Gamma_R\mathcal{Q}\right)\mathcal{Q}^T\mathbf{v}_R(s) = \left(\mathcal{Q}^T\mathcal{G}_K^{'}+\frac{1}{s}\mathcal{Q}^T\Gamma_K^{'}\right)\mathbf{u}_R(s), \tag{5.39}$$

which are the same matrices underlying our reduced second-order system (5.28)-(5.31). Note also that if moments at $s_0 = 0$ are to be matched, then from (5.35) we have that

$\mathcal{K}_R = \Gamma_R$. From (5.38) this would require $\mathcal{K}_R = \Gamma_R$ to be invertible.

As in the first-order case, we revise the question as to why not zero-out $\mathcal{G}_K$ instead of $\mathcal{C}_K$ in (5.17). Indeed, since in the second-order form the current variables $\mathbf{i}_L$ are eliminated, the transformation (5.18) preserves the passive form automatically via congruence *without any splitting*. Hence, contrary to the first-order case, having $\mathcal{W} = -\mathcal{G}_R^{-1}\mathcal{G}_K$ in (5.18) is justified as far as passivity is concerned. Although the option to zero-out $\mathcal{G}_K$ is not completely ruled-out, two words of caution are mentioned for this scenario. (a) The input-output matrices of $\mathcal{Y}_R^{'}(s)$ would have the form $s\mathcal{C}_K^{'} + \frac{1}{s}\Gamma_K^{'}$, in which case the linearization described above would not eliminate the dependency on s therein. A possible direction to resolve this could rest in [10], which derives moment matching projections for systems with input-output matrices that may depend on different powers of s. (b) Due to the analogy between the first and second-order admittance responses from Lemma 5.2.1 (both based on $\mathbf{C}_C$ and $\mathcal{C}_K$ zeroed-out) the reduction in the second-order from inherits the moment matching properties from the first-order form. Whether/which moments would be matched by zeroing-out $\mathcal{G}_K$ remains an open question. The same arguments could be raised in case Γ_K is zeroed-out. While the PACT-based approach [55, 57] analyzed here needs the zeroing-out of $\mathcal{C}_K$ to match moments, an alternative to avoid this altogether would be to reduce each subnet with SPRIM [27] in the first-order form, or with the second-order Arnoldi-based methods [9, 66, 89] in the second-order form.

Synthesis from the second-order form

Writing the reduced system (5.28) in compact form:

$$\left(\widehat{\mathcal{G}} + s\widehat{\mathcal{C}} + \frac{1}{s}\widehat{\Gamma}\right)\widehat{\mathbf{v}} = \widehat{\mathcal{B}}\mathbf{u}(s), \tag{5.40}$$

the final step is to synthesize (5.40) as a netlist to be re-used in simulation. While most publications related to the second-order form consider the reduced model (5.40) to be readily synthesizable, in practice this is not directly achieved. It seems natural to unstamp the matrices from (5.40) into the corresponding resistor, capacitor and inductor values respectively. While unstamping $\widehat{\mathcal{G}}$ and $\widehat{\mathcal{C}}$ poses no challenges, $\widehat{\Gamma}$ usually contains inductor loops. These in turn generate simulation failures: inductor loops cause short circuits when the circuit operates at DC (at DC, the capacitor acts as an open-circuit and the inductor as a short). These limitations have been reported in the RLCSYN [93]. There, a post-processing of $\widehat{\Gamma}$ is proposed, which diagonalizes $\widehat{\Gamma}_R$ as to ensure that inductors are connected only to ground, at the same time preserving the structure of the input matrix in (5.28). Due to this latter requirement, the procedure from [93] only holds when $\widehat{\Gamma}_K = \mathbf{0}$ (based on the assumption that $\Gamma_K = \mathbf{0}$) and is thus restricted to netlists where no inductors are connected to terminals. In practice however parasitic extracted *RLC* netlist have been encountered which do have inductors connected to terminals. Next, we show that a post-processing of $\widehat{\Gamma}$ which eliminates inductor loops and preserves the input-incidence structure can be applied in general also to circuits where inductors are

connected to ports.

Since Γ is real, symmetric, positive-semi-definite and $\widehat{\Gamma}$ is obtained from congruence transformations, $\widehat{\Gamma}$ will also be real, symmetric, positive-semi-definite. Hence a rank revealing Schur decomposition of $\Gamma = \mathcal{U}\mathcal{D}\mathcal{U}^T$ with $\mathcal{U}$ unitary and $\mathcal{D}$ diagonal can be partitioned into:

$$\widehat{\Gamma} = \begin{bmatrix} \overline{\mathcal{U}} & \mathcal{U}_0 \end{bmatrix} \begin{bmatrix} \overline{\mathcal{D}} & \mathbf{0} \\ \mathbf{0} & \mathcal{D}_0 \end{bmatrix} \begin{bmatrix} \overline{\mathcal{U}}^T \\ \mathcal{U}_0^T \end{bmatrix} = \overline{\mathcal{U}}\,\overline{\mathcal{D}}\,\overline{\mathcal{U}}^T + \mathcal{U}_0\mathcal{D}_0\mathcal{U}_0^T = \overline{\Gamma} + \Gamma_0, \tag{5.41}$$

where $\mathcal{D}_0$ contains the zero eigenvalues of $\widehat{\Gamma}$. Similarly to [93], eigenvalues smaller than a given tolerance may also be collected in the Γ_0 term, as these would correspond to over-large inductors. Hence the Γ_0 term is disregarded and $\overline{\Gamma}$ is retained. Defining $\overline{\mathcal{L}} = \overline{\mathcal{D}}^{-1}$, we have $\overline{\Gamma} = \overline{\mathcal{U}}\,\overline{\mathcal{L}}^{-1}\overline{\mathcal{U}}^T$, and notice the analogy with (5.16). It is thus possible to cast the reduced model (5.40) directly in the first-order form:

$$\left(\begin{bmatrix} \widehat{\mathcal{G}} & \overline{\mathcal{U}} \\ \overline{\mathcal{U}}^T & \mathbf{0} \end{bmatrix} + s \begin{bmatrix} \widehat{\mathcal{C}} & \mathbf{0} \\ \mathbf{0} & -\overline{\mathcal{L}} \end{bmatrix} \right) \begin{bmatrix} \widehat{\mathbf{v}} \\ -\widehat{\mathbf{i}}_{\overline{\mathcal{L}}} \end{bmatrix} = \begin{bmatrix} \widehat{\mathcal{B}} \\ \mathbf{0} \end{bmatrix} \mathbf{u}(s), \tag{5.42}$$

where the last equation contains a sign change as to obtain a symmetric representation. The symmetric reduced model (5.42) can now be unstamped into an *RC* equivalent circuit and re-simulated. As most netlists obtained from unstamping, the reduced netlist will also contain some negative *R* and *C* values, nevertheless it successfully passes the Spectre [16] simulation. Two important observations are made regarding (5.42). (a) $\overline{\mathbf{U}}$ has full column rank, in contrast to (5.15) which had rank deficiencies. Hence a DC solution of the synthesized netlist is found during the re-simulation. (b) The rank revealing decomposition of $\widehat{\Gamma}$ allows the conversion from the reduced second-order form (5.40) into (5.42) without spoiling the structure of the input matrix $\widehat{\mathcal{B}}$, and makes no assumptions about the structure of the original Γ. In this respect, the rank revealing procedure proposed here is a step further from that in [93].

Difficulties with the second-order form

The main difficulty with reduction in the second-order form arises when attempting to match moments of $\mathcal{Y}_R'(s)$ from (5.24) at $s = 0$. This requires Γ_R to be invertible, which is not always the case in practice. A more detailed discussion on this aspect is given in Sect. 5.3.1.

5.2.3 A parallel between 1st and 2nd order form

As a summary of Sect. 5.2.1 and Sect. 5.2.2, table 5.1 collects the main derivations steps for reduction in the first vs. second order form in the PACT framework and the analogies between the two. The reduction flow for the first or second order form respectively is shown in the corresponding column. The column "**link**" makes the connection between the operations on the first and and second order form at each step.

5.3 Partition-based reduction

As motivated at the beginning of Sect. 5.2.2, the second-order formulation provides an intuitive approach for reducing an *RLC* network by parts. To illustrate the partition-based reduction in the second-order form, let (5.16) be reordered according to the BBD structure[2], as was done for *RC* circuits in Chapter 4:

$$\left(\begin{bmatrix} \mathcal{G}_{11_R} & \mathcal{G}_{11_K} & \mathbf{0} & \mathbf{0} & \mathcal{G}_{13_R} \\ \mathcal{G}_{11_K}^T & \mathcal{G}_{11_S} & \mathbf{0} & \mathbf{0} & \mathcal{G}_{13_S} \\ \mathbf{0} & \mathbf{0} & \mathcal{G}_{22_R} & \mathcal{G}_{22_K} & \mathcal{G}_{23_R} \\ \mathbf{0} & \mathbf{0} & \mathcal{G}_{22_K}^T & \mathcal{G}_{22_S} & \mathcal{G}_{23_S} \\ \mathcal{G}_{13_R}^T & \mathcal{G}_{13_S}^T & \mathcal{G}_{23_R}^T & \mathcal{G}_{23_S}^T & \mathcal{G}_{33} \end{bmatrix} + s \begin{bmatrix} \mathcal{C}_{11_R} & \mathcal{C}_{11_K} & \mathbf{0} & \mathbf{0} & \mathcal{C}_{13_R} \\ \mathcal{C}_{11_K}^T & \mathcal{C}_{11_S} & \mathbf{0} & \mathbf{0} & \mathcal{C}_{13_S} \\ \mathbf{0} & \mathbf{0} & \mathcal{C}_{22_R} & \mathcal{C}_{22_K} & \mathcal{C}_{23_R} \\ \mathbf{0} & \mathbf{0} & \mathcal{C}_{22_K}^T & \mathcal{C}_{22_S} & \mathcal{C}_{23_S} \\ \mathcal{C}_{13_R}^T & \mathcal{C}_{13_S}^T & \mathcal{C}_{23_R}^T & \mathcal{C}_{23_S}^T & \mathcal{C}_{33} \end{bmatrix} + \dots \right.$$
$$\left. + \frac{1}{s} \begin{bmatrix} \Gamma_{11_R} & \Gamma_{11_K} & \mathbf{0} & \mathbf{0} & \Gamma_{13_R} \\ \Gamma_{11_K}^T & \Gamma_{11_S} & \mathbf{0} & \mathbf{0} & \Gamma_{13_S} \\ \mathbf{0} & \mathbf{0} & \Gamma_{22_R} & \Gamma_{22_K} & \Gamma_{23_R} \\ \mathbf{0} & \mathbf{0} & \Gamma_{22_K}^T & \Gamma_{22_S} & \Gamma_{23_S} \\ \Gamma_{13_R}^T & \Gamma_{13_S}^T & \Gamma_{23_R}^T & \Gamma_{23_S}^T & \Gamma_{33} \end{bmatrix} \right) \begin{bmatrix} \mathbf{x}_{1_R} \\ \mathbf{x}_{1_S} \\ \mathbf{x}_{2_R} \\ \mathbf{x}_{2_S} \\ \mathbf{x}_3 \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathcal{B}_1 \\ \mathbf{0} \\ \mathcal{B}_2 \\ \mathcal{B}_3 \end{bmatrix} \mathbf{u}(s), \quad (5.43)$$

where each subnet is further split into the blocks corresponding to internal nodes to be eliminated $\mathbf{x}_{i_R}$, and terminals to be preserved $\mathbf{x}_{i_S}$, $i = 1, 2$ and $\mathbf{x}_3$ are the separator nodes. The projection which, for each subnet $i = 1, 2$ zeroes out the $\mathcal{C}_{ii_K}$, $\mathcal{C}_{i3_R}$ connection blocks, and reduces the internal matrices $\mathcal{G}_{ii_R}$, $\mathcal{C}_{ii_R}$, Γ_{ii_R} together with the connection blocks $\mathcal{G}_{ii_K}$, Γ_{ii_K} is:

$$\mathcal{X}_Q = \begin{bmatrix} \mathcal{Q}_1 & \mathcal{W}_{11} & \mathbf{0} & \mathbf{0} & \mathcal{W}_{13} \\ \mathbf{0} & \mathbf{I}_{S_1} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathcal{Q}_2 & \mathcal{W}_{22} & \mathcal{W}_{23} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I}_{S_2} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I}_3 \end{bmatrix}, \quad \text{where} \quad \begin{aligned} \mathcal{W}_{11} &= -\mathcal{C}_{11_R}\mathcal{C}_{11_K} \\ \mathcal{W}_{13} &= -\mathcal{C}_{11_R}\mathcal{C}_{13_R} \\ \mathcal{W}_{22} &= -\mathcal{C}_{22_R}\mathcal{C}_{22_K} \\ \mathcal{W}_{13} &= -\mathcal{C}_{22_R}\mathcal{C}_{23_R} \end{aligned} \quad (5.44)$$

[2] For clarity, only the 2-way partitioning is shown: two subnets and one separator.

Table 5.1: *Diagram with the main derivations for reduction in the first- and second-order form, in the PACT framework*

	1st order	← link →	2nd order
Start	$\left(\left[\begin{array}{c\|cc} \mathcal{G}_S & \mathcal{G}_K^T & \mathcal{E}_S^T \\ \hline \mathcal{G}_K & \mathcal{G}_R & \mathcal{E}_R^T \\ \mathcal{E}_S & \mathcal{E}_R & \mathbf{0} \end{array}\right] + s\left[\begin{array}{c\|cc} \mathcal{C}_S & \mathcal{C}_K^T & \mathbf{0} \\ \hline \mathcal{C}_K & \mathcal{C}_R & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & -\mathcal{L} \end{array}\right]\right)\left[\begin{array}{c} \mathbf{v}_S \\ \hline \mathbf{v}_R \\ -\mathbf{i}_L \end{array}\right] = \left[\begin{array}{c} \mathcal{B}_S \\ \hline \mathbf{0} \\ \mathbf{0} \end{array}\right]\mathbf{u}$ $\Leftrightarrow \left(\left[\begin{array}{c\|c} \mathbf{G}_P & \mathbf{G}_C^T \\ \hline \mathbf{G}_C & \mathbf{G}_I \end{array}\right] + s\left[\begin{array}{c\|c} \mathbf{C}_P & \mathbf{C}_C^T \\ \hline \mathbf{C}_C & \mathbf{C}_I \end{array}\right]\right)\left[\begin{array}{c} \mathbf{x}_P \\ \hline \mathbf{X}_I \end{array}\right] = \left[\begin{array}{c} \mathbf{b}_P \\ \hline \mathbf{0} \end{array}\right]\mathbf{u}$ $\Downarrow$	$\Gamma_R = \mathcal{E}_R^T\mathcal{L}^{-1}\mathcal{E}_R,$ $\Gamma_S = \mathcal{E}_S^T\mathcal{L}^{-1}\mathcal{E}_S,$ $\Gamma_K = \mathcal{E}_S^T\mathcal{L}^{-1}\mathcal{E}_R$	$\left(\left[\begin{array}{cc} \mathcal{G}_R & \mathcal{G}_K \\ \mathcal{G}_K^T & \mathcal{G}_S \end{array}\right] + s\left[\begin{array}{cc} \mathcal{C}_R & \mathcal{C}_K \\ \mathcal{C}_K^T & \mathcal{C}_S \end{array}\right] + \frac{1}{s}\left[\begin{array}{cc} \Gamma_R & \Gamma_K \\ \Gamma_K^T & \Gamma_S \end{array}\right]\right)\left[\begin{array}{c} \mathbf{v}_R \\ \mathbf{v}_S \end{array}\right] = \left[\begin{array}{c} \mathbf{0} \\ \mathcal{B}_S \end{array}\right]\mathbf{u}$ $\Downarrow$
Transformation	$\mathbf{X} = \left[\begin{array}{cc} \mathbf{I} & \mathbf{0} \\ \mathbf{W} & \mathbf{I} \end{array}\right]$, where $\mathbf{W} = -\mathbf{C}_I^{-1}\mathbf{C}_C$ $\Downarrow$	$\mathbf{C}_I^{-1}\mathbf{C}_C = \left[\begin{array}{c} \mathcal{C}_R^{-1}\mathcal{C}_K \\ \hline \mathbf{0} \end{array}\right]$	$\mathcal{X} = \left[\begin{array}{c\|c} \mathbf{I} & \mathcal{W} \\ \hline \mathbf{0} & \mathbf{I} \end{array}\right]$, where $\mathcal{W} = -\mathcal{C}_R^{-1}\mathcal{C}_K$ $\Downarrow$
Transformed model	$\left(\left[\begin{array}{c\|c} \mathbf{G}_P' & \mathbf{G}_C'^T \\ \hline \mathbf{G}_C' & \mathbf{G}_I \end{array}\right] + s\left[\begin{array}{c\|c} \mathbf{C}_P' & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{C}_I \end{array}\right]\right)\left[\begin{array}{c} \mathbf{x}_P' \\ \hline \mathbf{x}_I \end{array}\right] = \left[\begin{array}{c} \mathbf{b}_P \\ \hline \mathbf{0} \end{array}\right]\mathbf{u}(s)$ $\Downarrow$		$\left(\left[\begin{array}{cc} \mathcal{G}_R & \mathcal{G}_K' \\ \mathcal{G}_K'^T & \mathcal{G}_S' \end{array}\right] + s\left[\begin{array}{cc} \mathcal{C}_R & \mathbf{0} \\ \mathbf{0}^T & \mathcal{C}_S' \end{array}\right] + \frac{1}{s}\left[\begin{array}{cc} \Gamma_R & \Gamma_K' \\ \Gamma_K'^T & \Gamma_S' \end{array}\right]\right)\left[\begin{array}{c} \mathbf{v}_R \\ \mathbf{v}_S' \end{array}\right] = \left[\begin{array}{c} \mathbf{0} \\ \mathcal{B}_S \end{array}\right]\mathbf{u}$ $\Downarrow$
Transfer function	$\mathbf{Y}'(s) = \underbrace{\left(\mathbf{G}_P' + s\mathbf{C}_P'\right)}_{\mathbf{Y}_P'(s)} - \underbrace{\mathbf{G}_C'^T\left(\mathbf{G}_I + s\mathbf{C}_I\right)^{-1}\mathbf{G}_C'}_{\mathbf{Y}_I'(s)}$	$\mathbf{Y}'(s) = \mathcal{Y}'(s)$	$\mathcal{Y}'(s) = \underbrace{\left(\mathcal{G}_S' + s\mathcal{C}_S' + \frac{1}{s}\Gamma_S'\right)}_{\mathcal{Y}_S'(s)} - \underbrace{\left(\mathcal{G}_K' + \frac{1}{s}\Gamma_K'\right)^T\left(\mathcal{G}_R + s\mathcal{C}_R + \frac{1}{s}\Gamma_R\right)^{-1}\left(\mathcal{G}_K' + \frac{1}{s}\Gamma_K'\right)}_{\mathcal{Y}_R'(s)}$
Matching moments	Rational Krylov on $\mathbf{Y}_I'(s)$ (for matching at $s_0 = 0$, $\mathbf{G}_I$ must be invertible, otherwise apply transformations proposed in [55, 57]) to isolate the singularities at DC.	$\mathbf{Y}_I'(s) = -\frac{1}{s}\Gamma_S' + \mathcal{Y}_R'(s)$	Rational Krylov on linearized version of $\mathcal{Y}_R'(s)$, i.e. SAPOR [66] (for matching at $s_0 = 0$, Γ_R must be invertible)
Reduced transfer function	$\widehat{\mathbf{Y}}(s) = \underbrace{\left(\mathbf{G}_P' + s\mathbf{C}_P'\right)}_{\mathbf{Y}_P'(s)} - \underbrace{\widehat{\mathbf{G}}_C^T(\widehat{\mathbf{G}}_I + s\widehat{\mathbf{C}}_I)^{-1}\widehat{\mathbf{G}}_C}_{\widehat{\mathbf{Y}}_I(s)}$ $\Downarrow$	$\widehat{\mathbf{Y}}(s) = \widehat{\mathcal{Y}}(s)$	$\widehat{\mathcal{Y}}(s) = \underbrace{\left(\mathcal{G}_S' + s\mathcal{C}_S' + \frac{1}{s}\Gamma_S'\right)}_{\mathcal{Y}'(s)} - \underbrace{\left(\widehat{\mathcal{G}}_K + \frac{1}{s}\widehat{\Gamma}_K\right)^T\left(\widehat{\mathcal{G}}_R + s\widehat{\mathcal{C}}_R + \frac{1}{s}\widehat{\Gamma}_R\right)^{-1}\left(\widehat{\mathcal{G}}_K + \frac{1}{s}\widehat{\Gamma}_K\right)}_{\widehat{\mathcal{Y}}_R(s)}$ $\Downarrow$
Reduced model	$\left(\left[\begin{array}{c\|c} \mathbf{G}_P' & \widehat{\mathbf{G}}_C^T \\ \hline \widehat{\mathbf{G}}_C & \widehat{\mathbf{G}}_I \end{array}\right] + s\left[\begin{array}{c\|c} \mathbf{C}_P' & \mathbf{0} \\ \hline \mathbf{0} & \widehat{\mathbf{C}}_I \end{array}\right]\right)\left[\begin{array}{c} \mathbf{x}_P' \\ \hline \widehat{\mathbf{X}}_I \end{array}\right] = \left[\begin{array}{c} \mathbf{b}_P \\ \hline \mathbf{0} \end{array}\right]\mathbf{u} \Leftrightarrow$ $\left(\left[\begin{array}{c\|cc} \mathcal{G}_S' & \widehat{\mathcal{G}}_K^T & \widehat{\mathcal{E}}_S^T \\ \hline \widehat{\mathcal{G}}_K & \widehat{\mathcal{G}}_R & \widehat{\mathcal{E}}_R^T \\ \widehat{\mathcal{E}}_S & \widehat{\mathcal{E}}_R & \mathbf{0} \end{array}\right] + s\left[\begin{array}{c\|cc} \mathcal{C}_S' & \mathbf{0} & \mathbf{0} \\ \hline \mathbf{0} & \widehat{\mathcal{C}}_R & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & -\widehat{\mathcal{L}} \end{array}\right]\right)\left[\begin{array}{c} \mathbf{v}_S \\ \hline \widehat{\mathbf{v}}_R \\ -\widehat{\mathbf{i}}_L \end{array}\right] = \left[\begin{array}{c} \mathcal{B}_S \\ \hline \mathbf{0} \\ \mathbf{0} \end{array}\right]\mathbf{u}$	$\widehat{\Gamma}_R = \widehat{\mathcal{E}}_R^T\widehat{\mathcal{L}}^{-1}\widehat{\mathcal{E}}_R,$ $\Gamma_S' = \widehat{\mathcal{E}}_S^T\widehat{\mathcal{L}}^{-1}\widehat{\mathcal{E}}_S,$ $\widehat{\Gamma}_K = \widehat{\mathcal{E}}_S^T\widehat{\mathcal{L}}^{-1}\widehat{\mathcal{E}}_R$	$\left(\left[\begin{array}{cc} \widehat{\mathcal{G}}_R & \widehat{\mathcal{G}}_K \\ \widehat{\mathcal{G}}_K^T & \mathcal{G}_S' \end{array}\right] + s\left[\begin{array}{cc} \widehat{\mathcal{C}}_R & \mathbf{0} \\ \mathbf{0}^T & \mathcal{C}_S' \end{array}\right] + \frac{1}{s}\left[\begin{array}{cc} \widehat{\Gamma}_R & \widehat{\Gamma}_K \\ \widehat{\Gamma}_K^T & \Gamma_S' \end{array}\right]\right)\left[\begin{array}{c} \widehat{\mathbf{v}}_R \\ \mathbf{v}_S' \end{array}\right] = \left[\begin{array}{c} \mathbf{0} \\ \mathcal{B}_S \end{array}\right]\mathbf{u}$ $\Leftrightarrow \left(\widehat{\mathcal{G}} + s\widehat{\mathcal{C}} + \frac{1}{s}\widehat{\Gamma}\right)\widehat{\mathbf{v}} = \widehat{\mathcal{B}}\mathbf{u}$ $\Downarrow$
Model for synthesis	$\left(\left[\begin{array}{cc} \widehat{\mathcal{G}} & \overline{U} \\ \overline{U}^T & \mathbf{0} \end{array}\right] + s\left[\begin{array}{cc} \widehat{\mathcal{C}} & \mathbf{0} \\ \mathbf{0} & -\overline{\mathcal{L}} \end{array}\right]\right)\left[\begin{array}{c} \widehat{\mathbf{v}} \\ -\widehat{\mathbf{i}}_{\overline{\mathcal{L}}} \end{array}\right] = \left[\begin{array}{c} \widehat{\mathcal{B}} \\ \mathbf{0} \end{array}\right]\mathbf{u}$	$\overline{\mathcal{L}} = \overline{\mathcal{D}}^{-1}$ $\Leftarrow$	Rank revealing transformation: $\widehat{\Gamma} = \left[\begin{array}{cc} \overline{U} & U_0 \end{array}\right]\left[\begin{array}{cc} \overline{\mathcal{D}} & \mathbf{0} \\ \mathbf{0} & \mathcal{D}_0 \end{array}\right]\left[\begin{array}{c} \overline{U}^T \\ U_0^T \end{array}\right] = \overline{U}\,\overline{\mathcal{D}}\,\overline{U}^T + \overline{U}_0\overline{\mathcal{D}}_0\overline{U}_0^T = \overline{\Gamma} + \Gamma_0$

and $\mathcal{Q}_1$, $\mathcal{Q}_2$ are the projections (for instance obtained via SAPOR) reducing the internal contributions $\mathcal{Y}_{i_R}^{'}(s)$ for each subnet. Projecting (5.43) with $\mathcal{X}_Q$ gives the reduced model in BBD form:

$$\left(\begin{bmatrix} \widehat{\mathcal{G}}_{11_R} & \widehat{\mathcal{G}}_{11_K} & \mathbf{0} & \mathbf{0} & \widehat{\mathcal{G}}_{13_R} \\ \widehat{\mathcal{G}}_{11_K}^T & \mathcal{G}_{11_S}^{'} & \mathbf{0} & \mathbf{0} & \mathcal{G}_{13_S}^{'} \\ \mathbf{0} & \mathbf{0} & \widehat{\mathcal{G}}_{22_R} & \widehat{\mathcal{G}}_{22_K} & \widehat{\mathcal{G}}_{23_R} \\ \mathbf{0} & \mathbf{0} & \widehat{\mathcal{G}}_{22_K}^T & \mathcal{G}_{22_S}^{'} & \mathcal{G}_{23_S}^{'} \\ \widehat{\mathcal{G}}_{13_R}^T & \mathcal{G}_{13_S}^{'T} & \widehat{\mathcal{G}}_{23_R}^T & \mathcal{G}_{23_S}^{'T} & \overline{\mathcal{G}}_{33} \end{bmatrix} + s \begin{bmatrix} \widehat{\mathcal{C}}_{11_R} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathcal{C}_{11_S}^{'} & \mathbf{0} & \mathbf{0} & \mathcal{C}_{13_S}^{'} \\ \mathbf{0} & \mathbf{0} & \widehat{\mathcal{C}}_{22_R} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathcal{C}_{22_S}^{'} & \mathcal{C}_{23_S}^{'} \\ \mathbf{0} & \mathcal{C}_{13_S}^{'T} & \mathbf{0} & \mathcal{C}_{23_S}^{'T} & \overline{\mathcal{C}}_{33} \end{bmatrix} + \dots \right.$$
$$\left. + \frac{1}{s} \begin{bmatrix} \widehat{\Gamma}_{11_R} & \widehat{\Gamma}_{11_K} & \mathbf{0} & \mathbf{0} & \widehat{\Gamma}_{13_R} \\ \widehat{\Gamma}_{11_K}^T & \Gamma_{11_S}^{'} & \mathbf{0} & \mathbf{0} & \Gamma_{13_S}^{'} \\ \mathbf{0} & \mathbf{0} & \widehat{\Gamma}_{22_R} & \widehat{\Gamma}_{22_K} & \widehat{\Gamma}_{23_R} \\ \mathbf{0} & \mathbf{0} & \widehat{\Gamma}_{22_K}^T & \Gamma_{22_S}^{'} & \Gamma_{23_S}^{'} \\ \widehat{\Gamma}_{13_R}^T & \Gamma_{13_S}^{'T} & \widehat{\Gamma}_{23_R}^T & \Gamma_{23_S}^{'T} & \overline{\Gamma}_{33} \end{bmatrix} \right) \begin{bmatrix} \mathbf{x}_{1_R} \\ \mathbf{x}_{1_S} \\ \mathbf{x}_{2_R} \\ \mathbf{x}_{2_S} \\ \mathbf{x}_3 \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathcal{B}_1 \\ \mathbf{0} \\ \mathcal{B}_2 \\ \mathcal{B}_3 \end{bmatrix} \mathbf{u}(s). \quad (5.45)$$

Finally, the rank-revealing decomposition of $\widehat{\Gamma}$ in (5.45) can be performed to cast the reduced model (5.45) in first-order form symmetric (5.42), which is then synthesized as an *RC* netlist and simulated.

5.3.1 Towards solving the DC moment matching problem

In circuit analysis, often the first type of simulation performed is a DC analysis, which computes a solution to the circuit equations when all capacitors act as open-circuits and all inductors as short circuits. It is thus desirable for the reduced circuit to have the same DC solution as the original circuit. In other words, the reduction should preserve at least one moment of the original transfer function at *DC*. For *RC* circuits, this is easily achieved (see Chapter 4), while for *RLC* additional numerical challenges arise when the underlying conductance matrix is singular. These are discussed next, as well as possibilities to overcome them. Recalling (1.13), for moment matching at DC (at $s = 0$) the system matrix $\mathbf{A}$ must be invertible (w.l.o.g. we refer in the next discussion to $\mathbf{A}$, rather than to $\mathcal{G}$, or $\mathbf{G}$). In typical Krylov-based reduction setups, if $\mathbf{A}$ is singular a non-zero expansion point $s_0 \neq 0$ is chosen and moments are matched around this point. In this manner however, the explicit matching at DC is not directly addressed. Here we are interested in computing a DC solution even when $\mathbf{A}$ is singular.

In the *RC* case, singularities in $\mathbf{A} = -\mathcal{G}$ occur only if there are nodes with no connections to resistors, or nodes which are isolated at DC from the port nodes (what [55, 57] call type-1 singularities). In Chapter 4, the reducing projection for *RC* circuits, whether in the partitioned or unpartitioned case, was able to easily match moments at DC, even when the relevant $\mathbf{A}$ matrix was singular. This was possible because the singularities could be easily isolated from the network (where easily means without additional fill-

creating transformations), so that there always remained an invertible sub-block of $\mathbf{A}$ (details are given in Sect. 4.2.2).

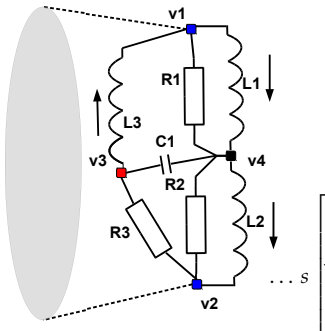
In the *RLC* case, type-1 singularities can be removed in a similar manner, however the so-called type-2 singularities [55, 57] pose additional challenges (we refer to singularities of $\mathbf{A}$, which represents the conductance matrix of an *RLC* circuit in the first-order form). Type-2 singularities can occur if the network: (a) has inductor loops or (b) has inductive paths between terminals. If a network is well-defined, such situations do not occur in practice. However, even when type-2 singularities are not present in the original, unpartitioned problem, inductive paths can appear between terminals and separator nodes, or between separator nodes, in the subnets resulting from partitioning. We discuss possibilities to resolve these.

One elegant solution to avoid type-2 singularities would be to ensure that partitioning does not pick as separator nodes those which would create inductive paths between the subnetwork's terminals (recall that the terminals of a subnet are a subset of the network terminals plus the separator nodes pertaining to the subnet). Another solution (assuming that the partitioning is done naively without accounting for the singularities) would be to isolate that part of a subnet which contains an inductive path between terminals and not reduce it (similar actions have been proposed in [69]). This alternative is justified as in practice such subnets do not appear often or are typically small. A third alternative was proposed in [55, 57] and involves a set of split transformations based on the range and nullspace of $\mathbf{A}$. These split the network into a part with no DC singularities, to be reduced, and one containing all DC singularities which is preserved. The transformations however introduce fill-in, which is undesirable (an example on how to achieve the same effect in a sparse manner is given in [55], however the feature is not documented in detail). Next, we investigate whether the second-order form provides additional alternatives.

Singularities at *DC*: first vs. second-order form

One of the motivations for performing reduction in the second vs. the first-order form was to avoid singularities which may be introduced in the $\mathbf{G}_I$ internal matrices per subnet. This can happen when nodes to which inductors are connected become separator nodes. An example is provided next, which illustrates how a singularity which appears in the first-order form, can be by-passed, under certain conditions, in the second-order form.

Fig. 5.1 shows an *RLC* subnet that may result from partitioning, where the blue nodes v_1, v_2 are separator nodes, connecting this subnet to the rest of the circuit, v_3 is an actual terminal (in red) and v_4 is an internal node. Hence, when considered separately from the rest of the network, nodes v_1, v_2 become terminals of the subnet, along with v_3. The governing MNA equations for this subnet, in the first-order form partitioned according to (5.3)-(5.4), are shown on the right of Fig. 5.1. Notice that the internal matrix $\mathbf{G}_I$ re-



$$(\mathbf{G}+s\mathbf{C})\mathbf{x}=\mathbf{B}\mathbf{u} \;\Leftrightarrow$$

$$\left(\left[\begin{array}{ccc|ccc} g_1 & 0 & 0 & -g_1 & 1 & 0 & -1 \\ 0 & g_2+g_3 & -g_3 & -g_2 & 0 & -1 & 0 \\ 0 & -g_3 & g_3 & 0 & 0 & 0 & 1 \\ \hline -g_1 & -g_2 & 0 & g_1+g_2 & -1 & 1 & 0 \\ 1 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 1 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 & 0 & 0 & 0 \end{array}\right] + \ldots \right.$$

$$\left. \ldots\, s \left[\begin{array}{ccc|cccc} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & c_1 & -c_1 & 0 & 0 & 0 \\ \hline 0 & 0 & -c_1 & c_1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -L_1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -L_2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -L_3 \end{array}\right] \right) \left[\begin{array}{c} v_1 \\ v_2 \\ v_3 \\ \hline v_4 \\ i_{L_1} \\ i_{L_2} \\ i_{L_3} \end{array}\right] = \left[\begin{array}{ccc} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{array}\right] \left[\begin{array}{c} i_1 \\ i_2 \\ i_3 \end{array}\right]$$

Figure 5.1: *Left: RLC sub-component of a larger circuit illustrating singularities at DC. Right: MNA circuit matrices partitioned according to (5.3)-(5.4) with a singular* $\mathbf{G}_I$ *block.*

sulting from the structure of Fig. 5.1 is singular: $\mathcal{E}_R$ has two linearly dependent columns and also a zero column. Matching moments at $s_0 = 0$ (DC) via the Krylov subspace (5.13) is not possible directly for this subnet, hence [57] propose further transformations to eliminate the singularities from $\mathbf{G}_I$, at the price of introducing more fill-in. It would be possible though to form a Krylov subspace (5.13) for an expansion point $s_0 \neq 0$.

When written in the second-order form (5.17), the governing MNA equations for this *RLC* subnet are:

$$\left(\left[\begin{array}{ccc|c} g_1 & 0 & 0 & -g_1 \\ 0 & g_2+g_3 & -g_3 & -g_2 \\ 0 & -g_3 & g_3 & 0 \\ \hline -g_1 & -g_2 & 0 & g_1+g_2 \end{array}\right] + \frac{1}{s}\left[\begin{array}{ccc|c} \gamma_1+\gamma_3 & 0 & -\gamma_3 & -\gamma_1 \\ 0 & \gamma_2 & 0 & -\gamma_2 \\ -\gamma_3 & 0 & \gamma_3 & 0 \\ \hline -\gamma_1 & -\gamma_2 & 0 & \gamma_1+\gamma_2 \end{array}\right]\right) \left[\begin{array}{c} v_1 \\ v_2 \\ v_3 \\ \hline v_4 \end{array}\right] = \left[\begin{array}{ccc} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{array}\right] \left[\begin{array}{c} i_1 \\ i_2 \\ i_3 \end{array}\right],$$

where the internal matrix blocks are $\mathcal{G}_R = g_1 + g_2$, $\Gamma_R = \gamma_1 + \gamma_2$. After the linearization of $\mathcal{Y}_R'(s)$ according to (5.38), it becomes possible to express and match the moment at DC for this subnet. This avoids the fill-introducing operations that would otherwise be necessary in the first-order form. In this example, matching at DC is possible in the second-order form because Γ_R is invertible (although in the first-order form $\mathbf{G}_I$ is otherwise singular); in practice, networks have been encountered where Γ_R is singular, in which case the DC moment matching problem in the second-order form still remains open.

An extended version of the previous subnet is presented in Fig. 5.2, which has the same three terminals $v_1, \ldots, v_3$ and five internal nodes $v_4, \ldots v_8$. Note that, as in the previous example, if the system is expressed in first-order form, the internal matrix $\mathbf{G}_I$ will

be singular. This circuit is reduced nonetheless in the second-order form with SAPOR based on matching one moment of $\mathcal{Y}_R^{'}(s)$ at $s = 0$. The comparison between the response of the original and the reduced circuit is shown in Fig. 5.2, where the accurate approximation for the low frequencies is visible.

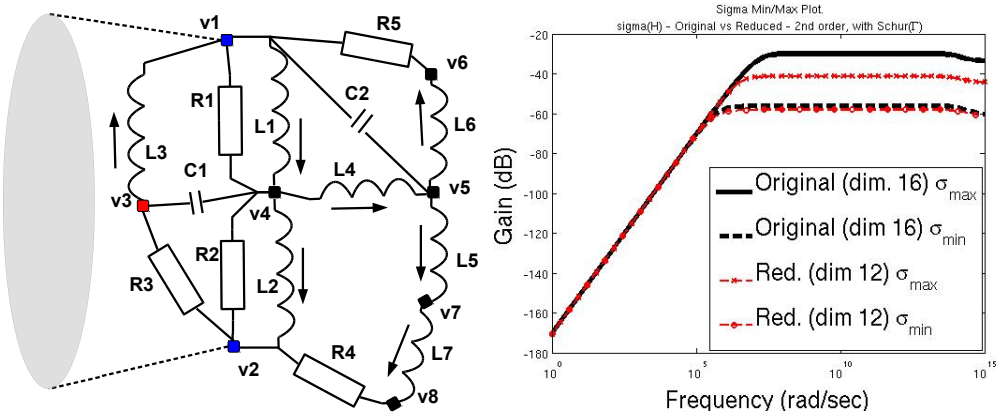


Figure 5.2: *Left: RLC subnet reduced with moment matching at DC, in the second-order form. Right: the curves for the maximum and minimum singular values of the frequency response, for the original (black) and reduced (red) circuits respectively. The reduced circuit matches the behavior of the original well for low frequencies, expected due to moment matching at DC.*

5.3.2 Identifying fill-in

For simplicity, in this discussion we denote by $\star$ an arbitrary block from the reduced matrices in (5.45). As for multi-terminal *RC* reduction, the fill generated in the $\star_S^{'}$ and $\overline{\star}_{33}$ blocks is minimized through partitioning and the preservation of separator nodes. For *RC* circuits, reduction by matching the two moments at DC usually guarantees sufficient accuracy, these being entirely captured by the $\star_S^{'}$ blocks (see Chapter 4). In contrast, in the *RLC* reduction scenario contributions from the reduced internal blocks are present in the form of $\widehat{\star}_R$ and $\widehat{\star}_K$, as seen from (5.45). So, while the $\star_S^{'}$ and $\overline{\star}_{33}$ terms are sparse, the $\widehat{\star}_R$ and $\widehat{\star}_K$ blocks are dense. The density of $\widehat{\star}_R$ and $\widehat{\star}_K$ is due to two factors: (a) the transformation which zeros-out the $\mathcal{C}_{ii_K}, \mathcal{C}_{i3_R}$ connection blocks from the original model (5.43), and (b) the formation of the Krylov subspaces reducing $\mathcal{Y}_{i_R}^{'}(s)$ per subnet, which results in dense projections $\mathcal{Q}_i$.

It is nevertheless possible to improve sparsity by simultaneously diagonalizing the $\widehat{\mathcal{G}}_{ii_R}$, $\widehat{\mathcal{C}}_{ii_R}$ blocks via an eigenvalue decomposition [these are symmetric, positive semi-definite matrices hence the pencil $(\widehat{\mathcal{G}}_{ii_R}, \widehat{\mathcal{C}}_{ii_R})$ has real eigenvalues]. After this operation, the

reduced $\widehat{\mathcal{C}}$ matrix becomes very sparse as seen from its structure in (5.45). The remaining dense block appears as $\overline{\mathbf{U}}$ when (5.45) is cast in the first-order form (5.42). $\overline{\mathbf{U}}$ could be obtained in a sparse manner for instance via a sparse LDL^T decomposition of $\widehat{\Gamma}$, provided that $\widehat{\Gamma}$ is sparse. In most examples encountered however, the resulting $\widehat{\Gamma}$ of (5.45) has dense $\widehat{\star}_R$ and $\widehat{\star}_K$ blocks as previously explained.

Hence, the final reduced system (5.42) when unstamped as an *RC* circuit will typically have very few capacitors and many resistors, sometimes more resistors than in the original circuit. A much sparser representation could be obtained if (a), the transform to zero-out the $\mathcal{C}_{ii_K}$, $\mathcal{C}_{i3_R}$ blocks would be by-passed, (b) a sparse representation for the Krylov subspaces $\mathcal{Q}_i$ would be found and (c) $\widehat{\Gamma}$ could be factorized as into an LDL^T in a sparse manner. Point (a) could be avoided if for instance a splitting of nodes into $\mathbf{x}_{i_R}$ and $\mathbf{x}_{i_S}$ per subnet would be found, as to ensure directly that $\mathcal{C}_{ii_K} = \mathbf{0}$, $\mathcal{C}_{i3_R} = \mathbf{0}$. Point (b) especially raises a new question in itself: the formation of sparse Krylov subspaces from sparse matrices. Point (c) would be possible as long as $\widehat{\Gamma}$ is sparse, which in turn would be satisfied with the help of (a) and (b). Having identified the causes of fill-in and the potential directions to reduce it, implementing these aspects remains for further research. For another partition-based approach which relies on macro-model realization rather than on unstamping, and thus controls sparsity in a different manner, we refer to PartMOR [70].

5.3.3 Moment matching and the dimensions of the reduced blocks

The internal contributions $\mathcal{Y}'_{i_R}(s)$ to be reduced per subnet have the form (5.24). For instance, during the reduction of subnet i ($i = 1, 2$), the internal blocks corresponding to (5.24) are as follows:

$$\mathcal{G}_R := \mathcal{G}_{ii_R}, \;\; \mathcal{C}_R := \mathcal{C}_{ii_R}, \;\; \Gamma_R := \Gamma_{ii_R} \tag{5.46}$$

$$\mathcal{G}'_K := \begin{bmatrix} \mathcal{G}'_{ii_K} & \mathcal{G}'_{i3_R} \end{bmatrix}, \;\; \Gamma'_K := \begin{bmatrix} \Gamma'_{ii_K} & \Gamma'_{i3_R} \end{bmatrix}. \tag{5.47}$$

As for any Krylov-based reduction, the number of columns of $\mathcal{G}'_K + \frac{1}{s}\Gamma'_K$ dictates the dimension M_i for one moment of $\mathcal{Y}'_{i_R}(s)$. This is equal to the number of terminals of subnet i plus the number of separator nodes through which subnet i communicates with the rest of the circuit, i.e.: $M_i = |\mathbf{x}_{i_S}| + |\mathbf{x}_{33}|$. Thus, if m moments of $\mathcal{Y}'_{i_R}(s)$ are to be matched per subnet, the dimension of the reduced internal matrices $\widehat{\star}_{ii_R}$ will be $m \times M_i$ (this will be also the number of rows of $\widehat{\star}_{ii_K}$ and $\widehat{\star}_{i3_R}$). Hence, as the accuracy of the reduced model increases with the number of moments matched, it is desirable to match as many moments as possible, however without introducing much fill-in. This can be controlled by ensuring that the partitioning spreads terminals per partition sufficiently, and by generating as few separator nodes as possible. In this manner, M_i, the dimension of a block moment per subnet, can be kept small. In addition, a careful selection of

expansion points at which the moments of $\mathcal{Y}_{i_R}^{'}(s)$ are generated could help to achieve a suitable trade-off between small dimensionality and accuracy. For instance, with knowledge from [78], the optimal shifts could be chosen close to the dominant eigenvalues of a suitable representation of $\mathcal{Y}_{i_R}^{'}(s)$ [for instance via a linearized version of $\mathcal{Y}_{i_R}^{'}(s)$]. Finding an appropriate linearization or even more, an analogy between the eigenmodes of the $\mathcal{Y}_{i_R}^{'}(s)$ term and those of the start-point system (5.43) are new research questions and remain for further investigation.

5.3.4 When partitioning is advantageous

For very large circuits with node numbers exceeding tens of thousands, partitioning becomes advantageous irrespective of how many terminals the circuit may have. Most importantly, it provides the computational benefit of performing reduction per subnet, when for the unpartitioned problem it is too costly or even unfeasible to form a Krylov projection as with PRIMA [71], SPRIM [27] SAPOR [66], or [8]. These advantages could be brought to full potential, provided that the sparsity considerations identified above are resolved.

The partition-based derivations of this section however do not directly apply to *RLCk* circuits, that is circuits with mutual inductances. First, for *RLCk* circuits the inductance matrix $\mathcal{L}$ is typically dense, due to the presence of mutuals. Hence Γ to start with is dense, and a reasonable partition of the graph $G := nzp(\mathcal{G} + \mathcal{C} + \Gamma)$ cannot be found, due to the fact that there are no "good" separator nodes (the graph G is fully connected). A good partition *could* be found by ignoring the mutuals, however the true Γ (with mutuals re-included after the appropriate permutations) is no longer in BBD-form, which was otherwise a necessary assumption for the derivations above. Hence, a more efficient way to partition *RLCk* networks is needed.

5.4 Numerical results

An RLC transmission line was reduced according to the partition-based approach in second-order form, as described in Sect. 5.3. After partitioning, two moments at $s_1 = 10^4$ and one moment at $s_2 = 10^{12}$ are matched per subnet. Table 5.2 collects the reduction statistics, where a partitioned-based and an unpartitioned reduction are attempted. The dimension n_i denotes the number of internal variables[3] of the original model (5.1) compared to the reduced model (5.42). The partition-based reduction was faster to perform than the unpartitioned version, and also resulted in a more accurate model. This is also reflected in the Matlab bode plots from Fig. 5.3, and the AC Spectre simulation comparisons from Fig. 5.4. As expected from the analysis of Sect. 5.3.2, there are more resis-

[3] Internal variables are all those except the voltages at the terminal nodes.

Table 5.2: *RLC reduction statistics: partitioned vs. unpartitioned*

Net	Type	n_i	#R	#C	#L	Sim. time (s)	Red. time (s)
1. **Tline** $p = 22$	Original	3629	6020	6142	135	1	-
	Red: partitioned ($N = 4$ subnets)	583	19922	821	0	0.9	11.8
	Red: unpartitioned	141	5105	351	0	0.14	26.5

tors in the reduced than in the original circuit; this is also reflected in the re-simulation time, which is almost the same as for the original circuit. Hence further investigation is required into improving the sparsity of the netlist, especially in the number of resistors. This example clearly reveals the importance of re-using the reduced models in simulation. Only when this final step is carried out, is one able to truly asses both the quality and the efficiency gains obtained from model reduction.

5.5 Concluding remarks

The reduction of multi-terminal *RLC* circuits is addressed in this chapter. In particular, the PACT approach [57] is analyzed especially in the context of partitioning. Based on the second-order formulation, a hierarchical procedure to reduce *RLC* networks is derived. Through the border-block-diagonal hierarchy, the blocks which contain fill-in are identified, based on which directions to further improve sparsity are provided. The problem of preserving moments at *DC* was also analyzed and suggestions to overcome it were provided. An example from circuit simulation was reduced in this framework, through which the computational advantages of partitioning are revealed. The results motivate a more thorough investigation into achieving a better trade-off between small dimensionality, sparsity and accuracy. The first recommendation in this respect is to derive a similar partition-based framework using SPRIM [27] for the first-order form or SAPOR [66] for the second-order form to reduce each subnet, rather than the PACT approach [57] analyzed here (the basics of such an approach are included in Appendix 5.6.1). For future work, the extension to *RLCk* circuits is also relevant.

As a summarizing comparison between reduction in the first and second order form, Table 5.3 collects the advantages, limitations and future research questions for each scenario, both for the unpartitioned, as well as the partitioned framework.

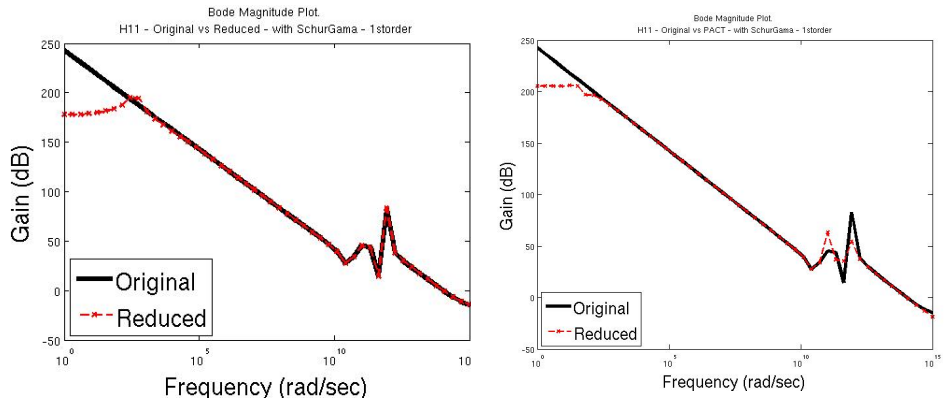


Figure 5.3: ***RLC tline****. Transfer function for the original model compared to two reduced models: partition-based (left) and unpartitioned (right). The partition based model captures the oscillations better.*

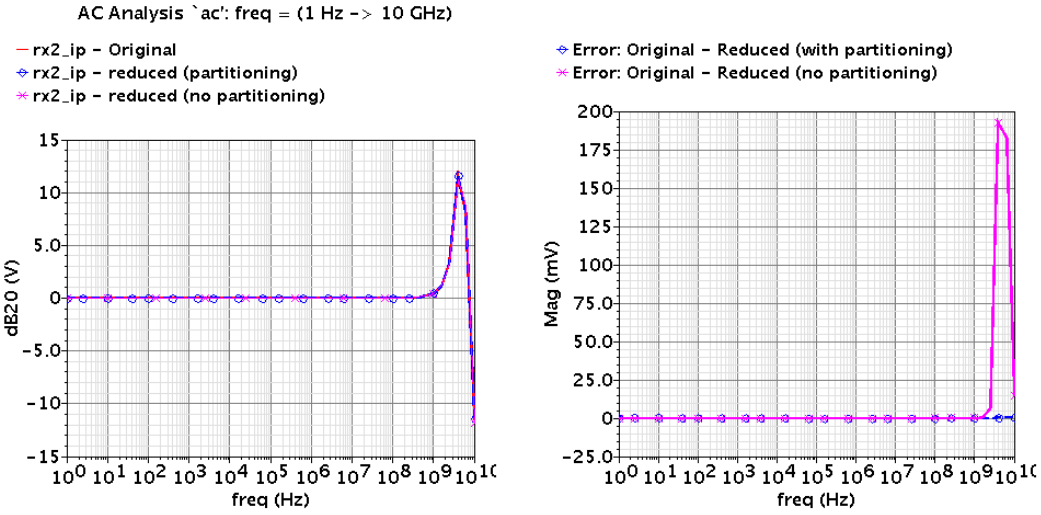


Figure 5.4: ***RLC tline****: AC analysis of the synthesized circuits. Original (red) and Reduced with partitioning (blue) match very well, while the reduced model obtained without partitioning (magenta) deviates slightly. The error plots on the right also reflect that the partitioning-based model is more accurate.*

5.6 Appendix

5.6.1 An SPRIM partition-based reduction approach

The structure preserving method SPRIM [27] or the block version BSMOR [95] have recently become popular for reducing *RLC* circuits. The main benefit of such approaches is that, from a subspace $\mathbf{V}$ which spans a moment-matching Krylov subspace, a block

version can be constructed which matches more moments (proportional to the number of blocks [95]). As a consequence, any block structure in the original circuit matrices will also be preserved in the reduced matrices. This concept has been further exploited in the RLCSYN paper [93], as to also preserve the structure of the input/output incidence matrices; this ensures that reduced models can be unstamped into *RLC* netlists without controlled sources (the RLCSYN [93] authors denote this special input-output structure preserving projection as SPRIM/IOPOR).

Here, an SPRIM/IOPOR reduced model is constructed based on the border- block- diagonal (BBD) matrix structure resulting from partitioning the *RLC* netlist. The reduced model (a) preserves the input/output structure of the original system, and with that, the terminal connectivity and (b) preserves the BBD structure of the original partitioned model. The partition-based SPRIM/IOPOR procedure proposed here shows strong potential for reducing very large netlists with many terminals. Furthermore, it exploits the terminal grounding and recovery action proposed in Chapter 6, and demonstrates its functionality on industrial *RLC* examples.

The main concept of the partition-based SPRIM/IOPOR reduction and synthesis procedure is outlined next, using a partitioning into two subnets and one separator (finer partitions are treated similarly):

1. Start with the multi-terminal *RLC* system in first order form (5.1)
2. Partition the *RLC* network using for instance nested dissection [e.g. based on the second order form (5.16) the partitioning induces the BBD reordered system (5.43)]
3. Rearrange the partitioned *RLC* system in first order form[4]:

$$\left(\left[\begin{array}{ccccc|c} \mathcal{G}_{11_R} & \mathcal{G}_{11_K} & \mathbf{0} & \mathbf{0} & \mathcal{G}_{13_R} & \\ \mathcal{G}_{11_K}^T & \mathcal{G}_{11_S} & \mathbf{0} & \mathbf{0} & \mathcal{G}_{13_S} & \\ \mathbf{0} & \mathbf{0} & \mathcal{G}_{22_R} & \mathcal{G}_{22_K} & \mathcal{G}_{23_R} & \widetilde{\mathcal{E}} \\ \mathbf{0} & \mathbf{0} & \mathcal{G}_{22_K}^T & \mathcal{G}_{22_S} & \mathcal{G}_{23_S} & \\ \mathcal{G}_{13_R}^T & \mathcal{G}_{13_S}^T & \mathcal{G}_{23_R}^T & \mathcal{G}_{23_S}^T & \mathcal{G}_{33} & \\ \hline & & -\widetilde{\mathcal{E}}^T & & & \mathbf{0} \end{array}\right] + s \left[\begin{array}{ccccc|c} \mathcal{C}_{11_R} & \mathcal{C}_{11_K} & \mathbf{0} & \mathbf{0} & \mathcal{C}_{13_R} & \\ \mathcal{C}_{11_K}^T & \mathcal{C}_{11_S} & \mathbf{0} & \mathbf{0} & \mathcal{C}_{13_S} & \\ \mathbf{0} & \mathbf{0} & \mathcal{C}_{22_R} & \mathcal{C}_{22_K} & \mathcal{C}_{23_R} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathcal{C}_{22_K}^T & \mathcal{C}_{22_S} & \mathcal{C}_{23_S} & \\ \mathcal{C}_{13_R}^T & \mathcal{C}_{13_S}^T & \mathcal{C}_{23_R}^T & \mathcal{C}_{23_S}^T & \mathcal{C}_{33} & \\ \hline & & \mathbf{0} & & & \mathcal{L} \end{array}\right]\right) \left[\begin{array}{c} \mathbf{x}_{1_R} \\ \mathbf{x}_{1_S} \\ \mathbf{x}_{2_R} \\ \mathbf{x}_{2_S} \\ \mathbf{x}_3 \\ \hline \mathbf{i}_L \end{array}\right] = \left[\begin{array}{c} \mathbf{0} \\ \mathcal{B}_1 \\ \mathbf{0} \\ \mathcal{B}_2 \\ \mathcal{B}_3 \\ \hline \mathbf{0} \end{array}\right] \mathbf{u}(s), \tag{5.48}$$

where $\widetilde{\mathcal{E}}$ is the reordered incidence matrix of inductor connections (the partition-

[4]The procedure could continue directly in the second order form, using for instance the second order approach SAPOR/IOPOR [66, 93], however this in turn would require a linearization to a first order form.

ing induces a special structure in $\widetilde{\mathcal{E}}$ as well, but is ignored here for simplicity). In (5.48), the nodes of each non-separator subnet $i = 1, 2$ are split into internal nodes $\mathbf{x}_{i_R}$ and terminals $\mathbf{x}_{i_S}$.

4. System (5.48) is usually ungrounded, hence the underlying matrix pencil is singular. To make the system numerically consistent, ground the network by removing one terminal (together with its corresponding row and column) from (5.48). To avoid notation overloading, we simply denote (5.48) as being already grounded, and denote the underlying system matrices as $\mathbf{G}_g$, $\mathbf{C}_g$, $\mathbf{B}_g$.
5. Pick a set of expansion points s_k, and construct a full-rank matrix $\mathbf{V}$ which spans the Krylov subspace associated with the desired number of moments for each expansion point.
6. Split $\mathbf{V}$ according to the block structure of (5.48):

$$\mathbf{V}^T = \begin{bmatrix} \mathbf{V}_{1_R}^T & \mathbf{V}_{1_S}^T & \mathbf{V}_{2_R}^T & \mathbf{V}_{2_S}^T & \mathbf{V}_3^T & \mathbf{V}_L^T \end{bmatrix}, \tag{5.49}$$

compress the blocks corresponding to internal nodes and inductor currents to full column rank: $\widetilde{\mathbf{V}}_{1_R}$, $\widetilde{\mathbf{V}}_{2_R}$, $\widetilde{\mathbf{V}}_L$, and replace the blocks corresponding to terminals and separator nodes with the appropriately sized identity matrices $\mathbf{I}_{1_S}$, $\mathbf{I}_{2_S}$, $\mathbf{I}_3$.
7. Construct the block-diagonal projection:

$$\widetilde{\mathbf{V}} = \begin{bmatrix} \widetilde{\mathbf{V}}_{1_R} & & & & & \\ & \mathbf{I}_{1_S} & & & & \\ & & \widetilde{\mathbf{V}}_{2_R} & & & \\ & & & \mathbf{I}_{2_S} & & \\ & & & & \mathbf{I}_3 & \\ & & & & & \widetilde{\mathbf{V}}_L \end{bmatrix}, \tag{5.50}$$

8. Construct the reduced model $\widehat{\mathbf{G}}_g = \widetilde{\mathbf{V}}^T \mathbf{G}_g \widetilde{\mathbf{V}}$, $\widehat{\mathbf{C}}_g = \widetilde{\mathbf{V}}^T \mathbf{C}_g \widetilde{\mathbf{V}}$, $\widehat{\mathbf{B}}_g = \widetilde{\mathbf{V}}^T \mathbf{B}$, which will reveal the following structure:

$$\left(\left[\begin{array}{ccccc|c} \widehat{\mathcal{G}}_{11_R} & \widehat{\mathcal{G}}_{11_K} & \mathbf{0} & \mathbf{0} & \widehat{\mathcal{G}}_{13_R} & \\ \widehat{\mathcal{G}}_{11_K}^T & \mathcal{G}_{11_S} & \mathbf{0} & \mathbf{0} & \mathcal{G}_{13_S} & \\ \mathbf{0} & \mathbf{0} & \widehat{\mathcal{G}}_{22_R} & \widehat{\mathcal{G}}_{22_K} & \widehat{\mathcal{G}}_{23_R} & \widehat{\mathcal{E}} \\ \mathbf{0} & \mathbf{0} & \widehat{\mathcal{G}}_{22_K}^T & \mathcal{G}_{22_S} & \mathcal{G}_{23_S} & \\ \widehat{\mathcal{G}}_{13_R}^T & \mathcal{G}_{13_S}^T & \widehat{\mathcal{G}}_{23_R}^T & \mathcal{G}_{23_S}^T & \mathcal{G}_{33} & \\ \hline & & & -\widehat{\mathcal{E}}^T & & \mathbf{0} \end{array}\right] + s \left[\begin{array}{ccccc|c} \widehat{\mathcal{C}}_{11_R} & \widehat{\mathcal{C}}_{11_K} & \mathbf{0} & \mathbf{0} & \widehat{\mathcal{C}}_{13_R} & \\ \widehat{\mathcal{C}}_{11_K}^T & \mathcal{C}_{11_S} & \mathbf{0} & \mathbf{0} & \mathcal{C}_{13_S} & \\ \mathbf{0} & \mathbf{0} & \widehat{\mathcal{C}}_{22_R} & \widehat{\mathcal{C}}_{22_K} & \widehat{\mathcal{C}}_{23_R} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \widehat{\mathcal{C}}_{22_K}^T & \mathcal{C}_{22_S} & \mathcal{C}_{23_S} & \\ \widehat{\mathcal{C}}_{13_R}^T & \mathcal{C}_{13_S}^T & \widehat{\mathcal{C}}_{23_R}^T & \mathcal{C}_{23_S}^T & \mathcal{C}_{33} & \\ \hline & & \mathbf{0} & & & \widehat{\mathcal{L}} \end{array}\right] \right) \left[\begin{array}{c} \widehat{\mathbf{x}}_{1_R} \\ \mathbf{x}_{1_S} \\ \widehat{\mathbf{x}}_{2_R} \\ \mathbf{x}_{2_S} \\ \mathbf{x}_3 \\ \hline \widehat{\mathbf{i}}_L \end{array}\right] = \left[\begin{array}{c} \mathbf{0} \\ \mathcal{B}_1 \\ \mathbf{0} \\ \mathcal{B}_2 \\ \mathcal{B}_3 \\ \hline \mathbf{0} \end{array}\right] \mathbf{u}(s), \tag{5.51}$$

9. Remove possible dependencies in $\widehat{\mathcal{E}}$ using the Schur decomposition of $\widehat{\Gamma} = \widehat{\mathcal{E}}\widehat{\mathcal{L}}^{-1}\widehat{\mathcal{E}}^T$ [see equation (5.41)], and cast the reduced system in the symmetric first order form (5.42).
10. Synthesize the reduced model (5.42) via RLCSYN [93] unstamping into the equivalent *RC* netlist. During the unstamping process, all *R*, *C* elements which would usually be connected to ground, are instead connected to the terminal which was removed by grounding in step 4 (see Chapter 6 for the theoretical interpretation of the terminal grounding and recovery step). Because the projection (5.50) preserves input/output structure, the resulting netlist contains no controlled sources.

Notice that in the reduced system (5.51), the matrix blocks corresponding to terminals and separator nodes are unchanged, hence remain sparse. The reduced blocks corresponding to internal nodes, $\mathcal{G}_{ii_R}$, $\mathcal{C}_{ii_R}$, $i = 1, 2$ are usually dense, however they can be easily simultaneously diagonalized using the eigendecomposition of the matrix pencils $(\mathcal{G}_{ii_R}, \mathcal{C}_{ii_R})$, or tridiagonalized according to [85]. Hence, ultimately the only dense parts will be $\widehat{\mathcal{G}}_{ii_K}$, $\widehat{\mathcal{G}}_{i3_R}$, $\widehat{\mathcal{C}}_{ii_K}$, $\widehat{\mathcal{C}}_{i3_R}$, and $\widehat{\mathcal{E}}$ [or $\overline{\mathcal{U}}$ of (5.42) after the rank revealing decomposition of $\widehat{\Gamma}$]. Future research could address means to directly construct a sparse basis for the internal projection blocks $\widetilde{\mathbf{V}}_{i_R}$, $\widetilde{\mathbf{V}}_L$, and for $\overline{\mathcal{U}}$, by exploiting for instance Householder transformations on these matrices.

Rather than constructing the projection $\mathbf{V}$ as in (5.49) from the whole matrices, and then split it into (5.50), an alternative would be to construct directly small SPRIM/IOPOR projections per subnet. This would require more careful book-keeping, as the structure of $\widetilde{\mathcal{E}}$ induced by the partitioning would have to be additionally accounted for. A consequence of exploiting structure in $\widetilde{\mathcal{E}}$ is that the number of columns of $\widehat{\mathcal{E}}$ in (5.51) would be larger, and therefore the reduced model will have more current unknowns $\widehat{\mathbf{i}}_L$. This could further improve the quality of the reduced model. Computational advantages may also be gained as smaller Krylov subspaces would be computed for each subnet. Implementing the more advanced version remains for future work.

The partition-based SPRIM reduction was applied on the *RLC* circuits in the table below: two *RLC* transmission line models with 22 terminals (**Tline1** and **Tline2**), and one *RLC* inductor model with 3 terminals (**Inductor**). The reduction statistics, as well as the number of partitions N used and the number of moments matched are listed in the table. The best reduction rates, both in internal nodes and circuit elements, was achieved for the **Inductor** model. For the other two circuits, further sparsification methods may be appropriate as above described, as to also drive down the number of circuit elements and improve the resimulation time.

The Spectre [16] simulation of the **Inductor** circuit in Fig. 5.5 shows perfect match between the response of the original and the reduced circuits. The same holds for the simulation of **Tline2** in Figure 5.6.

Table with RLC reduction results for the partition-based SPRIM approach

Net	Type	n_i	#R	#C	#L	Sim. time (s)	N # parts	Moments matched
1. **Tline1** $p = 22$	Original	3629	6020	6142	135	1	9	one at 10^7 and 10^{11}
	partSPRIM	262	12995	2949	0	0.9		
2. **Inductor** $p = 3$	Original	19557	31468	20375	129	11.2	15	two at 10^5, 10^7, 10^{11} and 10^{13}
	partSPRIM	227	4876	599	0	0.4		
3. **Tline2** $p = 22$	Original	11659	6382	7300	3988	6.9	23	one at 10^7 and 10^{11}
	partSPRIM	646	36386	8024	0	5.5		

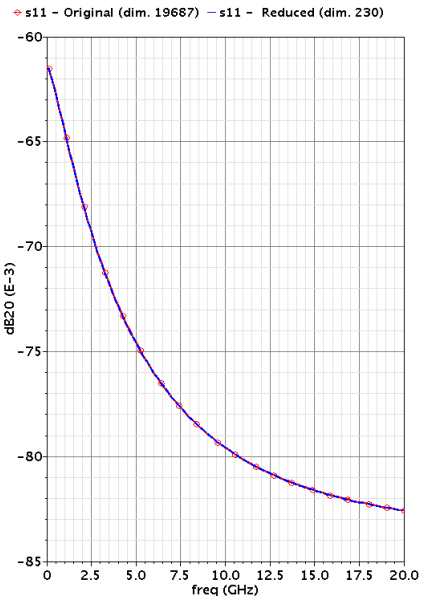


Figure 5.5: ***Inductor****: S-parameter of the original (red) and reduced (blue) circuit overlap.*

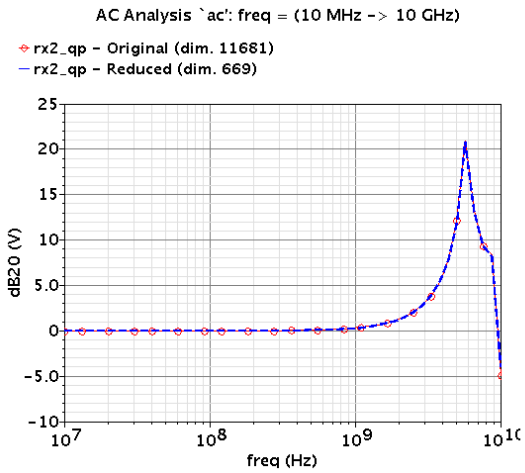


Figure 5.6: ***Tline2****. AC simulations of the original (red) and reduced (blue) circuits.*

Table 5.3: *Comparison between first and second-order form RLC reduction*

		First order	**Second order**: new
No partition	Advantages	• developed in [57] • passivity preserved by *split* congruence transforms • moments matched at s_i after $\mathbf{C}_C$ is zeroed, based on a structure preserving (split) Krylov projection of $\mathbf{Y}_I^{'}(s)$ in (5.10) • possibilities to match at DC (when $\mathbf{G}_I$ is singular, based on split transformations involving the null and spanning subspaces of $\mathbf{G}_I$) • applies to $RLCk$	• analogous with first-order (Lemma 5.2.1) • passivity preserved by congruence transforms (no splitting) • moments matched after $\mathcal{C}_K$ is zeroed-out, via a SAPOR reduction of $\mathcal{Y}_R^{'}(s)$ in (5.24) • rank deficiencies of the reduced model are eliminated via a decomposition of $\widehat{\Gamma} \Rightarrow$ successful re-simulations • applies to $RLCk$
	Limitations	• transformation to zero-out $\mathbf{C}_C$ introduces fill-in • when $\mathbf{G}_I$ is singular, matching moments at DC requires further network transformations • reduced $\widehat{\mathbf{G}}$ may have rank deficiencies which prejudice the re-simulation • expensive/unfeasible for circuits with many nodes • dense reduced matrices for many-terminal systems	• transformation to zero-out $\mathbf{C}_K$ introduces fill-in • the dimension for the linearized system of $\mathcal{Y}_R^{'}(s)$ from (5.24) is twice that of $\mathcal{G}_R$ • expensive/unfeasible for circuits with many nodes • dense reduced matrices for many-terminal systems
	Future work	• moment matching without zeroing out $\mathbf{C}_C$ • moment matching at DC avoiding the dense transformations from [55,57] to eliminate singularities at DC	• moment matching without zeroing out $\mathcal{C}_K$ • develop theory for matching moments at DC
Partition based: new	Advantages	• an SPRIM partition based alternative (see Appendix 5.6.1)	• natural form for partitioning $\Rightarrow \mathcal{G}, \mathcal{C}, \Gamma$ in BBD structure • simple implementation via congruence transforms operating on the BBD hierarchy, no need for splitting • efficient for systems with many nodes • especially for many-terminal systems, sparsity improved in the $\mathcal{G}_S^{'}, \mathcal{C}_S^{'}, \Gamma_S^{'}$ compared to unpartitioned case
	Limitations (aside from unpartitioned case)	• $\mathbf{G}_I$ matrices per subnet become singular if there are inductors connected to the separator nodes • fill-in still present from internal reduction per subnet	• moments at DC not well defined in general (possible when Γ_R is invertible) • fill-in still present from internal reduction per subnet, and the final rank revealing decomposition of $\widehat{\Gamma}$ • does not apply to $RLCk$
	Future work (aside from unpartitioned case)	• finding partitions which do not promote inductively connected nodes as separators • extension to $RLCk$ • apply SPRIM [27] directly per subnet rather than [57]	• finding sparse LDL^T decompositions of $\widehat{\Gamma}$ • finding suitable partitions for $RLCk$ • apply SAPOR [66] directly per subnet

Chapter 6

Using general reduced order models in simulations

A unifying framework is presented for the reduction of multi-terminal systems, which enables the use of reduced models in simulations irrespective of the reduction method chosen. In particular it is shown how, even when the reduction does not preserve the structure of the input/output incidence matrix, the reduced model can still be synthesized, using the unstamping approach, without controlled sources. The framework also allows ungrounded systems to be reduced via the numerically sound grounded representation, with the guarantee that an ungrounded reduced netlist is finally recovered. This is then re-inserted via all terminal connections in a simulation flow.

6.1 Introduction

To speed up the simulation of RLC electrical circuits such as interconnect models or parasitics networks, various model order reduction (MOR) methods [4] can nowadays be employed. Along with the reduction, a synthesis step is often necessary to convert the reduced model back into an electrical network. This must then be inserted in the desired simulation environment in place of the original circuit. A simple synthesis approach which reads the reduced MNA matrix entries into a circuit topology is the unstamping method [93], which is also used in this chapter.

Two major constraints[1] are known to limit the applicability of traditional reduction

[1] A third possible limitation of unstamping is that it does not guarantee positive circuit elements, which may be unacceptable for some circuit simulators. This problem lies outside the scope of this chapter; all simulations herein are performed with Spectre [16], which accepts negative elements.

methods and of the unstamping synthesis approach to *multi-terminal systems (e.g. circuits with many input/output nodes)*: (a) the matrix pencils underlying multi-terminal systems are often singular and (b) the underlying reducing projection may destroy the structure of the input/output matrices and with that, the physical interpretation of terminal nodes. An undesirable consequence of (b) is that controlled sources would be introduced during synthesis to model the connectivity of the reduced circuit at the terminal nodes.

The multi-terminal reduction methods presented in Chapters 4, 3, 5 automatically bypass both of these limitations due to the special way in which the reducing projection is formed. More specifically, the underlying projections preserve the structure of the input/output incidence matrices, allowing for synthesis without controlled sources. This holds both for systems which are grounded or ungrounded. Another class of methods which are still limited by (a) but overcome (b) are input-output structure preserving methods such as SPRIM/IOPOR from [93]. Many other more general MOR methods such as the established PRIMA [71] or the Loewner method [60] share none of these properties. Hence, although such methods may be qualitatively or computationally efficient, their reuse in practice (for instance inside a simulation setup) is cumbersome or at least inelegant.

This chapter brings a new contribution by showing how, despite the known limitations, even more general reduction methods are able to handle multi-terminals systems. The reduction-synthesis framework proposed here eliminates the pencil singularity using a simple pre-processing of the original circuit, and recovers the connectivity at all terminal nodes using a post-processing of the reduced model. With the proposed procedure, the re-use of reduced models obtained from various reduction techniques becomes straightforward. It is shown for instance how reduced-order macromodels obtained with the Loewner interpolation-based method [60] can be easily re-inserted in a circuit simulation flow, with all terminal connections preserved. The proposed framework enables measurements to be taken for systems which have an original ungrounded representation (hence an underlying singular pencil). This is especially important when reducing multi-terminal circuits which do not have a direct connection to ground (such a circuit could be for instance a sub-net of a larger netlist, with the ground node belonging to another part of this larger netlist).

6.1.1 Problem definition

Consider the modified nodal analysis (MNA) description of an RC[2] circuit:

$$\begin{cases} (\mathbf{G}+s\mathbf{C})\mathbf{v}(s) &= \mathbf{B}\mathbf{u}(s) \\ \mathbf{y}(s) &= \mathbf{B}^T\mathbf{v} \end{cases}, \tag{6.1}$$

[2]For simplicity, the derivations are based on RC reduction, but are generalized immediately to the RLC case especially when the system is expressed in the second order, e.g., as in [93].

where MNA matrices $\mathbf{G}$, $\mathbf{C}$ are symmetric, non-negative definite, corresponding to the stamps of resistor and capacitor values respectively. $\mathbf{x} \in \mathbb{R}^{n+p}$ denote the node voltages, measured at the *p terminals* and *n internal nodes*. $\mathbf{u} \in \mathbb{R}^p$ are the currents injected into the terminals. Assuming that the first p nodes are the terminals, the input incidence matrix has the special structure:

$$\mathbf{B} \;=\; \left[\begin{array}{c} \mathbf{I}_p \\ \mathbf{0} \end{array}\right] \in \mathbb{R}^{(n+p)\times p}, \tag{6.2}$$

where $\mathbf{I}_p \in \mathbb{R}^{p\times p}$ is the identity matrix of size p. The outputs are the voltage drops at the terminal nodes: $\mathbf{y}(s) = \mathbf{B}^T\mathbf{v} = \mathbf{v}_p$. In model reduction, an appropriate $\mathbf{V} \in \mathbb{R}^{(n+p)\times(k+p)}, k \geq 0$ is sought, such that the system matrices and unknowns are reduced to:

$$\widehat{\mathbf{G}} \;=\; \mathbf{V}^T\mathbf{G}\mathbf{V},\;\; \widehat{\mathbf{C}} = \mathbf{V}^T\mathbf{C}\mathbf{V} \;\; \in \mathbb{R}^{(k+p)\times(k+p)} \tag{6.3}$$

$$\widehat{\mathbf{B}} \;=\; \mathbf{V}^T\mathbf{B} \in \mathbb{R}^{(k+p)\times p},\;\; \widehat{\mathbf{x}} = \mathbf{V}^T\mathbf{x} \in \mathbb{R}^{k+p} \tag{6.4}$$

and satisfy: $(\widehat{\mathbf{G}} + s\widehat{\mathbf{C}})\widehat{\mathbf{x}}(s) = \widehat{\mathbf{B}}\mathbf{u}(s)$.

It is emphasized that the representation (6.1) corresponds to an *ungrounded* circuit. *Ungrounded* means that a reference node in (6.1) has not yet been chosen, this being fixed only in the simulation phase. From a mathematical viewpoint, we do not yet know which terminal will be set to ground in simulation - this is decided by the designer. Consequently, the matrix pair $(\mathbf{G}, \mathbf{C})$ underlying the ungrounded system (6.1) is singular. Furthermore, the projection $\mathbf{V}$ resulting from many model reduction methods such as PRIMA [71], or the Loewner method [60] does not retain the original structure of current injections (6.2), so the reduced $\widehat{\mathbf{B}}$ from (6.4) is dense. Hence, when a SPICE-equivalent circuit is derived from the reduced model using an unstamping method (e.g. RLCSYN [93]), controlled sources would be have to be used to unstamp the dense $\widehat{\mathbf{B}}$ [36,73]. Two problems emerge from this setup, which are resolved in this chapter:

(a) How can one use general projection based methods [which assume a regular ($\mathbf{G}$, $\mathbf{C}$) pencil] to reduce ungrounded circuits in such a way that all terminals re-appear in the reduced model?
(b) Assuming that such a reduction is possible, how can one unstamp a general reduced order model (6.3)-(6.4) without controlled sources?

Towards answering (a), the notion of a *generalized frequency response* or *generalized transfer function* is introduced. This expresses the input/output response of an ungrounded multi-port circuit in terms of the transfer function of the grounded circuit. In this manner, it becomes possible to reduce a grounded circuit while ensuring that the response of the ungrounded circuit at all terminals is recovered. The solution to (b) is given by an equivalence transformation which converts the reduced $\widehat{\mathbf{B}}$ into $\mathbf{T}^T\widehat{\mathbf{B}} = \left[\begin{array}{c} \mathbf{I} \\ \mathbf{0} \end{array}\right]$.

6.2 Ungrounded vs. grounded systems

In preparation for solving the above problems, the structure of matrices underlying ungrounded vs. grounded circuits is derived, together with the relationship between the solutions of the corresponding linear systems. Consider the simple resistor circuit example in Fig. 6.1: on the left, the ungrounded circuit is shown, while the grounded version is on the right. From Fig. 6.1, consider the ungrounded version on the left.

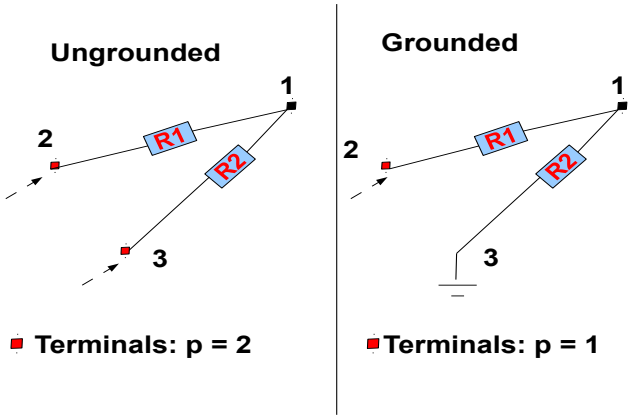


Figure 6.1: *Simple resistive circuit with two terminals. The left version is ungrounded, with currents flowing into nodes* 2 *and* 3 *(terminals). The right circuit is the grounded version, where node* 3 *is set to ground.*

Nodes 2 and 3 are terminals through which input currents i_2, i_3 are injected. The outputs are the voltages v_2, v_3 measured at the terminals. Node 1 is internal. From Kirchhoff's Current Law (KCL) and the branch constitutive equations the following MNA system results:

$$\underbrace{\begin{bmatrix} g_1+g_2 & -g_1 & -g_2 \\ -g_1 & g_1 & 0 \\ -g_2 & 0 & g_2 \end{bmatrix}}_{\mathbf{G}} \underbrace{\begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix}}_{\mathbf{v}} = \underbrace{\begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}}_{\mathbf{B}} \underbrace{\begin{bmatrix} i_2 \\ i_3 \end{bmatrix}}_{\mathbf{u}} \tag{6.5}$$

$$\mathbf{y} = \mathbf{B}^T\mathbf{v} \tag{6.6}$$

Note that $\mathbf{G}$ is singular, which is always the case for ungrounded circuit matrices. It is easy to see that the vector of all 1's $\in$ Null($\mathbf{G}$) [each row (and column) in $\mathbf{G}$ sum to 0]. Thus the solution to (6.5) cannot be determined by inverting $\mathbf{G}$. Denoting $\mathbb{1} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$,

the solution $\mathbf{v}$ can however be expressed as:

$$\mathbf{v} = \alpha \mathbb{1} + \mathbf{v}_0,$$

where $\mathbf{v}_0$ is the particular solution to (6.5) and α is a free parameter. Let us find $\mathbf{v}_0$ and the interpretation of α.

Expressing v_1 from the first row of (6.5) and replacing it in the second and third rows gives:

$$v_1 = \frac{g_1}{g_1+g_2}v_2 + \frac{g_2}{g_1+g_2}v_3 \tag{6.7}$$

$$\left(g_1 - \frac{g_1^2}{g_1+g_2}\right)v_2 - \frac{g_1 g_2}{g_1+g_2}v_3 = i_2 \tag{6.8}$$

$$-\frac{g_1 g_2}{g_1+g_2}v_2 + \left(g_2 - \frac{g_2^2}{g_1+g_2}\right)v_3 = i_3 \tag{6.9}$$

Note that $g_1 - \frac{g_1^2}{g_1+g_2} = g_2 - \frac{g_2^2}{g_1+g_2} = \frac{g_1 g_2}{g_1+g_2} := A$. Therefore the last two equations read:

$$\begin{aligned} Av_2 - Av_3 &= i_2 & (6.10)\\ -Av_2 + Av_3 &= i_3 \Rightarrow i_2 + i_3 = 0. & (6.11) \end{aligned}$$

As expected, one of the equations from (6.5) is redundant, and the unknown voltages are dependent. Let $v_3 = \alpha$. From (6.10) one obtains: $v_2 = \alpha + A^{-1}i_2 = \alpha + (\frac{1}{g_1} + \frac{1}{g_2})i_2$. Replacing v_2 and v_3 in the first row of (6.5), one obtains $v_1 = \alpha + \frac{1}{g_2}i_2$. Finally the general solution to the ungrounded system (6.5) is:

$$\mathbf{v} = \begin{bmatrix} \alpha + \frac{1}{g_2}i_2 \\ \alpha + (\frac{1}{g_1} + \frac{1}{g_2})i_2 \\ \alpha \end{bmatrix} = \alpha \underbrace{\begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}}_{\mathbb{1}} + \underbrace{\begin{bmatrix} \frac{1}{g_2}i_2 \\ (\frac{1}{g_1} + \frac{1}{g_2})i_2 \\ 0 \end{bmatrix}}_{\mathbf{v}_0}. \tag{6.12}$$

Next, it is shown that $\mathbf{v}_0$ is nothing but the solution of the grounded representation, where terminal node 3 is chosen as the reference node (i.e. when $\alpha = v_3 = 0$). Consider now the circuit version on the right of Fig. 6.1: the current injection into terminal 3 is removed and node 3 is grounded ($v_3 = 0$). The input is the current i_2 flowing into terminal 2, and the output is measured as the voltage of this node, namely v_2. The

corresponding MNA equations are:

$$\underbrace{\begin{bmatrix} g_1+g_2 & -g_1 \\ -g_1 & g_1 \end{bmatrix}}_{\mathbf{G}_g}\underbrace{\begin{bmatrix} v_1 \\ v_2 \end{bmatrix}}_{\mathbf{v}_g} = \underbrace{\begin{bmatrix} 0 \\ 1 \end{bmatrix}}_{\mathbf{B}_g}\underbrace{\begin{bmatrix} i_2 \end{bmatrix}}_{\mathbf{u}_g} \tag{6.13}$$

$$\mathbf{y}_g = \mathbf{B}_g^T\mathbf{v}_g \tag{6.14}$$

$\mathbf{G}_g$ is now invertible. The solution to (6.13) is completely determined, namely:

$$\mathbf{v}_g = \mathbf{G}_g^{-1}\mathbf{B}_g\mathbf{u}_g = \begin{bmatrix} \frac{1}{g_2}i_2 \\ (\frac{1}{g_1}+\frac{1}{g_2})i_2 \end{bmatrix} \tag{6.15}$$

Finally, note that the general solution (6.12) of the ungrounded system is expressed in terms of the solution (6.15) of the grounded system as follows:

$$\mathbf{v} = \alpha\mathbb{1} + \begin{bmatrix} \mathbf{v}_g \\ 0 \end{bmatrix}, \quad \alpha = v_3, \tag{6.16}$$

where $\alpha = v_3$ is taken as the reference voltage.

6.3 Generalized frequency response/transfer function

With the result (6.16) at hand, we proceed towards expressing the input/output transfer function of an ungrounded system in terms of that of the grounded system. From (6.14), the output of the grounded system is:

$$\mathbf{y}_g = \mathbf{B}_g^T\mathbf{v}_g = \underbrace{\mathbf{B}_g^T\mathbf{G}_g^{-1}\mathbf{B}_g}_{:=\mathbf{H}_g}\mathbf{u}_g, \tag{6.17}$$

from which the "transfer function" of the grounded circuit is identified as $\mathbf{H}_g$.

Next we form the *generalized transfer function* $\mathbf{H}$ for the ungrounded system (6.5) in terms of $\mathbf{H}_g$. Note that $\mathbf{B}$ is expressed in terms of $\mathbf{B}_g$ as follows: $\mathbf{B} = \begin{bmatrix} \mathbf{B}_g & \mathbb{0}_c \\ \mathbb{0}_r & 1 \end{bmatrix} \in \mathbb{R}^{(n+p)\times p}$, where $\mathbf{B}_g \in \mathbb{R}^{(n+p-1)\times(p-1)}$, and $\mathbb{0}_c$ and $\mathbb{0}_r$ are the row and column vector respectively of all 0s, corresponding to the dimensions of $\mathbf{B}_g$. For an ungrounded system of dimension $n+p$, we have $\mathbb{1} \in \mathbb{R}^{n+p}$, and for the grounded system $\mathbb{1}_g \in \mathbb{R}^{n+p-1}$. Recall also that $\mathbf{B} = \begin{bmatrix} \mathbf{I}_p \\ \mathbf{0} \end{bmatrix}$, so that $\mathbf{B}^T\mathbb{1} = \mathbb{1}_p \in \mathbb{R}^p$ is the vector of all 1's of length p. Similarly, $\mathbf{B}_g^T\mathbb{1}_g = \mathbb{1}_{p-1} \in \mathbb{R}^{p-1}$. Using (6.16), the general output for the ungrounded system (6.5)

is expressed as:

$$\begin{aligned}\mathbf{y}{=}\mathbf{B}^T\mathbf{v} &= \alpha\mathbf{B}^T\mathbb{1}+\mathbf{B}^T\mathbf{v}_0{=}\alpha\begin{bmatrix}\mathbf{B}_g^T & \mathbb{0}_r^T\\ \mathbb{0}_c^T & 1\end{bmatrix}\begin{bmatrix}\mathbb{1}_g\\ 1\end{bmatrix}+\begin{bmatrix}\mathbf{B}_g^T & \mathbb{0}_r^T\\ \mathbb{0}_c^T & 1\end{bmatrix}\begin{bmatrix}\mathbf{v}_g\\ 0\end{bmatrix}=\\ &= \alpha\begin{bmatrix}\mathbf{B}_g^T\mathbb{1}_g\\ 1\end{bmatrix}+\begin{bmatrix}\mathbf{B}_g^T\mathbf{v}_g\\ 0\end{bmatrix}{=}\alpha\begin{bmatrix}\mathbb{1}_{p-1}\\ 1\end{bmatrix}+\begin{bmatrix}\mathbf{H}_g\mathbf{u}_g\\ 0\end{bmatrix}=\begin{bmatrix}\alpha\mathbb{1}_{p-1}+\mathbf{H}_g\mathbf{u}_g\\ \alpha\end{bmatrix}\\ &= \underbrace{\begin{bmatrix}\mathbf{H}_g & \mathbb{1}_{p-1}\\ \mathbb{0} & 1\end{bmatrix}}_{\mathbf{H}}\underbrace{\begin{bmatrix}\mathbf{u}_g\\ \alpha\end{bmatrix}}_{\mathbf{u}_\alpha}.\end{aligned} \tag{6.18}$$

From (6.18) the generalized transfer function $\mathbf{H}$ of an ungrounded system is identified. It is expressed in terms of $\mathbf{H}_g$, the transfer function of the grounded circuit. The generalized input $\mathbf{u}_\alpha$ is the input $\mathbf{u}_g$ of the grounded circuit, and the free parameter α, the voltage of the reference node [e.g., the node v_3 from (6.5) which was set to ground in (6.13)].

The result (6.18) also holds for expressing the generalized transfer function $\mathbf{H}(s)\in\mathbb{C}^{p\times p}$ of an ungrounded multi-terminal *RC* circuit in terms of a grounded version, where $\mathbf{H}_g(s)=\mathbf{B}_g^T(\mathbf{G}_g+s\mathbf{C}_g)^{-1}\mathbf{B}_g\in\mathbb{C}^{(p-1)\times(p-1)}$. It is emphasized that for the ungrounded system, expressing "$\mathbf{H}(s)=\mathbf{B}^T(\mathbf{G}+s\mathbf{C})^{-1}\mathbf{B}$" directly is not allowed since the pencil $(\mathbf{G},\mathbf{C})$ is singular.

6.3.1 Recovering the reference node from the grounded system

From the generalized frequency response (6.18), system matrices associated with the ungrounded representation can be recovered from the grounded representation as follows:

$$\mathbf{H}(s){=}\begin{bmatrix}\mathbf{B}_g^T & \mathbb{1}_{p-1}\\ \mathbb{0}_c^T & 1\end{bmatrix}\left(\begin{bmatrix}\mathbf{G}_g & \mathbb{0}_c\\ \mathbb{0}_r & 1\end{bmatrix}{+}s\begin{bmatrix}\mathbf{C}_g & \mathbb{0}_c\\ \mathbb{0}_r & 0\end{bmatrix}\right)^{-1}\begin{bmatrix}\mathbf{B}_g & \mathbb{0}_c\\ \mathbb{0}_r & 1\end{bmatrix} \tag{6.19}$$

With system matrices so redefined, we are interested in rewriting the equations so that the node which was set to ground is re-introduced. Let v_p be the voltage drop at this node. The dynamical equations associated with (6.19) now read:

$$\begin{aligned}&\left(\begin{bmatrix}\mathbf{G}_g & \mathbb{0}_c\\ \mathbb{0}_r & 1\end{bmatrix}{+}s\begin{bmatrix}\mathbf{C}_g & \mathbb{0}_c\\ \mathbb{0}_r & 0\end{bmatrix}\right)\begin{bmatrix}\mathbf{v}_g\\ v_p\end{bmatrix} = \begin{bmatrix}\mathbf{B}_g & \mathbb{0}_c\\ \mathbb{0}_r & 1\end{bmatrix}\begin{bmatrix}\mathbf{u}_g\\ \alpha\end{bmatrix}\Leftrightarrow\\ &\Leftrightarrow\begin{cases}(\mathbf{G}_g+s\mathbf{C}_g)\mathbf{v}_g &= \mathbf{B}_g\mathbf{u}_g\\ v_p &= \alpha\end{cases}\end{aligned} \tag{6.20}$$

Let the voltage drops of the grounded net be split into voltages measured at internal and terminal nodes respectively: $\mathbf{v}_g = \begin{bmatrix} \mathbf{v}_{g_i} \\ \mathbf{v}_{g_t} \end{bmatrix}$. Note that if the ungrounded net has p terminals, there are $\mathbf{v}_{g_t} \in \mathbb{R}^{p-1}$ terminals in the grounded net. It follows that $\mathbf{B}_g^T \mathbf{v}_g = \mathbf{v}_{g_t}$. The output equation of (6.19) then reads:

$$\mathbf{y} = \begin{bmatrix} \mathbf{B}_g^T & \mathbb{1}_{p-1} \\ \mathbb{0}_r & 1 \end{bmatrix} \begin{bmatrix} \mathbf{v}_g \\ v_p \end{bmatrix} = \begin{bmatrix} \mathbf{B}_g^T \mathbf{v}_g + v_p \mathbb{1}_{p-1} \\ v_p \end{bmatrix} = \begin{bmatrix} \mathbf{v}_{g_t} + \alpha \mathbb{1}_{p-1} \\ \alpha \end{bmatrix}. \tag{6.21}$$

In other words, the outputs of the ungrounded net are the voltage drops at the $p-1$ terminals of the grounded net, measured as $\mathbf{v}_{g_t} \in \mathbb{R}^{p-1}$ plus the value of the reference voltage $v_p = \alpha$. The last output is the p'th terminal, the voltage $v_p = \alpha$ itself of the node which was removed from the network by grounding. This allows one to recover the ungrounded netlist by unstamping the grounded net specified by $\mathbf{G}_g$, $\mathbf{C}_g$, $\mathbf{B}_g$ as usually. However the branches that would usually be connected to a so called "ground" are now connected to the reference terminal node v_p. In this manner, the recovered netlist contains all terminals and can be reconnected directly to other circuit blocks and simulated.

6.3.2 Implications

Several implications emerge from the above derivations. These extend the applicability of state-of-the-art reduction methods to systems with dependencies such as ungrounded multi-terminal circuits.

1. The parameter α is independent of the frequency s, and represents the voltage of the chosen reference node.

2. After a reference node is chosen from one of the terminals, the grounded circuit (now with $p-1$ terminals) can be reduced with any reduction method which assume a regular pencil [what before we called $(\mathbf{G}_g, \mathbf{C}_g)$]. For instance, constructing reduced models from measurements of the original frequency response via the Loewner method [60], or PRIMA [71] becomes possible for ungrounded multi-terminal circuits. Although measurements of $\mathbf{H}(s)$ cannot be taken explicitly [due to the fact the the inverse $(\mathbf{G} + s_i\mathbf{C})$ does not exist as the pencil (**G**,**C**) is singular], they can be taken implicitly by measuring $\mathbf{H}_g(s)$ for different frequencies s_i [the pencil ($\mathbf{G}_g$,$\mathbf{C}_g$) is regular].

3. Connectivity via *all* terminals is recovered after reduction: the chosen reference node is simply re-inserted in the physical netlist as the p'th terminal (with the appropriate connections to the other circuit nodes) during the synthesis of the reduced model.

In the following section, the reduction and terminal recovery (without controlled sources) are described.

6.4 Model reduction

Let as before v_p be the terminal set to ground, and the grounded system be reduced via a projection $\mathbf{V} \in \mathbb{R}^{(n+p)\times(k+p)}$, $k+p$ being the dimension of the reduced model: $\widehat{\mathbf{G}}_g = \mathbf{V}^T\mathbf{G}_g\mathbf{V}$, $\widehat{\mathbf{C}}_g = \mathbf{V}^T\mathbf{C}_g\mathbf{V}$, $\widehat{\mathbf{B}}_g = \mathbf{V}^T\mathbf{B}_g$. The transfer function of the grounded reduced model is $\widehat{\mathbf{H}}_g(s) = \widehat{\mathbf{B}}_g^T(\widehat{\mathbf{G}}_g + s\widehat{\mathbf{C}}_g)^{-1}\widehat{\mathbf{B}}_g \in \mathbb{C}^{(p-1)\times(p-1)}$. Then the generalized transfer function of the *ungrounded reduced* model follows similarly to (6.18):

$$\widehat{\mathbf{H}}(s)=\begin{bmatrix} \widehat{\mathbf{H}}_g(s) & \mathbb{1}_{p-1} \\ \mathbb{0}_r & 1 \end{bmatrix}=\begin{bmatrix} \widehat{\mathbf{B}}_g^T & \mathbb{1}_{p-1} \\ \mathbb{0}_r & 1 \end{bmatrix}\left(\begin{bmatrix} \widehat{\mathbf{G}}_g & \mathbb{0}_c \\ \mathbb{0}_r & 1 \end{bmatrix}+s\begin{bmatrix} \widehat{\mathbf{C}}_g & \mathbb{0}_c \\ \mathbb{0}_r & 0 \end{bmatrix}\right)^{-1}\begin{bmatrix} \widehat{\mathbf{B}}_g & \mathbb{0}_c \\ \mathbb{0}_r & 1 \end{bmatrix}. \quad (6.22)$$

Let the unknowns in the reduced, ungrounded system be split as follows: $\widehat{\mathbf{v}} = \begin{bmatrix} \widehat{\mathbf{v}}_g^T \\ v_p \end{bmatrix} \in \mathbb{R}^{k+p}$, where $\widehat{\mathbf{v}}_g^T \in \mathbb{R}^{k+p-1}$ are the voltage drops at the nodes of the grounded circuit. Similarly to the original network, $\widehat{\mathbf{B}}_g^T\widehat{\mathbf{v}}_g = \widehat{\mathbf{v}}_{g_t}$. The dynamical system describing the reduced, ungrounded network associated with (6.22) reads:

$$\left\{\begin{array}{rcl} (\widehat{\mathbf{G}}_g + s\widehat{\mathbf{C}}_g)\widehat{\mathbf{v}}_g &=& \widehat{\mathbf{B}}_g\mathbf{u}_g \\ v_p &=& \alpha \\ \mathbf{y} &=& \begin{bmatrix} \widehat{\mathbf{v}}_{g_t} + \alpha\mathbb{1} \\ \alpha \end{bmatrix}. \end{array}\right. \quad (6.23)$$

The reduced netlist containing the $p-1$ terminals is unstamped from the reduced grounded net specified by $\widehat{\mathbf{G}}_g$, $\widehat{\mathbf{C}}_g$, $\widehat{\mathbf{B}}_g$. The output equation from (6.23) introduces the grounded terminal back in the physical netlist as a node with voltage $v_p = \alpha$. In other words the "ground" node is no longer fixed to voltage $v_p = \alpha = 0$ but is kept free as the p'th terminal, as in the original ungrounded model.

As a final step, the reduced input/output matrix $\widehat{\mathbf{B}}_g$ (which is usually dense) must be transformed to a form which enables straightforward synthesis of terminal re-connectivity, without introducing controlled sources. This is shown next.

6.4.1 Transforming a reduced model into synthesis-ready form

Towards transforming the reduced grounded model $\widehat{\mathbf{G}}_g$, $\widehat{\mathbf{C}}_g$, $\widehat{\mathbf{B}}_g$ into a form appropriate for unstamping without controlled sources, the terminal locations have to be retrieved

from $\widehat{\mathbf{B}}_g$. This is achieved as follows.

Assume that the dimension of the reduced model is K, where $K \geq p-1$, as is usually the case for projection based MOR methods. Thus $\widehat{\mathbf{B}}_g \in \mathbb{R}^{K\times(p-1)}$ is tall and thin (having linearly independent columns). A transformation $\mathbf{T} \in \mathbb{R}^{K\times K}$ is sought so that $\mathbf{T}^T\widehat{\mathbf{B}} = \begin{bmatrix} \mathbf{I}_{p-1} \\ \mathbf{0} \end{bmatrix} \in \mathbb{R}^{K\times(p-1)}$. This allows one to inject currents into the $p-1$ terminals of the reduced network just as in the original network. The transformation is based on the singular value decomposition (SVD) of $\widehat{\mathbf{B}}_g$: $\widehat{\mathbf{B}}_g = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$, where $\mathbf{U} \in \mathbb{R}^{K\times K}$, $\mathbf{\Sigma} \in \mathbb{R}^{K\times(p-1)}$, $\mathbf{V} \in \mathbb{R}^{(p-1)\times(p-1)}$. More precisely, $\mathbf{T}$ is obtained from:

$$\widehat{\mathbf{B}}_g = \mathbf{U}\begin{bmatrix} \mathbf{\Sigma}_{p-1} \\ \mathbf{0} \end{bmatrix}\mathbf{V}^T \;\Rightarrow\; \mathbf{T} = \mathbf{U}\begin{bmatrix} \mathbf{\Sigma}_{p-1}^{-1}\mathbf{V}^T & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_{K-p+1} \end{bmatrix}, \tag{6.24}$$

which transforms the reduced system into:

$$\Rightarrow \begin{cases} \widehat{\mathbf{B}}_g &= \mathbf{T}^T\widehat{\mathbf{B}}_g = \begin{bmatrix} \mathbf{I}_{p-1} \\ \mathbf{0} \end{bmatrix} \in \mathbb{R}^{K\times(p-1)} \\ \widehat{\mathbf{G}}_g &= \mathbf{T}^T\widehat{\mathbf{G}}_g\mathbf{T},\;\; \widehat{\mathbf{C}}_g = \mathbf{T}^T\widehat{\mathbf{C}}_g\mathbf{T} \end{cases} \tag{6.25}$$

In (6.25), the new incidence matrix $\widehat{\mathbf{B}}_g$ has the required form for direct synthesis via unstamping. The identity block $\mathbf{I}_{p-1} \in \mathbb{R}^{(p-1)\times(p-1)}$ represents the incidence of current injections into the $p-1$ terminals. The p'th terminal is re-inserted instead of the ground node as previously described [see equations (6.23) and comment thereafter]. Sect. 6.5.1 illustrates the procedure via a small example.

6.5 Numerical results and simulations

To demonstrate the functionality of the proposed multi-terminal reduction with terminal recovery, two examples are provided. The first is a simple *RC* network which clearly shows how the grounded terminal is re-introduced after reduction and the input/output transformation. The second is a larger industrial example of an *RC* transmission line with 22 terminals.

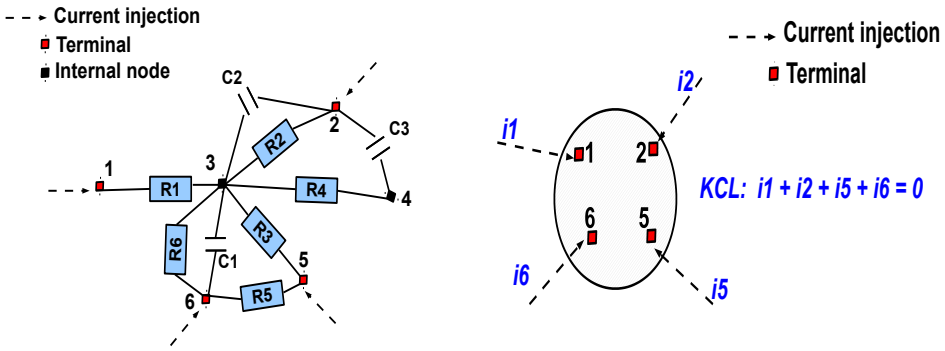


Figure 6.2: *Left: RC ungrounded circuit, where nodes* $1,2,5,6$ *are terminals, nodes* $3,4$ *are internal. Currents are injected into terminals. Right: KCL for the ungrounded network, where all nodes are viewed as one (the circle above). The sum of all currents entering this node is zero.*

6.5.1 Simple *RC* circuit

For the ungrounded simple *RC* network in Fig. 6.2, consider the MNA equations:

$$\underbrace{\left(\left[\begin{array}{ccccc|c} g_1{+}g_2{+}g_3{+}g_4{+}g_6 & -g_4 & -g_1 & -g_2 & -g_3 & -g_6 \\ -g_4 & g_4 & 0 & 0 & 0 & 0 \\ -g_1 & 0 & g_1 & 0 & 0 & 0 \\ -g_2 & 0 & 0 & g_2 & 0 & 0 \\ -g_3 & 0 & 0 & 0 & g_3{+}g_5 & -g_5 \\ \hline -g_6 & 0 & 0 & 0 & -g_5 & g_5{+}g_6 \end{array}\right] + \cdots\right)}_{\mathbf{G}} \tag{6.26}$$

$$s\underbrace{\left[\begin{array}{ccccc|c} c_1{+}c_2 & 0 & 0 & -c_2 & 0 & -c_1 \\ 0 & c_3 & 0 & -c_3 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ -c_2 & -c_3 & 0 & c_3{+}c_2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ \hline -c_1 & 0 & 0 & 0 & 0 & c_1 \end{array}\right]}_{\mathbf{C}}\underbrace{\left[\begin{array}{c} V_3(s) \\ V_4(s) \\ V_1(s) \\ V_2(s) \\ V_5(s) \\ \hline V_6(s) \end{array}\right]}_{\mathbf{v}} = \underbrace{\left[\begin{array}{ccccc|c} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 1 \end{array}\right]}_{\mathbf{B}}\underbrace{\left[\begin{array}{c} i_1 \\ i_2 \\ i_5 \\ \hline i_6 \end{array}\right]}_{\mathbf{u}}$$

Observe that the row/column sums of $\mathbf{G}$, $\mathbf{C}$ are zero, since the network is ungrounded. Before reduction, we ground the circuit by choosing a reference node, for example node 6. Hence the corresponding row, column, and input incidence are removed from (6.26) [shown in red in (6.26)]. The grounded system is thus obtained:

$$\left\{\begin{array}{rcl} (\mathbf{G}_g + s\mathbf{C}_g)\mathbf{v}_g & = & \mathbf{B}_g\mathbf{u}_g \\ \mathbf{y}_g & = & \mathbf{B}_g^T\mathbf{v}_g \end{array}\right. \tag{6.27}$$

We reduce (6.27) with the well-known method, PRIMA [71], and apply the terminal transformation and ground recovery proposed in Sect. 6.4. Note the original, grounded system dimensions: $\mathbf{G}_g, \mathbf{C}_g \in \mathbb{R}^{5\times5}$, $\mathbf{B}_g \in \mathbb{R}^{5\times3}$.

Let $\mathbf{V}$ be the projection which matches one moment of (6.27) at $s_0 = 10^2$, $\mathbf{V} = (\mathbf{G}_g + s_0\mathbf{C}_g)^{-1}\mathbf{B}_g$, $\mathbf{V} \in \mathbb{R}^{5\times3}$. The reduced matrices are $\widehat{\mathbf{G}}_g = \mathbf{V}^T\mathbf{G}_g\mathbf{V} \in \mathbb{R}^{3\times3}$, $\widehat{\mathbf{C}}_g = \mathbf{V}^T\mathbf{C}_g\mathbf{V} \in \mathbb{R}^{3\times3}$, $\widehat{\mathbf{B}}_g = \mathbf{V}^T\mathbf{B}_g \in \mathbb{R}^{3\times3}$. Note that $\widehat{\mathbf{B}}_g$ is *dense*. Next, the transformed reduced model (6.25) is computed so that the new $\widehat{\mathbf{B}}_g = \mathbf{I}_3$. The new reduced model as in (6.25) has the form:

$$\underbrace{\left(\begin{bmatrix}\widehat{g}_{1,1} & \widehat{g}_{1,2} & \widehat{g}_{1,3}\\ \widehat{g}_{1,2} & \widehat{g}_{2,2} & \widehat{g}_{2,3}\\ \widehat{g}_{1,3} & \widehat{g}_{2,3} & \widehat{g}_{3,3}\end{bmatrix} + s\begin{bmatrix}\widehat{c}_{1,1} & \widehat{c}_{1,2} & \widehat{c}_{1,3}\\ \widehat{c}_{1,2} & \widehat{c}_{2,2} & \widehat{c}_{2,3}\\ \widehat{c}_{1,3} & \widehat{c}_{2,3} & \widehat{c}_{3,3}\end{bmatrix}\right)}_{\widehat{\mathbf{C}}_g}\underbrace{\begin{bmatrix}V_1(s)-V_6(s)\\ V_2(s)-V_6(s)\\ V_5(s)-V_6(s)\end{bmatrix}}_{\widehat{\mathbf{v}}_g} = \underbrace{\begin{bmatrix}1&0&0\\0&1&0\\0&0&1\end{bmatrix}}_{\widehat{\mathbf{B}}_g}\underbrace{\begin{bmatrix}i_1\\i_2\\i_5\end{bmatrix}}_{\mathbf{u}_g} \quad (6.28)$$

In (6.28), $\widehat{\mathbf{v}}_g$ is expressed as a voltage drop measured between each node and the reference node V_6 which was set to ground. Since the transformations $\mathbf{V}$ and $\mathbf{T}$ preserve symmetry, the off-diagonal entries satisfy $\widehat{g}_{j,i} = \widehat{g}_{j,i}$, $\widehat{c}_{j,i} = \widehat{c}_{j,i}$, for $j \neq i$, $1 \leq i, j \leq 3$.

Rewriting (6.28) so that V_6 becomes a separate unknown gives:

$$\left(\begin{bmatrix}\widehat{g}_{1,1} & \widehat{g}_{1,2} & \widehat{g}_{1,3} & -\sum_{i=1}^{3}\widehat{g}_{1,i}\\ \widehat{g}_{2,1} & \widehat{g}_{2,2} & \widehat{g}_{2,3} & -\sum_{i=1}^{3}\widehat{g}_{2,i}\\ \widehat{g}_{3,1} & \widehat{g}_{3,2} & \widehat{g}_{3,3} & -\sum_{i=1}^{3}\widehat{g}_{3,i}\end{bmatrix} + s\begin{bmatrix}\widehat{c}_{1,1} & \widehat{c}_{1,2} & \widehat{c}_{1,3} & -\sum_{i=1}^{3}\widehat{c}_{1,i}\\ \widehat{c}_{1,2} & \widehat{c}_{2,2} & \widehat{c}_{2,3} & -\sum_{i=1}^{3}\widehat{c}_{2,i}\\ \widehat{c}_{1,3} & \widehat{c}_{2,3} & \widehat{c}_{3,3} & -\sum_{i=1}^{3}\widehat{c}_{3,i}\end{bmatrix}\right)\begin{bmatrix}V_1(s)\\V_2(s)\\V_5(s)\\V_6(s)\end{bmatrix} = \begin{bmatrix}1&0&0\\0&1&0\\0&0&1\end{bmatrix}\begin{bmatrix}i_1\\i_2\\i_5\end{bmatrix} \quad (6.29)$$

Notice how in (6.29) the matrix row sums are zero. Let:

$$\widehat{g}_{j,4} = -\sum_{i=1}^{3}\widehat{g}_{j,i} \quad , \quad \widehat{c}_{j,4} = -\sum_{i=1}^{3}\widehat{c}_{j,i}, \quad \text{for } j \in \{1,2,3\} \quad (6.30)$$

From the symmetry property and (6.30), summing the rows of (6.29) gives the additional equation:

$$\begin{aligned}&\sum_{i=1}^{3}\widehat{g}_{1,i}V_1 + \sum_{i=1}^{3}\widehat{g}_{2,i}V_2 + \sum_{i=1}^{3}\widehat{g}_{3,i}V_5 + (\widehat{g}_{1,4}+\widehat{g}_{2,4}+\widehat{g}_{3,4})V_6 + \dots\\ +s&\left[\sum_{i=1}^{3}\widehat{c}_{1,i}V_1 + \sum_{i=1}^{3}\widehat{c}_{2,i}V_2 + \sum_{i=1}^{3}\widehat{c}_{3,i}V_5 + (\widehat{c}_{1,4}+\widehat{c}_{2,4}+\widehat{c}_{3,4})V_6\right] = i_1+i_2+i_5 \Leftrightarrow \quad (6.31)\\ &\Leftrightarrow \widehat{g}_{1,4}V_1 + \widehat{g}_{2,4}V_2 + \widehat{g}_{3,4}V_5 - (\sum_{i=1}^{3}\widehat{g}_{i,4})V_6 + \dots\\ &+s\widehat{c}_{1,4}V_1 + s\widehat{c}_{2,4}V_2 + s\widehat{c}_{3,4}V_5 - s(\sum_{i=1}^{3}\widehat{c}_{i,4})V_6 = -(i_1+i_2+i_5) \quad (6.32)\end{aligned}$$

Denote:

$$\widehat{g}_{4,4} = -(\sum_{i=1}^{3} \widehat{g}_{i,4}) \quad , \quad \widehat{c}_{4,4} = -(\sum_{i=1}^{3} \widehat{c}_{i,4}) \quad , \quad i_6 = -(i_1 + i_2 + i_5) \tag{6.33}$$

Appending (6.32) to (6.29) and using (6.33) gives a new system of equations:

$$\underbrace{\left(\begin{bmatrix} \widehat{g}_{1,1} & \widehat{g}_{1,2} & \widehat{g}_{1,3} & \widehat{g}_{1,4} \\ \widehat{g}_{1,2} & \widehat{g}_{2,2} & \widehat{g}_{2,3} & \widehat{g}_{2,4} \\ \widehat{g}_{1,3} & \widehat{g}_{2,3} & \widehat{g}_{3,3} & \widehat{g}_{3,4} \\ \widehat{g}_{1,4} & \widehat{g}_{2,4} & \widehat{g}_{3,4} & \widehat{g}_{4,4} \end{bmatrix} + s \begin{bmatrix} \widehat{c}_{1,1} & \widehat{c}_{1,2} & \widehat{c}_{1,3} & \widehat{c}_{1,4} \\ \widehat{c}_{1,2} & \widehat{c}_{2,2} & \widehat{c}_{2,3} & \widehat{c}_{2,4} \\ \widehat{c}_{1,3} & \widehat{c}_{2,3} & \widehat{c}_{3,3} & \widehat{c}_{3,4} \\ \widehat{c}_{1,4} & \widehat{c}_{2,4} & \widehat{c}_{3,4} & \widehat{c}_{4,4} \end{bmatrix}\right)}_{\widehat{\mathbf{G}}} \underbrace{\begin{bmatrix} V_1(s) \\ V_2(s) \\ V_5(s) \\ V_6(s) \end{bmatrix}}_{\widehat{\mathbf{v}}} = \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}}_{\widehat{\mathbf{B}}} \underbrace{\begin{bmatrix} i_1 \\ i_2 \\ i_5 \\ i_6 \end{bmatrix}}_{\mathbf{u}} \tag{6.34}$$

System (6.34) represents the *ungrounded reduced model* with node 6 re-inserted. It can be readily verified that the rows/columns of $\widehat{\mathbf{G}}$, $\widehat{\mathbf{C}}$ respectively add up to zero. Hence the reduced netlist with all terminals recovered is obtained as usually by unstamping (6.34) with RLCSYN (see chapter 2). In particular the resistor and capacitor values between nodes V_i and V_j, $i, j \in \{1, 2, 5, 6\}$ are:

$$r(V_1, V_2) = -\frac{1}{\widehat{g}_{1,2}}, \quad r(V_1, V_5) = -\frac{1}{\widehat{g}_{1,3}}, \quad r(V_1, V_6) = -\frac{1}{\widehat{g}_{1,4}} \tag{6.35}$$

$$c(V_1, V_2) = -\widehat{c}_{1,2}, \quad c(V_1, V_5) = -\widehat{c}_{1,3}, \quad c(V_1, V_6) = -\widehat{c}_{1,4} \tag{6.36}$$

As the row/column sums of (6.34) are zero, there are no connections from any of the nodes to ground. The "ground" was the reference node V_6 which is mapped back as a terminal.

A final intuitive note regarding the current equation from (6.33); this is the Kirchhoff's current law for the reduced net, which says that the sum of all currents entering the network is zero. This holds for the original network as well [just sum the rows of the original ungrounded MNA system from Fig. 6.26 to get $i_6+i_2+i_3+i_5=0$]. This can be visualized on the right of Fig. 6.2.

Finally, a comparison between the AC simulation of the original and reduced netlist for this simple circuit is shown in Fig. 6.3. As this is a very small example, the quality of the approximation is not of major interest (although one can observe that the reduced response follows the original well especially for higher frequencies). More important is the synthesis result: the mathematical reduced model was converted to a simple *RC* representation without controlled sources and with all terminals recovered. Actually, during the AC simulation itself, node V_1 was set to ground, and the response at node V_6 was measured (shown in Fig. 6.3). This confirms that choosing a reference node prior to the reduction step can be done arbitrarily. Provided that this reference node is re-inserted as a terminal during synthesis, the response at all terminal nodes will be recovered.

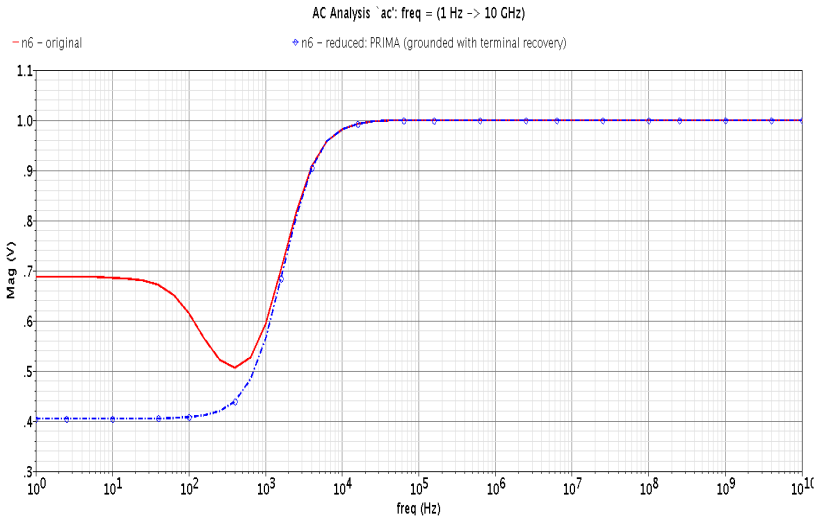


Figure 6.3: *AC analysis of the simple RC circuit. Original (red, dimension* 6*) vs. reduced with terminal recovery (blue, dimension* 4*). The voltage at node n6 (V_6) is measured.*

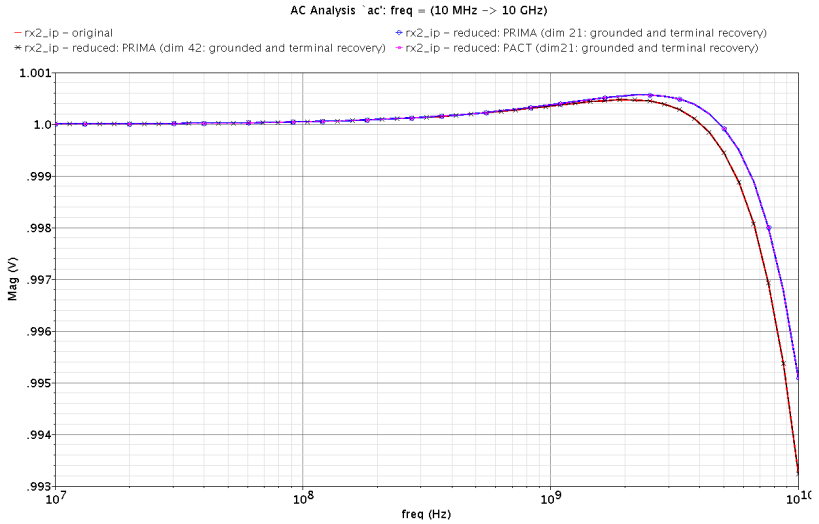


Figure 6.4: *Example 6.5.2. AC simulation of original (red,* 3253 *nodes), PACT reduced (magenta,* 22 *nodes), PRIMA reduced (blue,* 22 *nodes), PRIMA reduced (black,* 43*) nodes.*

6.5.2 *RC* transmission line

An *RC* transmission line with $n + p = 3253$ nodes of which 22 are terminals is reduced and synthesized in this generalized framework using three methods: PACT (see

Sect. 4.2.2), PRIMA [71] and the Loewner method [60]. Note that, while PACT applies by default to ungrounded systems and also preserves automatically the structure of the input/output matrices, the more general methods PRIMA and the Loewner method satisfy none of these properties. Nevertheless, using the framework proposed in this chapter, they also can be applied to multi-terminal systems. This is shown next.

The original network is ungrounded, and the procedure follows in the same manner as for the small example in Sect. 6.5.1. As before, from the 22 terminals, one node is arbitrarily chosen as a reference node and the grounded system (6.27) (now with 21 terminals) is obtained. The three reduction methods PACT (see Sect. 4.2.2), PRIMA [71] and the Loewner method [60] are applied on the grounded system. The reducing projections are constructed to match moments of the original response around $s = 0$. Because the reduced models obtained by PRIMA and the Loewner method do not preserve the input/output structure, they are transformed via the input/output transformation (6.24) to (6.25) [the PACT reduced model does not need the input/output transformation, since it naturally preserves the input/output structure]. Note that all reduced models are of the form (6.28), where the states are the voltages measured with respect to the grounded node. As in Sect. 6.5.1, for each reduced system the ungrounded reduced representation is obtained, with the reference node re-inserted as the 22'nd terminal. Finally, the synthesized netlist without controlled sources is obtained by RLCSYN unstamping (see Chapter 2). Figure 6.4 shows the comparison between the AC simulations (with Spectre [16]) of the synthesized netlists for the PACT model and two PRIMA-reduced models. The PACT model and the small PRIMA model have dimension 21 (22 nodes after terminal recovery) and are superimposed. These models deviate very little from the original and only beyond 1GHz. The larger PACT model of dimension 42 (43 nodes in the synthesized netlist) follows the original perfectly.

Two reduced models of dimension 21 and 42 [(LM1) and (LM2) respectively] were also obtained[3] with the Loewner method [60] by taking measurements of the original frequency response (of the grounded system). The maximum and minimum singular values $\sigma_{max}(\widehat{\mathbf{H}}_g(s))$ and $\sigma_{min}(\widehat{\mathbf{H}}_g(s))$ of the reduced models are compared to those of the original, $\sigma_{max}(\mathbf{H}_g(s))$ and $\sigma_{min}(\mathbf{H}_g(s))$. For LM1, Fig. 6.5 shows that the approximation for low frequencies may not be sufficiently accurate. The singular value plots for LM2 show, in Fig. 6.6, that this larger model provides a good approximation for the whole frequency range. Again, this is confirmed by the re-simulation result in Fig. 6.7. The AC simulations of the reduced synthesized models compared to the original simulation are shown in Fig. 6.7. The smaller model (LM1 with 22 nodes after synthesis) shows a slight mismatch in DC, which is expected given the result obtained in Fig. 6.5. However the larger model (LM2 with 43 nodes after synthesis) matches the original.

[3] The Loewner reduction itself was performed by the first author of [60].

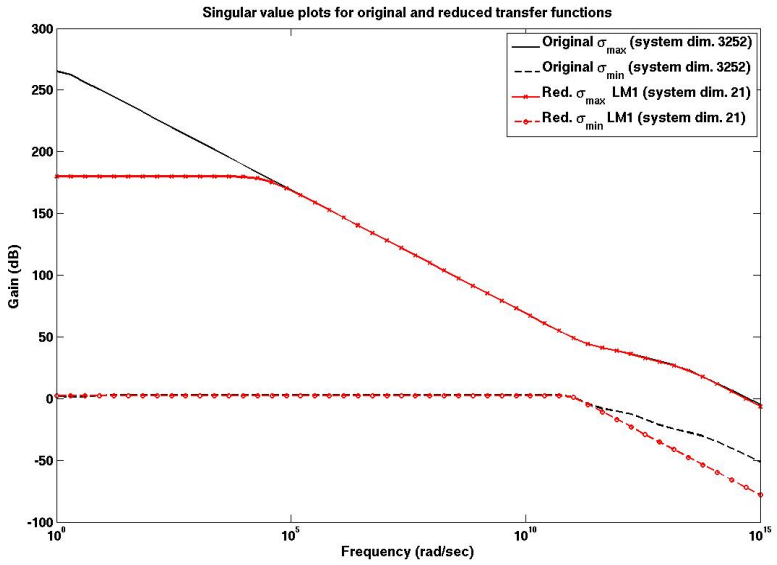


Figure 6.5: *Example 6.5.2. Maximum and minimum singular value plots for original (black,* 3252 *nodes), and reduced LM1 (red,* 21 *nodes).*

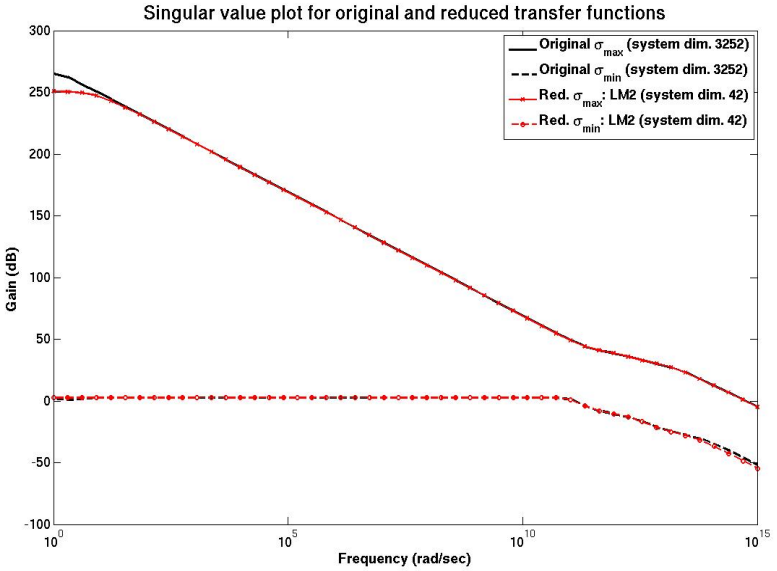


Figure 6.6: *Example 6.5.2. Maximum and minimum singular value plots for original (black,* 3252 *nodes), and reduced LM2 (red,* 42 *nodes)*

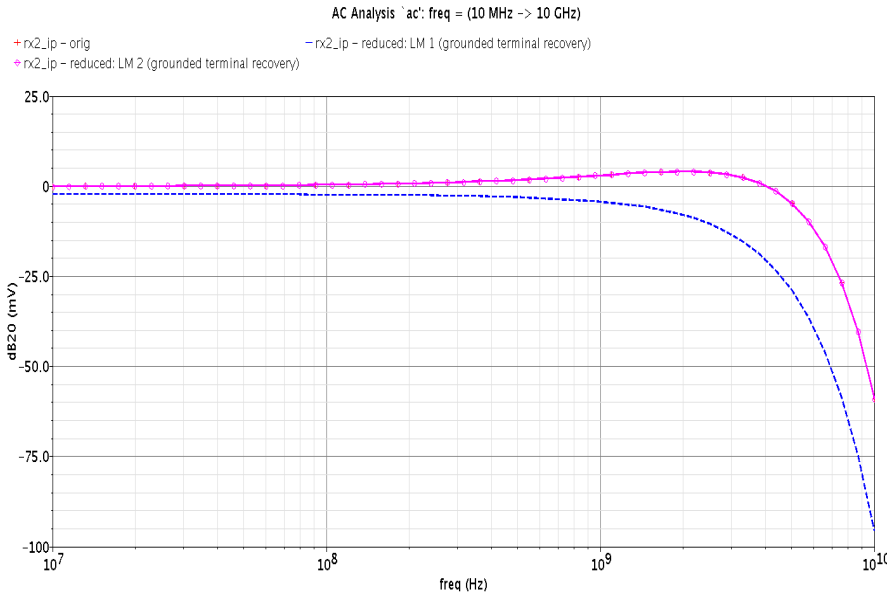


Figure 6.7: *Example 6.5.2. AC simulation of original (red,* 3253 *nodes), and two reduced netlist obtained from measurements [60]: LM1 (blue,* 22 *nodes) and LM2 (magenta,* 43 *nodes). The original and LM2 are indistinguishable.*

6.6 Concluding remarks

Two problems are resolved, which usually limit the applicability of traditional reduction methods to multi-terminal circuits: (a) the reduction of ungrounded systems and (b) the synthesis of general reduced models without controlled sources. A terminal removal and recovery action is proposed, which allows the reduction of ungrounded multi-terminal models. The synthesis problem is resolved using a transformation based on the input/output matrices of the reduced model. The framework enables the reduction of multi-terminal circuits with conventional MOR methods, and the usability of such reduced order models in simulations.

Chapter 7

Graph partitioning with separation of terminals

A graph partitioning problem is described, specifically targeted towards improving sparsity during the reduction of multi-terminal circuits. The traditional partitioning objectives are revised and new conditions are derived, aimed at directly minimizing fill-in. Upper bounds on fill-in are provided, which could serve as initial criteria for distributing nodes and terminals across partitions. Examples are provided to also illustrate that the upper bounds may be too loose to achieve the best sparsity results. The analysis motivates the need for a fill-estimating measure to be used directly as a partitioning objective. If incorporated in the new partitioning setup, the fill-minimizing objectives may improve sparsity in multi-terminal MOR beyond the level achieved via existing partitioning software.

7.1 Introduction

Graph partitioning has become the key ingredient in developing efficient solutions for large scale problems in many disciplines, with application areas ranging from technological sciences to natural and social sciences [23]. Put in simple terms, the objective of graph partitioning in general is to distribute a large data set into different parts that communicate with each-other as little as possible, usually so that the resulting parts are equal in size. For applications where these represent the governing criteria (e.g. parallel computing or sparse matrix factorizations [18]), state-of-the-art partitioning software (to name a few: METIS [52], hMETIS [51], Mondriaan [96]) successfully apply.

In the context of multi-terminal model reduction where the ultimate goal is to preserve

sparsity in the reduced model, a new partitioning problem emerges: *find the partitioning which distributes terminals across components so that the fill generated from eliminating the internal nodes of each component is minimized*. As in the usual partitioning setup, components should be minimally connected to one-another, so as to *preserve as few internal nodes (i.e. separator nodes) as possible* (the fewer the preserved internal nodes, the smaller the dimension of the reduced model).

Chapter 4 showed how, using graph partitioning, very large multi-terminal networks were split into components which were treated individually in subsequent reduction steps. The partitioning helped to efficiently obtain, from an original large circuit, a reduced circuit which is "small and sparse" [i.e., its graph representation ideally has fewer vertices (circuit nodes) and fewer edges (basic circuit elements) than the original]. It was also seen how the sizes of the different parts do not influence the approximation quality of the reduced model. However, as demonstrated next, the number of terminals falling in each part and the number of separator nodes influence the sparsity level achieved during reduction.

The purpose of this chapter is to provide an avenue for improving sparsity in multi-terminal MOR, even beyond the level achieved via existing partitioning software. A new partitioning criterion is proposed which eliminates some of the standard constraints (such as enforcing equal-sized partitions) and rather directly minimizes a fill-in objective, in addition to the usual partitioning objectives of minimizing (a) the edge cut or (b) the total communication volume[1]. See [52] for a definition of these objectives. We give an abstract formulation of the objective function which distributes terminals across partitions, so that fill-in generated from reducing each partition up to terminals is minimized. If incorporated inside partitioning tools, the terminal constraint could further enhance the reduction of challenging multi-terminal electrical circuits.

7.2 Formulation of multi-terminal partitioning criterion

In the following section, we formulate the multi-terminal partitioning criterion for a bisection. Afterwards a generalized formula for a partitioning into N components is derived.

7.2.1 2-way partitioning of multi-terminal networks

Fig. 4.2 provides a rough problem visualization and formulates the partitioning task. More precisely, let the undirected graph G be partitioned into two *subnets* $G_1 = (V_1, E_1)$, $G_2 = (V_2, E_2)$, which communicate via a *separator net* $G_3 = (K, E_K)$. In other words, the nodes V are partitioned into three disjoint sets $V_1 \cup V_2 \cup K = V$, such that there is no

[1] Minimizing the total communication volume becomes relevant especially for partitions of size > 2.

direct edge between any node from V_1 to a node from V_2, and the only path from a node of V_1 to a node from V_2 is through the separator nodes K. Denoting $|V_1| = N_1$, $|V_2| = N_2$, and $|K| = k$, the edges induced by V_1 are $E_1 = \{(v_i, v_j) \in E |\ v_i, v_j \in V_1;\ i \neq j\}$, by V_2 are $E_2 = \{(v_i, v_j) \in E |\ v_i, v_j \in V_2;\ i \neq j\}$, and by K are $E_K = \{(v_i, v_j) \in E |\ v_i, v_j \in K;\ i \neq j\}$. Let the remaining edges connecting K to V_1 be $E_{1K} = \{(v_i, v_j) \in E | v_i \in V_1;\ 1 \leq i \leq N_1,\ v_j \in K;\ 1 \leq j \leq k\}$, and the edges connecting K to V_2 be $E_{2K} = \{(v_i, v_j) \in E | v_i \in V_2;\ 1 \leq i \leq N_2,\ v_j \in K;\ 1 \leq j \leq k\}$. The complete set of edges is thus $E = E_1 \cup E_2 \cup E_K \cup E_{1K} \cup E_{2K}$, and the individual sets are disjoint. Through this partitioning, the terminals are split into the following disjoint sets: $P_1 \subset V_1$ terminals fall under V_1, $P_2 \subset V_2$ terminals fall under V_2, and $P_K \subset K$ terminals[2] fall inside the separator K, $|P_1| = p_1$, $|P_2| = p_2$, $|P_K| = p_k$, $P_1 \cup P_2 \cup P_K = P$.

Let there be n_1 internal nodes in G_1, and n_2 in G_2, so that $|V_1| = p_1 + n_1$, $|V_2| = p_2 + n_2$. Figure 7.1 (left) gives the representation of a matrix $\mathbf{G}$ partitioned in this manner, which is to be reduced by eliminating internal nodes n_1 and n_2. As the reduction progresses through the BBD hierarchy, fill is introduced in the blocks corresponding to the preserved nodes p_1, p_2 and k (as was also described in Chapter 4). For instance the reduced $\widehat{\mathcal{G}}$ of (4.31) numerically represents the Schur complement with respect to the blocks of *eliminated internal nodes*, and clearly will have different dimensions and fill rates depending on the p_1, n_1, p_2, n_2, and k nodes resulting from each partition. The form of the reduced matrix is shown on the right of Fig. 7.1. In the worst case, all matrix blocks ($\mathbf{G}_1$, $\mathbf{G}_2$, $\mathbf{G}_3$, $\mathbf{G}_{13}$, $\mathbf{G}_{23}$) in the reduced matrix are completely filled. In this setup, the 2-way multi-terminal partitioning problem is formulated:

Problem 1: multi-terminal 2-way partitioning
Find the smallest number of separator nodes k which partitions G into two parts such that the elimination of internal nodes n_1 and n_2, introduces the smallest amount of fill in the terminal blocks p_1, p_2, the separator block k, and the communication block from p_1 and p_2 to k respectively.

Towards formalizing the above partitioning criterion, let:

$$e_{fill_1}(n_1, [p_1, k]), \tag{7.1}$$

be an abstract function which estimates the fill created during the reduction of the first subnet. More precisely, the notation in (7.1) specifies the fill generated from the elimination of internal nodes n_1 inside the blocks corresponding to terminal nodes p_1 and separator nodes k. Similarly, a fill-estimating function for the reduction of the second subnet is $e_{fill_2}(n_2, [p_2, k])$. The partitioning objective of **Problem 1** can be written compactly as:

$$\min_{\min(k),\ k=|K| \subset V} e_{fill_1}(n_1, [p_1, k]) + e_{fill_2}(n_2, [p_2, k]), \tag{7.2}$$

where the additional constraint $min(k)$ specifies that a small number k of separator

[2] For multi-terminal MOR the p_k terminals are not relevant, as all the k separator nodes become terminals.

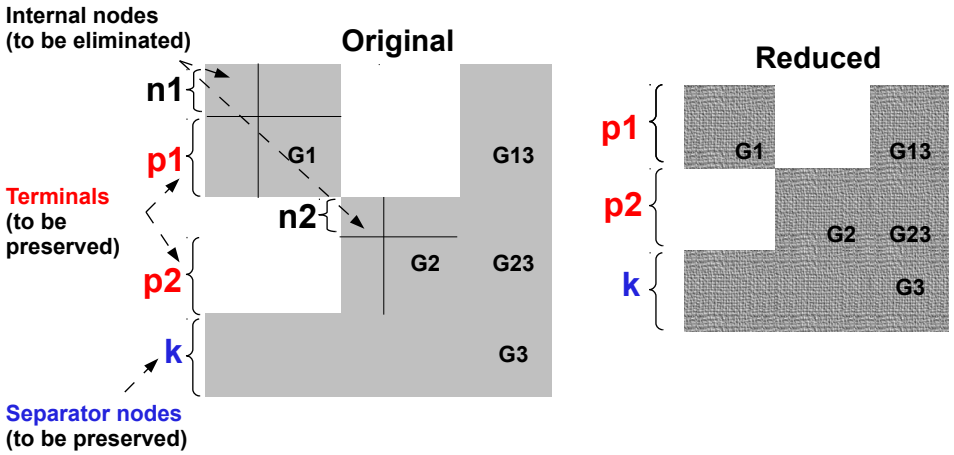


Figure 7.1: *Original and reduced circuit matrices for a two-way partitioning in BBD form. In the reduction step, internal nodes are eliminated, and blocks corresponding to terminal nodes and separator nodes are preserved but filled-in. The dimensions of the reduced model give the maximum fill value from (7.3). [A note on scale: in practice the number of internal nodes is larger than the number of terminals per block, but is kept small in the figure due to space limitations.]*

nodes is desired (as all separator nodes are preserved in the reduced model, the smaller the k the smaller is also the dimension of the reduced model). An upper bound for the fill-estimating function (7.2) is obtained immediately by inspection from Fig. 7.1 (right):

$$\min_{\min(k),\, k=|K|\subset V} \underbrace{\frac{p_1(p_1-1)}{2}+\frac{p_2(p_2-1)}{2}+\frac{k(k-1)}{2}+(p_1+p_2)k}_{Maxfill}, \tag{7.3}$$

which represents the *maximum fill-in* possibly generated from reducing each subnet individually up to its terminals p_i and to the k separator nodes. More precisely, the upper bound on e_{fill_1} is $\frac{p_1(p_1-1)}{2}+p_1k+\frac{k(k-1)}{2}$, the upper bound on e_{fill_2} is $\frac{p_2(p_2-1)}{2}+p_2k+\frac{k(k-1)}{2}$, so that the upper bound on (7.2)[3] becomes (7.3). Note that (7.3) is independent of the internal nodes n_1 and n_2. Hence in practice it may be a loose upper bound (this effect is shown in Fig. 7.8 for the more general case of an N-way partition).

It should be emphasized that in the multi-terminal partitioning setup above defined, the resulting subnet sizes N_1 and N_2 need not be the same. In other words, the partitioning must not be *balanced* in the dimension of the generated components (decompositions into equal parts pertain more to parallel computing related applications, where the main goal is to balance computation among processors). In the context of multi-terminal

[3]The upper bound on the sum of the two $\frac{k(k-1)}{2}$ factors is still $\frac{k(k-1)}{2}$, as the fill inside the upper triangle of a $k \times k$ matrix can never be greater than $\frac{k(k-1)}{2}$.

MOR, partitions should be balanced in a different measure, that which minimizes fill-in. For that, terminals must be appropriately distributed across the subnets, together with the usual condition of minimizing connectivity between subnets. Therefore significant improvements in sparsity could be achieved with the help of a partitioner which does not impose an equal-sized decomposition, but rather uses condition (7.2) to directly control the distribution of terminals. This remains a motivating problem of future research, as is the formulation of an explicit measure (7.2) [tighter than (7.3)] for estimating fill-in. Closely related is the *approximate minimum degree* criterion which underlies the AMD [1, 2] reordering algorithm for sparse matrix factorizations. AMD uses upper bounds on the degrees of the nodes in the graph, rather than exact degrees, to determine the order in which nodes should be eliminated from a graph (matrix) so that the least amount of fill-in is produced. The pivot selection at each step is done by choosing the node with the smallest upper bound on the degree [1, 2]. Hence, an estimator for (7.1) could use the approximate minimum degree criterion to determine which n_1 internal nodes should be eliminated and what the predicted fill-in would be.

Example for 2-way partitioning

To understand the influence of different partitions on the sparsity of the reduced model, the biconnected component network from [80] is chosen. In Fig. 7.2 (left) the graph representation of the original network is shown: it has $n + p = 244$ nodes of which $p = 54$ are terminals (shown in red). The nodes of the graph are actual circuit nodes, while the edges between nodes correspond to the resistive connections. The network has visible biconnected components (if the articulation nodes in the middle are removed, the two parts become disconnected).

Three partitioning choices are made (denoted as **P 1**, **P 2**, and **P 3** respectively), each with different separator nodes. For each partitioning scenario, the statistics are recorded in Table 7.1: $N_1 = n_1 + p_1$ and $N_2 = n_2 + p_2$ are the sizes of each subnet, p_1 and p_2 are the number of terminals falling in each partition, k is the number of separator nodes, *Maxfill* is the corresponding upper bound on fill-in (7.3). Reduced networks are obtained from each partition and the *true fill-in*, i.e. the number of resistors in the reduced circuit is recorded and compared to the *Maxfill*.

Table 7.1: *Comparison of different partitions and reductions for the bi-component circuit*

	P 1 *Fig. 7.2*		**P 2** *Fig. 7.3*		**P 3** *Fig. 7.4*	
	#nodes	#terminals	#nodes	#terminals	#nodes	#terminals
Subnet 1	$N_1 = 60$	$p_1 = 1$	$N_1 = 120$	$p_1 = 37$	$N_1 = 79$	$p_1 = 27$
Subnet 2	$N_2 = 183$	$p_2 = 53$	$N_2 = 119$	$p_2 = 17$	$N_2 = 153$	$p_2 = 27$
Separator	$k = 1$	$p_k = 0$	$k = 5$	$p_k = 0$	$k = 12$	$p_k = 0$
Maxfill	1432		1082		1416	
True fill	1432		1031		828	

The graph representations for the original and reduced network in each partitioning scenario are shown in Figures 7.2, 7.3 and 7.4 respectively. The corresponding reduced $\widehat{\mathcal{G}}$ matrices appear in Fig. 7.5.

For **P 1** one articulation node is chosen as the separator K, shown in blue in Fig. 7.2 (left). The reduced model (right) is dense: there is an edge (resistor element) in between almost all pairs of nodes. The reduced matrix (left of Fig. 7.5) clearly shows the fill-in. The *Maxfill* (1432) and *True fill* (1432) values in Table 7.1 are the same.

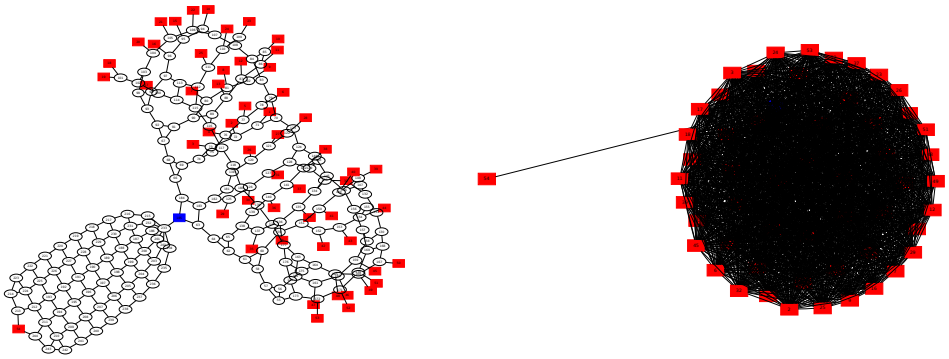


Figure 7.2: ***P 1****. Original circuit (left) with a biconnected component (terminals shown in red). The connection node (blue) is picked as separator. Circuit after reduction with all terminals and separator node preserved (right) is dense (*55 *nodes,* 1432 *resistors).*

The partitioning **P 2** is obtained with *Nested dissection (NESDIS)* [18], [52, 54], and picks the five separator nodes shown in blue in Fig. 7.3, left. The corresponding reduced network (Fig. 7.3, right) is sparser than the one from **P 1**. This is also reflected by the reduced matrix (middle of Fig. 7.5), which is more sparse. As there are 1031 resistors in this reduced net, **P 2** is a better partition than **P 1**.

For the **P 3** partitioning, separator nodes were chosen as shown in Fig. 7.4 (left). The resulting reduced model on the right is sparser than from **P 2** and **P 1**, also reflected by the reduced matrix in Fig. 7.5, right. There are 66 nodes and 828 resistors in this reduced network.

Comparing the number of resistors (*True fill*) in the reduced networks, it is clear that **P 3** is the partitioning which yields the reduced model with the least fill-in. It should be noted that this partition (and selection of separator nodes) was done by inspection, rather than by a partitioning algorithm (as was the case for **P 2** which was obtained from NESDIS). This supports the initial conjecture that in contrast to the usual partitioning objectives, partitioning for multi-terminal circuits requires a specialized criterion which directly minimizes fill-in.

Another important observation confirmed by this example is that the *Maxfill* value (7.3) is indeed only an upper bound which may be too loose to consider as a partitioning

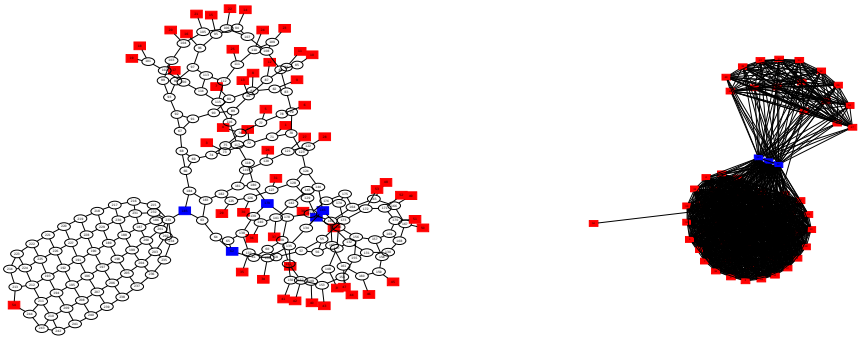


Figure 7.3: ***P 2****. Original circuit (left) with a biconnected component (terminals shown in red). Separator nodes (blue) are the ones given by NESDIS. Circuit after reduction (59 nodes, 1031 resistors) with all terminals and separator nodes preserved (right) is sparser than the one from Fig. 7.2, but still rather dense.*

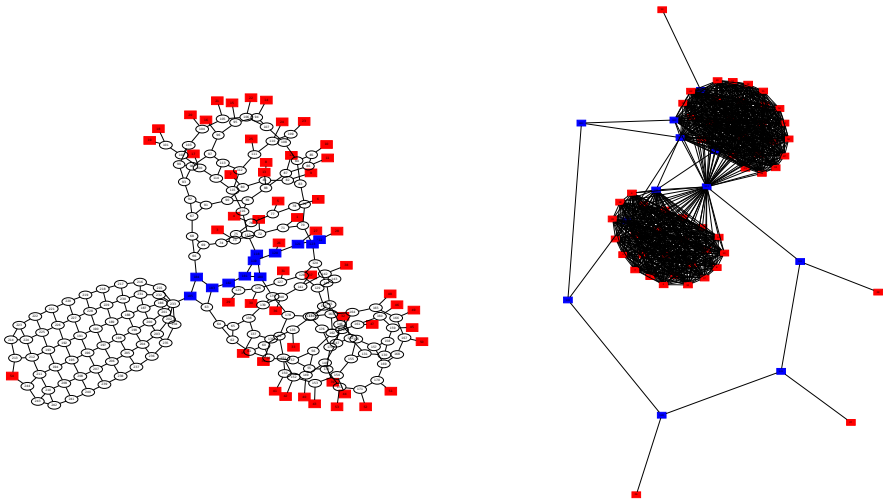
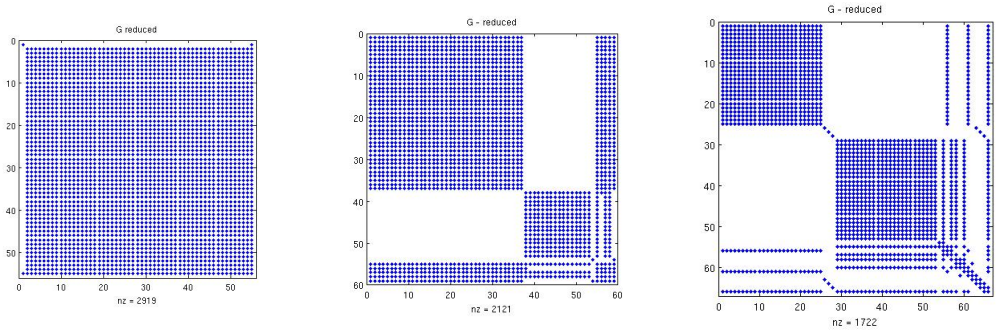


Figure 7.4: ***P 3****. Original circuit (left) with a biconnected component (terminals shown in red). More nodes are picked as separators (blue). Circuit after reduction (66 nodes, 828 resistors) with all terminals and separator nodes preserved (right) is sparser than the ones from Fig. 7.2 and Fig. 7.3.*

objective in practice. Notice in Table 7.1 that although from **P 3** a sparser reduced model was obtained than from **P 2** (828 vs. 1031 resistors respectively), partitioning **P 3** has a higher *Maxfill* value than **P 2** (1416 vs. 1082 respectively). If the *Maxfill* criterion (7.3) were used as a partitioning criterion, **P 2** would be considered a better partition than **P 3**, which was clearly not the case. Hence for better results, a partitioning criterion based on a tighter fill-estimating function [denoted abstractly as (7.2)] is necessary, which accounts for the degrees of the internal nodes to be eliminated as well, rather than for the number of terminals and separator nodes alone.



P 1 ***P 2*** ***P 3***

Figure 7.5: *Reduced matrices resulting from each partition (**P 1**, **P 2**, **P 3**, from left to right). These correspond to the reduced circuits on the right of Figures 7.2 (*55 *nodes,* 1432 *resistors), 7.3 (*59 *nodes,* 1031 *resistors) and 7.4 (*66 *nodes,* 828 *resistors) respectively.*

A final reduced model is computed based on partition **P 1**, where the nodes in each subnet are additionally re-ordered with CAMD [1,2]. From each subnet, internal nodes are eliminated one by one in the order determined by CAMD, and the fill generated in the entire matrix is monitored at each step. These actions were described in Chapter 4 (see Sect. 4.2.3, Sect. 4.3.2 and for instance Fig. 4.11. In this manner, the points of minimum fill in each subnet are recorded and the internal nodes beyond the minimum fill point are preserved along with the terminals to improve sparsity. The final reduced circuit is shown in Fig. 7.6 and is much sparser than the previous models (224 resistors), due to the additional internal nodes preserved (aside from terminals and separator nodes). There are 182 nodes in this circuit, only 24% less than the original 244, demonstrating the trade-off between small dimensionality and sparsity.

The last results indicate that reduced models with the least fill-in could be determined with the combined framework: partitioning + CAMD reordering per subnet + fill-tracking. For an original network (graph) $G = (V, E)$ with $|V|$ nodes and $|E|$ edges (circuit elements), the partitioning cost is $O(|E|)$ [53] and the CAMD reordering cost is $O(|V||E|)$ [1,2,38], hence very cheap to perform. The extra cost for tracking fill-in can however become too expensive, especially for networks with nodes and elements exceeding 10^4, and terminals exceeding 10^3. To monitor the actual fill-in from node-wise elimination, reduced matrices must be formed at each node elimination step, and the corresponding number of non-zero entries recorded. For example, for a subnet with $n_1 + p_1$ nodes, which communicates via k separator nodes to other subnets, n_1 internal nodes must be eliminated one by one. Hence n_1 reduced matrices must be formed, which also become denser and denser as fill is generated in the blocks corresponding to the non-eliminated nodes. The extra cost of repeatedly storing and retrieving from memory these reduced matrices may dominate the otherwise cheap partitioning and reordering operations. This effect was already shown for the **Filter** example of Chapter 4, see in particular the result in Table 4.2. There, the SpRC-mf reduced model formed with

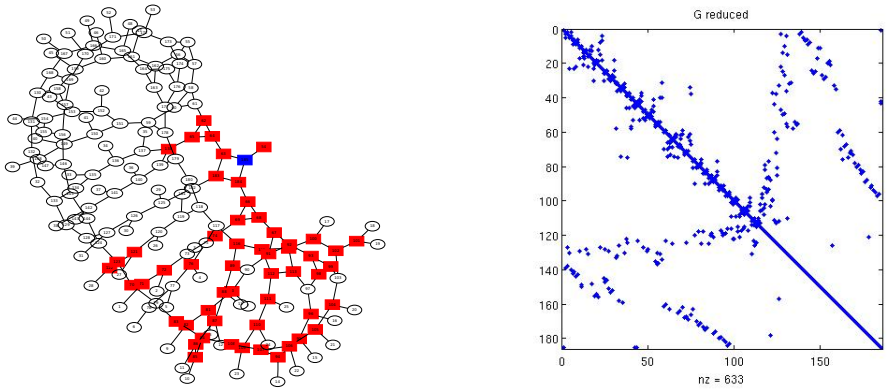


Figure 7.6: ***P 1 + CAMD****. Sparse reduced circuit (left) with the single connection node as separator (blue), but with additional internal nodes preserved [identified from CAMD re-ordering and fill-in tracking]. The corresponding sparse reduced matrix is on the right (*182 *nodes,* 224 *resistors).*

the combined framework: partitioning + CAMD reordering per subnet + fill-tracking, was indeed sparser than SpRC obtained from partitioning alone, but was almost 10 times slower to compute than SpRC.

A partitioning criterion for multi-terminal MOR is thus envisioned which uses a fill-estimating function (7.2) (similar to the approximate minimum degree criterion of [1,2]) as a direct objective to determine the best partition. In this manner reduced models with minimum-fill could be ensured already from the partitioning itself, and the extra fill-monitoring costs would be avoided.

7.2.2 N-way multi-terminal partitioning

The N-way partitioning of a graph $G = (V, E)$ is defined here as follows: find a partitioning of V into $2N-1$ disjoint subsets $V = V_1 \cup V_2 \cup \ldots \cup V_N \cup K_1 \cup K_2 \ldots \cup K_{N-1}$ where the V_i, $1 \leq i \leq N$ are the nodes of the individual subnets, such that there is no edge from any node in V_i to any node from V_j for all $i \neq j$. The K_j, $1 \leq j \leq N-1$ are the sets of separator nodes comprising the communication among the subnets induced by the V_is. This is best visualized using the example of a 4-way partitioning, shown in Fig. 7.7 in the form of a border-block-diagonal (BBD) matrix ordering together with the corresponding *separator tree*. In particular, assuming $N = 2^m$, $m \geq 1$, the N-way partitioning produces a binary tree of height m, with $N = 2^m$ leaf blocks[4] and $N-1 = 2^m - 1$ non-leaf blocks. The leaves of the tree are the individual subnets to be reduced, the non-leaves are the sets of separator nodes to be preserved. As for the 2-way partitioning, in each V_i, $1 \leq i \leq N$ there are n_i internal nodes and p_i terminals. The number of separator

[4]The traditional *node* of a tree is denoted here as a *block*, as not to mistake it for a node in the graph G itself.

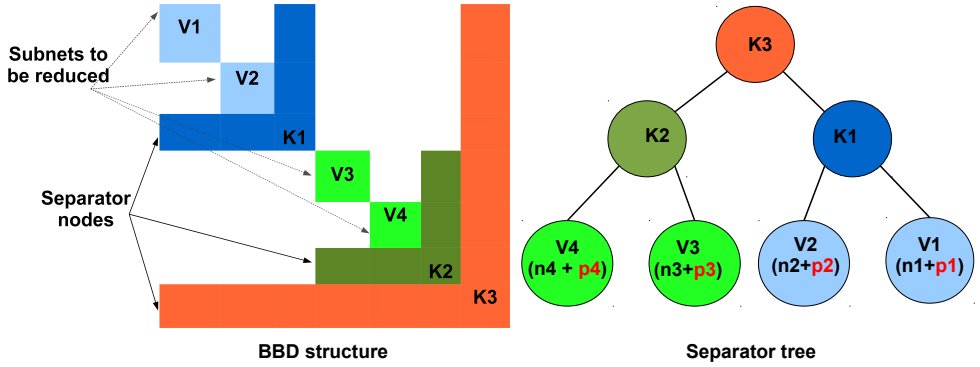


Figure 7.7: *Border block diagonal (BBD) matrix structure and separator tree for a 4-way partitioning. As $N = 4 = 2^2$, the separator tree has height $m = 2$, $N = 4$ leaves (subnets to be reduced) and $N - 1 = 3$ sets of separator nodes.*

nodes in each K_j set, $1 \leq j \leq N$, is k_j. The reduction eliminates the n_i internal nodes from the V_i subnets (the leaves of the separator tree). The multi-terminal partitioning criterion is formulated similarly as in Sect. 7.2.1:

Problem 2: multi-terminal N-way partitioning
Given a number of partitions N, find the smallest number of separator nodes $k_j, 1 \leq j \leq N-1$ which partitions G into N subnets such that the elimination of internal nodes $n_i, 1 \leq i \leq N$ from each subnet introduces the smallest amount of fill in the terminal blocks p_i, the separator blocks k_j, and the communication blocks from p_i to the k_j nodes.

Another natural question is determining the optimal number of partitions N itself, hence a generalized problem is formulated:

Problem 3: generalized multi-terminal partitioning
For a multi-terminal network G, find the number N of partitions which distributes the p terminals across partitions so that:

1. the number of separator nodes $k_j, 1 \leq j \leq N-1$ is minimized, and
2. the elimination of internal nodes $n_i, 1 \leq i \leq N$ from each subnet introduces the smallest amount of fill in the terminal blocks p_i, the separator blocks k_j, and the communication blocks from p_i and to the k_j nodes.

As for the 2-way partitioning, in the N-way multi-terminal partitioning the sizes of the different parts need not be equal. Rather, an objective function similar to (7.2) for estimating the fill described by **Problem 3** in the N-way case could be used directly as a partitioning criterion. For a partition into $N = 2^m$ subnets (and $N - 1$ separator blocks),

this is formulated based on the structure of the separator tree, using the following notation:

- let L be the set of leaf blocks, $|L| = N$
- let S be the set of non-leaf (i.e. separator) blocks, $|S| = N - 1$
- for a leaf block $i \in L$, let AL_i be the set of all its ancestors[5] on the path from i to the root, including the root itself, $|AL_i| = m$
- similarly, for a non-leaf (i.e. separator) block $i \in S$, let AS_i be the set of all its ancestors on the path from i to the root.

Let the following be an abstract notation for the fill generated by reducing subnet i:

$$e_{fill}(n_i, [p_i, k_j | j \in AL_i]), \tag{7.4}$$

more precisely (7.4) estimates the fill generated by eliminating the n_i internal nodes inside the matrix blocks determined by the terminal nodes p_i and all separator nodes k_j, with $j \in AL_i$ being the index of all ancestors of block i. As an example, consider the reduction of subnet 1 from the 4-way partition shown in Fig. 7.7. The maximum fill [upper bound on (7.4)] from eliminating the internal nodes n_1 is:

$$\begin{aligned} \max\{e_{fill}(n_1, [p_1, k_j | j \in AL_1])\} &= \dots \\ \dots = \frac{p_1(p_1-1)}{2} &+ p_1(k_1+k_3)+k_1k_3+\frac{k_1(k_1-1)}{2}+\frac{k_3(k_3-1)}{2} \end{aligned} \tag{7.5}$$

Summing over all subnets N, one can write compactly the partitioning objective of **Problems 2,3**:

$$\min_{\min(k),\ k=|K|\subset V} \sum_{i\in L} e_{fill}(n_i, [p_i, k_j | j \in AL_i]) \tag{7.6}$$

By similar bookkeeping with (7.5), the maximum fill generated in the reduced model determined by the terminal nodes p_i, $1 \le i \le N$ and separator nodes k_i, $1 \le i \le N-1$ from eliminating all internal nodes n_i, $1 \le i \le N$ is:

$$\sum_{i\in L}\left[\frac{p_i(p_i-1)}{2} + p_i \sum_{j\in AL_i} k_j\right] + \sum_{i\in S}\left[\frac{k_i(k_i-1)}{2} + k_i \sum_{j\in AS_i} k_j\right], \tag{7.7}$$

which is an upper bound on (7.6). For $N = 2$, the upper bound (7.7) is precisely (7.3). Similarly to the 2-way case, (7.7) serves only as an upper bound on the true fill-in if used directly as a partitioning objective in practice. This is reflected in Fig. 7.8, where the **PLL** network from Chapter 4 was reduced based on different numbers of subnets N, with the upper-bound (7.7) and the true fill-in (number of elements in the reduced network) being recorded for each N. The reduced model with minimum fill was obtained

[5] A node j is the ancestor of a node i if j is on the path from i to the root of the separator tree.

based on the partition with $N = 500$ subnets. As the plot shows, the Maxfill bound is loose (7.7) and would have predicted a much smaller, non-optimal N. Nevertheless the upper-bound serves as an indicator for keeping track of where fill-in is generated during the reduction process.

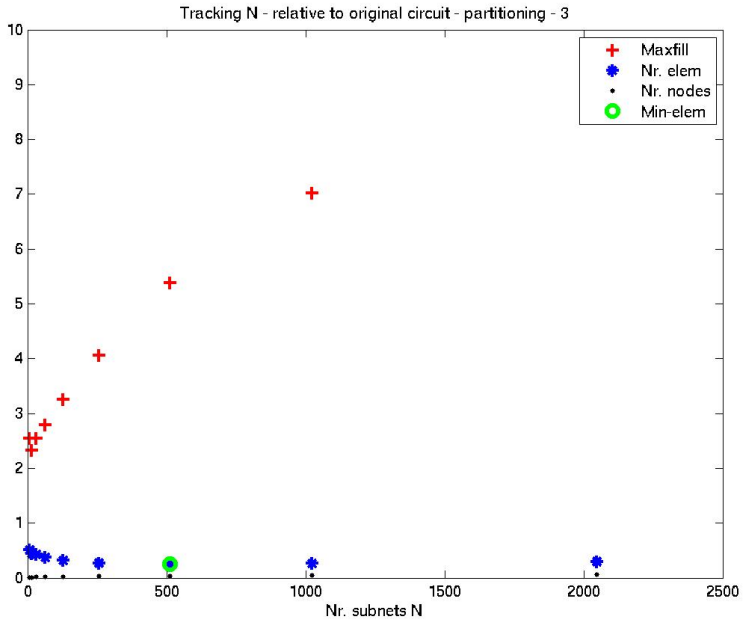


Figure 7.8: *Determining the reduced model with the fewest number of elements from an iteration of different numbers of subnets N (partitioning is done via nested dissection).*

7.3 Concluding remarks

In concluding the discussion on multi-terminal partitioning, the goals of partitioning in the context of model order reduction are reiterated:

1. efficiency: to reduce very large networks by breaking them into parts which can be handled separately, especially if reducing the full problem is either too costly or unfeasible, and
2. sparsity: to partition the network in such a way that the reduced matrices have minimum fill-in.

While the efficiency gains are easily understood, the sparsity considerations are more subtle, especially as regards the partitioning fineness, the distribution of nodes, terminals, and the preservation of separator nodes. With certainty, for the multi-terminal

MOR application (contrary to others), the distribution of nodes must not necessarily be done equally among partitions. As regards the optimal values for the number of subnets N, the number of terminals per partition, and the number of separator nodes, these could be determined by a partitioning which uses a fill estimating measure as an objective function.

It is worth pausing here and think of an ideal partitioning scenario where the graph is partitioned into N disconnected parts with no separator nodes (clearly this is not possible as the graph consists of one strongly connected component). Even so, in this fictitious scenario, the distribution of terminals p across partitions would easily have a solution in distributing p equally over N: each part should receive $\frac{p}{N}$ terminals. In that case, the fill-in would be:

$$\sum_{i=1}^{N} \frac{\frac{p}{N}(\frac{p}{N}-1)}{2} = \frac{\frac{p^2}{N}-p}{2} < \frac{p(p-1)}{2}, \tag{7.8}$$

which becomes smaller with a larger N. Hence the larger the N the smaller the fill.

In the real scenario however, finding an optimal terminal distribution is less clear, due to the presence of separator nodes. A partition which supposedly balances the p terminals across the N parts may not necessarily yield as separator nodes those which best encompass the communication among subnetworks. For instance, imposing that each part gets $\frac{p}{N}$ terminals may result in a high number of separator nodes, and consequently a large reduced model. Hence it cannot be stated with certainty that terminals should be distributed equally.

Finer partitions (large N) do indeed decrease the number of terminals p_i per partition, and improve sparsity through smaller fill contributions in the sum (7.7). Also, the finer the partition, the higher will the number of separator nodes be, and by preserving them sparsity is improved (recall that separator nodes are typically the nodes of highest degrees, which if otherwise eliminated cause fill-in). This however must again pay the dimensionality price: the dimension of the reduced model increases with the number of separator nodes (which are all preserved). For this reason, one cannot argue that partitions should be as fine as possible.

These considerations have led to the formulation of **Problem 3** which, possibly with the help of a fill-estimating function similar to the CAMD approximation of node degrees, gives a partitioning criterion for explicitly minimizing the fill in the multi-terminal reduced order models. Incorporating these objectives inside available partitioning software remains for future work, most suitable with contributions from scientists in the field of combinatorial and scientific computing.

Chapter 8

Conclusions and future research

In this thesis, methods were developed for reducing multi-terminal systems arising in circuit simulation. The main purpose of this research was to provide a framework for model reduction which, aside from aiming at accuracy and small dimensionality, addresses other challenging aspects which appear when the reduced model is to replace the original in the desired simulation environment. Each chapter provided its individual conclusions and identified the relevant directions for future work. Here, a summary is provided for the main challenges addressed, the solutions provided, and the recommendations for further research which result from this work.

Chapter 1 described the application background, gave an overview of model order reduction, and motivated the need for new reduction methods for multi-terminal systems.

In Chapter 2 the multi-terminal problem was treated from a perspective which jointly addressed reduction and synthesis. The appropriate setup of system equations was derived, which ensures that reduced order models are later easily synthesized and reinserted in the simulation flow. The proposed setup is shown to especially suit reduction methods based on input/output structure preservation, a feature which ensures that synthesis does not generate unwanted circuit elements. Synthesis was separately addressed by comparing two approaches: Foster and unstamping. It was concluded that unstamping is the most suitable for multi-terminal models. The multi-terminal reduction and synthesis framework in this chapter gives the theoretical foundation for the approaches taken in the later chapters.

In Chapter 3, the attention was turned to the reduction of multi-terminal R/RC networks. The reduction of both R and RC networks is based on the same projection matrix, which has several important properties. A proof was given that the reduced R/RC models obtained from this projection have only positive resistors. It was also demonstrated that reduction can be implemented in a partitioned manner. A partitioned re-

duction of *RC* networks was proposed based on the strongly connected components of the conductance matrix. This partitioning gave a simple solution for computing the path resistances between network terminals. The most suitable circuits for reduction in this framework were those with a large number of internal nodes compared to terminals, and with terminal numbers within a few hundreds. For such circuits, numerical experiments showed reduction rates of usually 100% in the number of internal nodes, mostly above 80% in both the number of circuit elements, and up to 270 times faster simulations. As for future work, it is worth investigating whether/how the projection underlying R/RC reduction could be so constructed as to also guarantee positive capacitors.

Chapter 4 was dedicated to the reduction of more challenging *RC* networks with terminal numbers exceeding thousands. As preserving sparsity during reduction becomes critical for such circuits, more advanced reduction strategies than those of Chapter 3 were required. This led to the development of SparseRC, which employs graph partitioning and fill-reducing node reorderings to ensure that the reduced matrices retain sparsity. It was shown that, by splitting the network into minimally connected components and by reducing these individually, DC moments of the transfer function are matched per subnet and also globally, for the recombined network. This is an important result which ensures the accuracy of the approach irrespective of the chosen partitioning criterion or the number of subnets. SparseRC was tested on several industrial examples and demonstrated superior computational performance, higher reduction rates and improved sparsity compared to traditional approaches. It also successfully reduced circuits which could not be simulated in their original dimension, or those which could not even be approached with a traditional reduction method. Among the circuits which may still be challenging for SparseRC are those which contain significantly more capacitors than resistors. For such circuits, the reduced capacitance matrix can be still too dense. One remedy would be to further sparsify this matrix by deleting some capacitors from the reduced netlist, however with a robust error control which accounts for frequency dependency. An approach which is frequency independent can be found in the work of [91, Chapter 5] for sparsifying resistor networks. Another topic of further research would be the reduction of netlists with a small number of internal nodes relative to terminals. In that case, more advanced partitioners would help, a topic which is investigated in Chapter 7.

In Chapter 5 the reduction of multi-terminal *RLC* networks was addressed, with a focus on partitioning. Two reduction scenarios were considered in parallel, based on first-order and second-order dynamical systems respectively. Using the second-order form, a rank-revealing decomposition of the reduced susceptance matrix was proposed which ensures that the reduced network has no undesirable dependencies and that it successfully simulates. The second-order formulation was also the most natural to use in a partitioned manner, and the mathematical framework to achieve this was derived. Compared to *RC* reduction, in which the size of the reduced model was entirely determined by the number of terminals and separator nodes, for *RLC* systems additional blocks need to be preserved for the sake of accuracy. Thus, although the blocks corresponding to terminals and separator nodes can be made sparse as a result of partition-

ing, the additional internal blocks remain dense. Further research should address ways to achieve a better trade-off between small dimensionality, sparsity and accuracy, and the extension to *RLCk* reduction.

The methods in Chapters 3, 4, and 5 have two governing properties due to the special way in which the reducing projection is constructed: (1) they are able to reduce ungrounded systems, and (2) they preserve the structure of the input/output incidence matrices, thus ensuring that synthesis by unstamping is free of controlled sources or transformers. Many standard reduction methods however cannot handle ungrounded systems due to the fact that the underlying matrix pair $(\mathbf{A}, \mathbf{E})$ is singular, and also do no preserve input/output structure. Chapter 6 brought an additional contribution in this sense. It showed that, with an appropriate setup of system equations before reduction, followed by a post-processing step after reduction, even more basic reduction methods can reduce ungrounded systems and be synthesized without controlled sources or transformers. The contributions of Chapter 6 thus provide a new avenue for handling multi-terminal systems with other methods than those of Chapters 3, 4, and 5. Through experiments, it was shown how methods which are not input-output structure preserving, such as the standard PRIMA [71] or the new Loewner method [60], successfully apply in this new framework. What the framework does not yet ensure is sparsity. Hence, networks with terminal numbers within a few hundreds are the most suitable candidates for the approach, the extension to systems with a larger number of ports remaining for further research.

As this thesis shows, retaining sparsity is key to the success of model reduction for systems with many terminals. Chapter 4 in particular showed that through graph partitioning and component-wise reduction, sparsity is improved as: (1) terminals are spread across components and (2) fill-creating nodes are identified and not eliminated during the reduction process. Motivated by this achievement, Chapter 7 addressed separately the partitioning problem in the context of multi-terminal model reduction. Despite the significant sparsity gains achieved already with standard partitioning algorithms such as nested dissection, the analysis of Chapter 7 showed that sparsity could be further improved with the help of more specialized partitioners. Hence, the fill minimizing objectives relevant for the multi-terminal model reduction problem were derived. Their implementation into existing partitioning software is a motivating task for future work, especially with expert knowledge in graph theory.

Bibliography

[1] P.R. Amestoy, T.A. Davis, and I.S. Duff. An approximate minimum degree ordering algorithm. *SIAM Journal on Matrix Analysis and Applications*, 17(4):886–905, 1996.

[2] P.R. Amestoy, T.A. Davis, and I.S. Duff. Algorithm 837: AMD, an approximate minimum degree ordering algorithm. *ACM Trans. Math. Softw.*, 30(3):381–388, 2004.

[3] B.D.O Anderson and S. Vongpanitlerd. *Network Analysis and Synthesis*. Prentice-Hall, 1973.

[4] A. C. Antoulas. *Approximation of Large-Scale Dynamical Systems*. SIAM, Philadelphia, PA, 2005.

[5] A. C. Antoulas. A new result on passivity preserving model reduction. *Systems and Control Letters*, 54:361–374, 2005.

[6] A. C. Antoulas, C.A. Beattie, and S. Gugercin. Interpolatory mode reduction for large-scale dynamical systems. In J. Mohammadpour and K.M. Grigoriadis, editors, *Efficient modeling and control of large-scale systems*. Springer Verlag, 2010.

[7] W.E. Arnoldi. The principle of minimized iteration in the solution of the matrix eigenvalue problem. *Quart. Appl. Math*, 9(1):17–29, 1951.

[8] Z. Bai, R. Li, and Y. Su. A unified Krylov projection framework for structure-preserving model reduction. In W. Schilders, H. van der Vorst, and J. Rommes, editors, *Model order reduction*, volume 11 of *Mathematics in Industry*. Springer, Berlin, Germany, 2008.

[9] Z. Bai and Y. Su. SOAR: A second-order Arnoldi method for the solution of the quadratic eigenvalue prob. *SIAM Journal on Matrix Analysis and Applications*, 26(3):640–659, 2005.

[10] C. Beattie and S. Gugercin. Interpolatory projection methods for structure-preserving model reduction. *Systems & Control Letters*, 58(3):225 – 232, 2009.

[11] P. Benner. Advances in balancing-related model reduction for circuit simulation. In Janne Roos and Luis R.J. Costa, editors, *Scientific Computing in Electrical Engineering SCEE 2008*, volume 14 of *Mathematics in Industry*, pages 469–482. Springer, 2010.

[12] P. Benner, V. Mehrmann, and D. Sorensen, editors. *Dimension Reduction of Large-Scale Systems*, volume 45 of *Lecture Notes in Computational Science and Engineering*. Springer, 2005.

[13] P. Benner and A. Schneider. On stability, passivity and reciprocity preservation of ESVDMOR. In P. Benner, M. Hinze, and J. ter Maten, editors, *Model Reduction for Circuit Simulation*, Lecture Notes in Electrical Engineering, Berlin/Heidelberg, 2011. Springer-Verlag. Proceedings of the 2008 SyreNe/O-Moore-Nice Workshop on Model Reduction for Circuit Simulation, Hamburg, Germnay.

[14] R. Bott and R.J. Duffin. Impedance synthesis without the use of transformers. *Journal of Applied Physics*, 20:816, 1949.

[15] O. Brune. Synthesis of a finite two-terminal network whose driving point impedance is a prescribed function of frequency. *Journal of Mathematics and Physics*, 10:191–236, 1931.

[16] Cadence. Spectre. Version 7.1.1.071. Online: `http://www.cadence.com`.

[17] M.K. Camlibel and R. Frasca. Extension of KalmanYakubovichPopov lemma to descriptor systems. *Systems & Control Letters*, 58:795–803, 2009.

[18] Y. Chen, T. A. Davis, W. W. Hager, and S. Rajamanickam. Algorithm 887: CHOLMOD, supernodal sparse cholesky factorization and update/downdate. *ACM Trans. Math. Softw.*, 35(3):1–14, 2008. Software online: `http://www.cise.ufl.edu/research/sparse/cholmod/`.

[19] T.H. Cormen, C.E. Leiserson, and R.L. Rivest. *Introduction to Algorithms*. MIT Press, Cambridge, MA, 1 edition, 1990. Ch. 22.

[20] D. Deschrijver. *Broadband macromodeling of linear systems by vector fitting*. Phd. dissertation, Univ. Antwerpen, Antwerpen, Belgium, Oct. 2007.

[21] P. Dirac. *The principles of quantum mechanics*. Oxford at the Clarendon Press, 4th edition, 1958.

[22] P.J.H. Elias and N.P. van der Meijs. Extracting circuit models for large RC interconnections that are accurate up to a predefined signal frequency. In *Design Automation Conf. Proc. 1996, 33rd*, pages 764 –769, June 1996.

[23] E. Estrada and D. J. Higham. Network properties revealed through matrix functions. *SIAM Review*, 52(4):696–714, 2010.

[24] P. Feldmann and F. Liu. Sparse and efficient reduced order modeling of linear subcircuits with large number of terminals. In *CAD, 2004. ICCAD-2004. IEEE/ACM Int. Conf. on*, pages 88 – 92, 2004.

[25] M. Fiedler. *Special Matrices and Their Applications in Numerical Mathematics*. Martinus Nijhoff, 1986.

[26] R. W. Freund. Model reduction methods based on Krylov subspaces. *Acta Numerica*, 12:267–319, 2003.

[27] R.W. Freund. Sprim: structure-preserving reduced-order interconnect macromodeling. *Computer Aided Design, 2004. ICCAD-2004. IEEE/ACM International Conference on*, pages 80 – 87, nov. 2004.

[28] A. Ghosh, S. Boyd, and A. Saberi. Minimizing effective resistance of a graph. *SIAM Rev.*, 50:37–66, February 2008.

[29] G.H. Golub and C.F. Van Loan. *Matrix Computations*. Johns Hopkins University Press, Baltimore, Maryland, third edition, 1996.

[30] E. J. Grimme. *Krylov projection methods for model reduction*. PhD thesis, University of Illinois, 1997.

[31] E. J. Grimme, D. C. Sorensen, and P. Van Dooren. Model reduction of state space systems via an Implicitly Restarted Lanczos method. *Numer. Algorithms*, 12:1–31, 1996.

[32] S. Gugercin, A.C. Antoulas, and C.A. Beattie. A rational Krylov iteration for optimal $\mathcal{H}_2$ model reduction. In *Proceedings of the 17th Int. Symposium on Mathematical Theory of Networks and Systems,*, pages 1665–1667, MTNS06, Kyoto, 2006.

[33] A. Guille, M. Hanssen, and J. Niehof. Comparison between RLCk extraction and EM simulation on RF circuits. Technical report, NXP Semiconductors, August 2008.

[34] E. A. Guillemin. *Synthesis of passive networks*. John Wiley, 2 edition, 1959.

[35] B. Gustavsen and A. Semlyen. Rational approximation of frequency domain responses by vector fitting. *Power Delivery, IEEE Transactions on*, 14(3):1052 –1061, jul 1999.

[36] P. J. Heres. *Robust and efficient Krylov subspace methods for model order reduction*. PhD thesis, Eindhoven University of Technology, 2005.

[37] Chung-Wen Ho, A. Ruehli, and P. Brennan. The modified nodal approach to network analysis. *Circuits and Systems, IEEE Transactions on*, 22(6):504–509, 1975.

[38] S. Ingram. *Minimum Degree Reordering Algorithms: A Tutorial*. University of British Columbia, 2011 (accessed March 19, 2011). Online: `http://www.cs.ubc.ca/~sfingram/cs517_final.pdf`.

[39] R. Ionutiu, S. Lefteriu, and A. C. Antoulas. Comparison of model reduction methods with applications to circuit simulation. In G. Ciuprina and D. Ioan, editors, *Scientific Computing in Electrical Engineering SCEE 2006*, volume 11 of *Mathematics in Industry*, pages 3–24. Springer, 2007.

[40] R. Ionutiu and J. Rommes. Circuit synthesis of reduced order models. Thechnical Note 2008/00316, NXP Semiconductors, 2009.

[41] R. Ionutiu and J. Rommes. A framework for synthesis of reduced order models. CASA-report 09-28, Technische Universiteit Eindhoven, 2009.

[42] R. Ionutiu and J. Rommes. Model order reduction for multi-terminal circuits. CASA-report 09-29, Technische Universiteit Eindhoven, 2009.

[43] R. Ionutiu and J. Rommes. On synthesis of reduced order models. In P. Benner, M. Hinze, and J. ter Maten, editors, *Model Reduction for Circuit Simulation*, Lecture Notes in Electrical Engineering, Berlin/Heidelberg, 2011. Springer-Verlag. Proceedings of the 2008 SyreNe/O-Moore-Nice Workshop on Model Reduction for Circuit Simulation, Hamburg, Germnay.

[44] R. Ionutiu, J. Rommes, and A.C. Antoulas. Passivity-preserving model reduction using dominant spectral-zero interpolation. *CAD of Integrated Circuits and Systems, IEEE Trans. on*, 27(12):2250 –2263, 2008.

[45] R. Ionutiu, J. Rommes, and A.C. Antoulas. A new approach to passivity preserving model reduction: The dominant spectral zero method. In Hans-Georg Bock, Frank Hoog, Avner Friedman, Arvind Gupta, Helmut Neunzert, William R. Pulleyblank, Torgeir Rusten, Fadil Santosa, Anna-Karin Tornberg, Vincenzo Capasso, Robert Mattheij, Helmut Neunzert, Otmar Scherzer, Janne Roos, and Luis R.J. Costa, editors, *Scientific Computing in Electrical Engineering SCEE 2008*, volume 14 of *Mathematics in Industry*, pages 491–498. Springer Berlin Heidelberg, 2010.

[46] R. Ionutiu, J. Rommes, and W. H. A. Schilders. Model reduction for multi-terminal RC circuits. In *Scientific Computing in Electrical Engineering SCEE 2010*, Mathematics in Industry, 2010. To appear.

[47] R. Ionutiu, J. Rommes, and W. H. A. Schilders. Graph partitioning with separation of terminals. In M. Beckers, Johannes Lotz., V. Mosenkis, and U. Naumann, editors, *Fifth SIAM Workshop on Combinatorial Scientific Computing*, volume Technical Report/Aachener Informatik-Bericht (AIB), pages 39–41, Darmstadt, Germany, May 2011. Department of Computer Science, RWTH Aachen University. Book of Abstracts.

[48] R. Ionutiu, J. Rommes, and W. H. A. Schilders. SparseRC: sparsity preserving model reduction for RC circuits with many terminals. *CAD of Integrated Circuits and Systems, IEEE Trans. on*, 2011. Accepted (to appear).

[49] R. Ionutiu, J. Rommes, and W. H. A. Schilders. SparseRC: sparsity preserving model reduction for RC circuits with many terminals. CASA Report 11-05, Technische Universiteit Eindhoven, 2011.

[50] T. Kailath. *Linear Systems*. Prentice-Hall, 1980.

[51] G. Karypis and V. Kumar. *hMETIS, A Hypergraph Partitioning Package*. University of Minnesota, Department of Computer Science / Army HPC Research Center, Minneapolis, MN, version 1.5.3 edition, November 1998. `http://glaros.dtc.umn.edu/gkhome/metis/hmetis/download`.

[52] G. Karypis and V. Kumar. *METIS, A Software Package for Partitioning Unstructured Graphs, Partitioning Meshes, and Computing Fill-Reducing Orderings of Sparse Matrices*. University of Minnesota, Department of Computer Science / Army

HPC Research Center, Minneapolis, MN, version 4.0 edition, September 1998. `http://glaros.dtc.umn.edu/gkhome/metis/metis/download`.

[53] G. Karypis and V. Kumar. Multilevel k-way partitioning scheme for irregular graphs. *Journal of Parallel and Distributed Computing*, 48:96–129, 1998.

[54] G. Karypis and V. Kumar. A fast and highly quality multilevel scheme for partitioning irregular graphs. *SIAM Journal on Scientific Computing*, 20(1):359392, 1999.

[55] K.J. Kerns. *Accurate and Stable Reduction of RLC Networks Using Split Congruence Transformations*. Ph.d. dissertation, University of Washington, Seattle, WA, September 1996.

[56] K.J. Kerns and A.T. Yang. Stable and efficient reduction of large, multiport RC networks by pole analysis via congruence transformations. *CAD of Integrated Circuits and Systems, IEEE Trans. on*, 16(7):734 –744, jul 1997.

[57] K.J. Kerns and A.T. Yang. Preservation of passivity during RLC network reduction via split congruence transformations. *CAD of Integrated Circuits and Systems, IEEE Trans. on*, 17(7):582 –591, jul 1998.

[58] C. Lanczos. An iteration method for the solution of the eigenvalue problem of linear differential and integral operators. *J. Res. Natl. Bur. Stand.*, 45(4):225–280, 1950.

[59] Yu-Min Lee, Yahong Cao, Tsung-Hao Chen, J.M. Wang, and C.C.-P. Chen. HiPRIME: hierarchical and passivity preserved interconnect macromodeling engine for RLKC power delivery. *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, 24(6):797 – 806, 2005.

[60] S. Lefteriu and A. C. Antoulas. A new approach to modeling multiport systems from frequency-domain data. *CAD of Integrated Circuits and Systems, IEEE Trans. on*, 29(1):14 –27, 2010.

[61] S. Lefteriu and A.C. Antoulas. Modeling multi-port systems from frequency response data via tangential interpolation. In *IEEE Workshop on Signal Propagation on Interconnects*, pages 1–4, May 2009.

[62] S. Lefteriu and A.C. Antoulas. A new adaptive approach to modeling measured multi-port scattering parameters. In Janne Roos and Luis R.J. Costa, editors, *Scientific Computing in Electrical Engineering SCEE 2008*, volume 14 of *Mathematics in Industry*, pages 21–28. Springer, 2010.

[63] S. Lefteriu and A.C. Antoulas. Topics in model order reduction with applications to circuit simulation. In Hinze M. Benner, P. and E. J. W. ter Maten, editors, *Model Reduction for Circuit Simulation*, volume 13 of *Lecture Notes in Electrical Engineering*. Springer, 2011.

[64] S. Lefteriu, A. Ionita, and A. Antoulas. Modeling systems based on noisy frequency and time domain measurements. In Jan Willems, Shinji Hara, Yoshito Ohta, and

Hisaya Fujioka, editors, *Perspectives in Mathematical System Theory, Control, and Signal Processing*, volume 398 of *Lecture Notes in Control and Information Sciences*, pages 365–378. Springer, 2010.

[65] A.C.S. Lima, B. Gustavsen, and A.B. Fernandes. Inaccuracies in network realization of rational models due to finite precision of RLC branches. In *Proc. Int. Conf. Power Syst. Transients*, June 4–7 2007.

[66] B. Liu, X. Zeng, Y. Su, J. Tao, Z. Bai, C. Chiang, and D. Zhou. Block SAPOR: block second-order arnoldi method for passive order reduction of multi-input multi-output RCS interconnect circuits. In *Proceedings of the 2005 Asia and South Pacific Design Automation Conference*, ASP-DAC '05, pages 244–249, New York, NY, USA, 2005. ACM.

[67] Inc. Mathworks. Matlab. Version R2007a. Online: `http://www.mathworks.com/`.

[68] V. Mehrmann and T. Stykel. *Dimension Reduction of Large-Scale Systems*, volume 45 of *Lecture Notes in Computational Science and Engineering*, chapter Balanced Truncation Model Reduction for Large-Scale Systems in Descriptor Form, pages 83–115. Springer-Verlag, Berlin/Heidelberg, 2005.

[69] P. Miettinen, M. Honkala, J. Roos, and M. Valtonen. Improving model-order reduction methods by singularity exclusion. In B.L. Michielsen and J.-R. Poirier, editors, *The 8th international conference of Scientific Computing in Electrical Engineering*, 2010. Book of abstracts.

[70] P. Miettinen, M. Honkala, J. Roos, and M. Valtonen. PartMOR: Partitioning-based realizable model-order reduction method for RLC circuits. *CAD of Integrated Circuits and Systems, IEEE Trans. on*, 30(3):374–387, 2011.

[71] A. Odabasioglu, M. Celik, and L.T. Pileggi. PRIMA: passive reduced-order interconnect macromodeling algorithm. *CAD of Integrated Circuits and Systems, IEEE Trans. on*, 17(8):645–654, 1998.

[72] A.V. Oppenheim, A.S. Willsky, and S.H. Nawab. *Signals and Systems*. Signal Processing. Prentice-Hall, 1997.

[73] T. Palenius and J. Roos. Comparison of reduced-order interconnect macromodels for time-domain simulation. *Microwave Theory and Techniques, IEEE Trans. on*, 52(9):191–236, 2004.

[74] J.R. Phillips, L. Daniel, and L.M. Silveira. Guaranteed passive balancing transformations for model order reduction. *CAD of Integrated Circuits and Systems, IEEE Trans. on*, 22(8):1027–1041, 2003.

[75] J.R. Phillips and L.M. Silveira. Poor man's TBR: a simple model reduction scheme. *CAD of Integrated Circuits and Systems, IEEE Trans. on*, 24(1):43 – 55, 2005.

[76] T. Reis. Circuit synthesis of passive descriptor systems - a modified nodal approach. *Intl. Journal of Circuit Theory and Appl.*, 38(1):44–68, 2010.

[77] T. Reis and T. Stykel. PABTEC: Passivity-preserving balanced truncation for electrical circuits. *CAD of Integrated Circuits and Systems, IEEE Trans. on*, 29(9):1354 –1367, sep. 2010.

[78] J. Rommes. *Methods for eigenvalue problems with applications in model order reduction*. PhD thesis, Utrecht University, 2007.

[79] J. Rommes and N. Martins. Efficient computation of multivariable transfer function dominant poles using subspace acceleration. *Power Systems, IEEE Trans. on*, 21(4):1471 –1483, nov. 2006.

[80] J. Rommes and W. H. A. Schilders. Efficient methods for large resistor networks. *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, 29(1):28 –39, jan. 2010.

[81] A. Ruhe. Rational Krylov, a practical algorithm for large sparse nonsymmetric matrix pencils. *SIAM J. Sci. Comput*, 19:1535–1551, 1998.

[82] W. H. A. Schilders, H. A. van der Vorst, and J. Rommes. *Model order reduction: theory, research aspects and applications*, volume 13 of *Mathematics in Industry*. Springer, 2008.

[83] L. Schwartz. *Théorie des distributions*. Hermann, 1997.

[84] B.N. Sheehan. Realizable reduction of RC networks. *CAD of Integrated Circuits and Systems, IEEE Trans. on*, 26(8):1393 –1407, 2007.

[85] Roger Sidje. On the simultaneous tridiagonalization of two symmetric matrices. *Numerische Mathematik*, 118:549–566, 2011. 10.1007/s00211-010-0357-9.

[86] D.C. Sorensen. Passivity preserving model reduction via interpolation of spectral zeros. *Systems and Control Letters*, 54:347–360, 2005.

[87] R. A. Stein. Comments on "Synthesis of RC multiport impedance functions". IEEE Trans. on Circuit Theory, January 1972.

[88] A. Steinbrecher and T. Stykel. Model order reduction of nonlinear circuit equations. Preprint 2011/02, Technische Universität Berlin, 2011.

[89] Y. Su, J. Wang, X. Zeng, Z. Bai, C. Chiang, and D. Zhou. SAPOR: second-order Arnoldi method for passive order reduction of RCS circuits. In *Computer Aided Design, 2004. ICCAD-2004. IEEE/ACM International Conference on*, pages 74 – 79, 2004.

[90] S. X. D. Tan and L. He. *Advanced model order reduction techniques in VLSI design*. Cambridge University Press, 2007.

[91] M. Ugryumova. *Applications of Model Order Reduction for IC Modeling*. PhD thesis, Eindhoven University of Technology, 2011.

[92] EECS Dept. Univ. of California at Berkeley. Spice. Online: `http://bwrc.eecs.berkeley.edu/classes/icbook/spice/`.

[93] F. Yang, X. Zeng, Y. Su, and D. Zhou. RLC equivalent circuit synthesis method for structure-preserved reduced-order model of interconnect in VLSI. *Commun. in comput. physics*, 3(2):376–396, 2008.

[94] Zuochang Ye, D. Vasilyev, Zhenhai Zhu, and J.R. Phillips. Sparse implicit projection (SIP) for reduction of general many-terminal networks. In *ICCAD 2008. IEEE/ACM International Conference on*, pages 736 –743, 2008.

[95] Hao Yu, Lei He, and S.X.D. Tan. Block structure preserving model order reduction. In *Behavioral Modeling and Simulation Workshop, 2005. BMAS 2005. Proc. of the 2005 IEEE International*, pages 1 – 6, 2005.

[96] A. N. Yzelman and R. H Bisseling. Cache-oblivious sparse matrix-vector multiplication by using sparse matrix partitioning methods. *SIAM Journal on Scientifc Computing*, 31:3128–3154, 2009. Software online: `http://www.staff.science.uu.nl/~bisse101/Mondriaan`.

[97] A.I. Zecevic and D.D. Siljak. Balanced decompositions of sparse systems for multilevel parallel processing. *Circuits and Systems I: Fundamental Theory and Appl., IEEE Trans. on*, 41(3):220 –233, 1994.

Summary

Ever since its beginnings in the 1950's, the integrated circuit (IC) has profoundly changed our lives. The way we work, travel, communicate, or address medical problems today has been facilitated by advances in microelectronics, which permit more functionality to be built on the same silicon area, at decreasing cost. As the feature size of devices on a chip shrink and circuits operate at increasing frequencies, the electromagnetic coupling effects between different IC components can no longer be ignored. To understand their impact on chip performance, these so called parasitic effects must be simulated. Parasitic networks are often so large, that state of the art simulation tools are insufficient to handle them: the simulations are either too lengthy, or cannot be carried out at all. The mathematical reason behind this is that the underlying systems are too large to be solved with the numerical algorithms implemented in simulation software. *Model order reduction (MOR)* provides one avenue for enabling faster simulations at little accuracy loss. However, when the systems have many input/output nodes, i.e., they have *many terminals*, performing the reduction itself becomes even more challenging. In this thesis model reduction methods are developed for multi-terminal systems arising in industrial problems. To solve these effeciently, the methods rely jointly on concepts from numerical linear algebra, electrical engineering, and computer science.

This thesis begins with an overview of MOR in Chapter 1 and places the *reduction of multi-terminal systems* in the context of existing approaches. Aside from being efficient and accurate, multi-terminal MOR methods should also ensure that the reduced model is easily inserted in the simulation environment in place of the original, and that it is indeed cheaper to simulate. Although all reduction methods are expected to satisfy these properties, this thesis shows that such expectations are rarely met when traditional approaches are applied to very large electrical networks with many terminals which arise in industrial problems. Hence, an improved framework for multi-terminal model reduction and synthesis is proposed. The methods developed in this thesis address three global problems: (1) the efficient and accurate reduction of multi-terminal circuits, (2) the appropriate synthesis of the reduced model into a netlist equivalent with the same terminal nodes, and (3) the re-simulation of the reduced circuit (instead of the original) with emphasis on accuracy and simulation time.

In Chapter 2, a basic framework is developed for the reduction of multi-terminal net-

works and the synthesis of reduced multi-terminal models. Chapter 2 shows that, if the circuit equations are prepared appropriately, a multi-terminal RLC network can be reduced so that the synthesis step is also greatly simplified. In particular, from the reduced mathematical construction, an equivalent circuit containing only RLC elements can be obtained, without introducing unintended circuit elements such as controlled sources. In addition, the reduced circuit has the same terminal nodes as the original and is coupled easily to other circuit blocks in the simulation setup. The framework establishes the mathematical principle which allows voltage sources, non-linear devices or other parts of a larger network to be de-coupled from to specific linear part to be reduced. It also ensures that these elements can be re-coupled in the simulation phase to the reduced circuit via its terminals.

In Chapters 3 and 4, new methods for reducing large, multi-terminal R, and RC networks are derived, with emphasis on accuracy, efficiency and sparsity. It is shown that, if the projection which reduces an R/RC network performs a Schur-complement operation on the original conductance matrix, the resulting reduced network will have only positive resistors, which may be important for certain circuit simulators. This projection is also shown to exactly preserve the path resistance between terminals, and for RC circuits, the slope of the response in addition (in system theoretic terms, two multi-port admittance moments at DC are matched). The efficiency and sparsity considerations are dealt with especially in Chapter 4. These become critical for circuits with terminal numbers exceeding thousands, and node numbers exceeding hundreds of thousands. Reducing them by traditional means is either inefficient, or results in dense reduced models which are more expensive to simulate than the originals. Chapter 4 however develops a new method which is able to reduce efficiently such challenging RC netlists, while ensuring that the reduced models are sparse and fast to simulate. The key principles of the approach are graph partitioning, fill-reducing node re-orderings, and a reducing projection which is constructed accordingly. This preserves sparsity and also maintains accuracy by moment matching irrespective of how the circuit is partitioned. Based on the result of Chapter 2, the reduced models thus obtained are synthesized without controlled sources, have the same terminal nodes as the original ones, and are therefore inserted easily in the desired simulation flow.

With the decrease of transistor feature sizes and increase of operating frequencies, it becomes important to also investigate the effects of parasitic inductances (e.g., skin or proximity effects) on chip performance. This requires time-consuming simulations of very large multi-terminal RLC(K) networks, and thus motivates the need for appropriate reduction methods. Chapter 5 identifies the main challenges of multi-terminal RLC reduction, and presents a skeleton for approaching them based on partitioning principles similar to those of Chapter 4. Important problems pertaining to RLC reduction are also identified in Chapter 5, which are otherwise rarely explicitly addressed in the literature. These include matching the response at DC when the underlying conductance matrix is singular, or the presence of singularities in the reduced conductance matrix which in turn negatively affect the simulation of the reduced model. For the latter problem especially, a solution is proposed in Chapter 5 .

Two constraints which are known to limit the applicability of traditional reduction methods to multi-terminal systems are as follows: (a) the underlying matrix pencils are often singular and (b) the reducing projections may destroy the structure of the input/output matrices and with that, the physical interpretation of terminal nodes. The advanced methods of Chapters 3, 4 and 5 automatically by-pass these limitations due to the special way in which the reducing projection is formed. Chapter 6 brings an additional contribution by showing how, despite the known limitations, more general reduction methods (e.g., the Loewner approach) are also able to handle multi-terminal systems. The new reduction-synthesis framework of Chapter 6 eliminates the pencil singularity using a simple pre-processing of the original circuit, and recovers the connectivity at all terminal nodes using a post-processing of the reduced model.

Among the main results of this thesis is the improvement in reduced model sparsity and reduction efficiency, achieved with the help of graph partitioning. State-of-the-art partitioning algorithms such as nested dissection or Mondriaan already served this purpose, nevertheless Chapter 7 shows that the results could be further strengthened with the help of partitioning criteria designed especially for multi-terminal MOR. A high level description of the desired objectives is also derived there.

Samenvatting

Model Orde Reductie voor Multiterminal Systemen met Toepassingen binnen Circuit Simulatie

Vanaf het begin der 50'er jaren van de vorige eeuw hebben gïntegreerde schakelingen onze levens behoorlijk veranderd. Onze manier van werken, reizen en communiceren en de wijze waarop medische problemen worden aangepakt zijn allemaal beïnvloed door vorderingen binnen de micro-elektronica, welke voortdurend tot meer functionaliteit leiden op eenzelfde stukje silicium tegen een dalende kostprijs. Vanwege de voortdurende miniaturisering en oplopende frequenties kan de elektromagnetische koppeling tussen verschillende componenten in een schakeling niet langer worden genegeerd. Teneinde hun effect op het gedrag van een schakeling te kunnen begrijpen, dienen deze zogenaamde parasitaire effecten te worden gesimuleerd. Parasitaire netwerken zijn echter vaak zodanig groot dat zij onhandelbaar zijn voor de beschikbare simulatietechnieken: simulaties duren veel te lang of kunnen helemaal niet uitgevoerd worden. De wiskundige reden hierachter is dat de onderliggende stelsels te groot zijn om opgelost te kunnen worden met de numerieke algoritmen welke beschikbaar zijn in de simulatiesoftware. Model orde reductie (MOR) is een van de mogelijkheden om (snellere) simulaties uit te voeren bij slechts een gering verlies aan nauwkeurigheid. Echter, wanneer de systemen teveel invoer/uitvoer punten bevatten, zogenaamde multiterminal systemen, dan wordt ook de reductie een grote uitdaging. In dit proefschrift ontwikkelen we methoden voor model orde reductie voor multiterminal systemen welke voorkomen bij industriële problemen. Teneinde deze efficiënt aan te kunnen pakken zijn deze methoden gebaseerd op een combinatie van concepten uit de numerieke lineaire algebra, elektronica en informatica.

Dit proefschrift begint in hoofdstuk 1 met een overzicht van MOR, en plaatst de reductie van multiterminal systemen in de context van bestaande aanpakken. Naast efficiëntie en nauwkeurigheid dienen MOR methoden voor multiterminal systemen ook de eigenschap te bezitten dat de originele modellen eenvoudig te vervangen zijn door gereduceerde modellen binnen de gebruikte simulatieomgevingen, en dat deze ook daadwerkelijk goedkoper zijn om mee te simuleren. Hoewel men zou verwachten dat alle reductiemethoden deze eigenschappen hebben, laten we in dit proefschrift zien dat deze

verwachtingen zelden worden bewaarheid als gangbare aanpakken worden toegepast op zeer grote elektrische netwerken met vele terminals zoals voorkomend bij industriële problemen. Dientengevolge stellen we een verbeterd raamwerk voor wanneer het gaat om model orde reductie voor multiterminal systemen en het hieraan gerelateerde probleem van de synthese. De in dit proefschrift ontwikkelde methoden zijn gericht op globaal de volgende drie problemen: (1) het efficiënt en nauwkeurig reduceren van schakelingen met vele terminals; (2) een adequate synthese van gereduceerde modellen in een netlijst welke dezelfde terminals bevat; en (3) de hernieuwde simulatie van de gereduceerde schakeling (in plaats van de originele) met de nadruk op nauwkeurigheid en simulatietijd.

In hoofdstuk 2 wordt een basisraamwerk ontwikkeld voor de reductie van multiterminal netwerken en de synthese van gereduceerde modellen met vele terminals. Dit hoofdstuk laat zien dat, wanneer de netwerkvergelijkingen op passende wijze worden voorbereid, een RLC netwerk met vele terminals zodanig kan worden gereduceerd dat de er op volgende synthesestap drastisch wordt vereenvoudigd. In het bijzonder kan middels de mathematische constructie een equivalent netwerk afgeleid worden dat enkel RLC elementen bevat, zonder dat er ongewenste elementen zoals gecontroleerde bronnen worden geïntroduceerd. Daarnaast heeft het gereduceerde netwerk dezelfde invoer- en uitvoerpunten als het originele netwerk, zodat het eenvoudig verbonden kan worden met andere blokken in het betreffende ontwerp. Het raamwerk huldigt het wiskundige principe dat het mogelijk maakt om spanningsbronnen, niet-lineaire componenten en andere onderdelen van een grotere schakeling los te koppelen van het specifieke lineaire deel welk gereduceerd dient te worden. Tevens zorgt het ervoor dat deze elementen in de simulatiefase weer gekoppeld kunnen worden aan het gereduceerde netwerk via de terminals.

In hoofdstukken 3 en 4 worden nieuwe methoden afgeleid voor het reduceren van grote R en RC netwerken met vele terminals, waarbij de nadruk ligt op nauwkeurigheid, efficiëntie en ijlheid. Er wordt aangetoond dat, als de projectie voor de reductie van een R of RC netwerk een Schur-decompositie uitvoert op de originele geleidbaarheidsmatrix, het resulterende netwerk enkel positieve weerstanden zal bevatten; dit kan belangrijk zijn voor bepaalde netwerksimulatoren. Daarnaast tonen we aan dat deze projectie ook een exacte weergave geeft van de padweerstanden tussen terminals, terwijl deze voor RC systemen leidt tot het matchen van de helling van de respons (in termen van systeemtheorie betekent dit dat twee momenten van de admittantie worden gematched in de DC conditie). Hoofdstuk 4 houdt zich vooral bezig met de aspecten van efficiëntie en ijlheid. Deze aspecten zijn cruciaal voor netwerken met vele duizenden terminals en honderdduizenden knopen. Reductie van zulke netwerken met traditionele methoden is ofwel zeer inefficiënt, of resulteert in zeer dichtbezette gereduceerde modellen welke duurder zijn qua simulatie dan de oorspronkelijke modellen. De in hoofdstuk 4 ontwikkelde nieuwe methode is in staat om zulke grote multiterminal RC netwerken efficiënt te reduceren en er tevens voor zorg te dragen dat de gereduceerde modellen ijl zijn alsmede zeer snel simuleerbaar. Kern van de aanpak zijn partitionering van de bij het netwerk behorende graaf, alsmede vullingreducerende ordeningen en bijbehorende projecties voor de reductie. Combinatie van deze aspecten zorgt, middels matching

van momenten, voor behoud van de ijlheid en de nauwkeurigheid, onafhankelijk van hoe het netwerk is gepartitioneerd. Gebruikmakend van de resultaten van hoofdstuk 2 kunnen de op deze wijze verkregen gereduceerde netwerken gesynthetiseerd worden zonder gecontroleerde bronnen, en met dezelfde hoeveelheid terminals als het originele netwerk. Daardoor zijn deze ook eenvoudig op te nemen in de gewenste simulatieomgeving.

Met het afnemen van de afmetingen van transistoren en het toenemen van de gebruikte frequenties wordt het ook belangrijk om parasitaire inductieve effecten (zoals het randeffect of het nabijheidseffect) op het gedrag van schakelingen te onderzoeken. Dit vergt tijdrovende simulaties van zeer grote multiterminal RLC(K) netwerken, en is daarmee een motivatie voor adequate reductiemethoden voor dit soort systemen. Hoofdstuk 5 identificeert de belangrijkste uitdagingen in verband met de reductie van multiterminal RLC systemen, en presenteert een raamwerk voor het aanpakken van dit type problemen gebaseerd op partitioneringsprincipes verwant aan de methoden gepresenteerd in hoofdstuk 4. Er worden belangrijke vraagstellingen geïdentificeerd met betrekking tot RLC netwerkreductie welke vaak niet worden vermeld in de literatuur. Onder deze is het matchen van de respons bij DC condities, wanneer de achterliggende matrix singulier is; ook worden genoemd singulariteiten in de gereduceerde geleidingsmatrix welke een negatieve invloed hebben op het simuleren van de gereduceerde modellen. Voor dat laatste probleem wordt in hoofdstuk 5 een oplossing voorgesteld.

Twee aspecten welke de toepasbaarheid van gangbare reductiemethoden op multiterminal netwerken beperken zijn de volgende: (a) de achterliggende matrices zijn vaak singulier, en (b) de projecties voor de reductie kunnen de structuur van de invoer- en uitvoermatrices in negatieve zin beïnvloeden. Dit laatste leidt dan tot problemen met de fysische interpretatie van de knooppunten geassocieerd met de terminals. De geavanceerde methoden in hoofdstukken 3, 4 en 5 vermijden automatisch deze beperkingen vanwege de speciale wijze waarop de projecties voor de reductie worden geconstrueerd. Hoofdstuk 6 levert hier een additionele bijdrage aan door aan te tonen dat, ondanks de bekende beperkingen, algemenere reductiemethoden (zoals de Loewner aanpak) ook in staat zijn om te gaan met multiterminal systemen. Het nieuwe raamwerk voor reductie en synthese gepresenteerd in hoofdstuk 6 elimineert de singulariteit van de matrix pencil door gebruik te maken van een simpele preprocessing van het originele netwerk, terwijl de connectiviteit tussen de terminals wordt hersteld middels een postprocessing van het gereduceerde model.

Onder de belangrijkste resultaten in dit proefschrift is de substantiële verbetering in de ijlheid van het gereduceerde model en de efficiëntie van dit model, welks bereikt wordt door gebruik te maken van partitionering van de onderliggende graaf. Geavanceerde algoritmen voor partitionering, zoals de zogenaamde nested dissection of het Mondriaan algoritme, voldeden hier reeds aan. Hoofdstuk 7 laat zien dat de resultaten verder verbeterd kunnen worden met criteria voor partitionering geformuleerd speciaal voor model orde reductie van multiterminal netwerken. Ook leiden we hier een hoog niveau beschrijving van de gewenste doelen af.

Acknowledgements

I would like to begin by thanking my promotors, Prof. Wil Schilders at TU Eindhoven and Prof. Athanasios Antoulas at Jacobs University, for giving me the opportunity to carry out this research under the framework of a joint doctorate program[1]. I wish to thank Prof. Schilders and Prof. Antoulas for inspiring me with the clarity of their thinking and vision, for the freedom they allowed in organizing my work, at the same time helping me stay on track and focus on the most interesting and relevant topics. I thank them for also encouraging and supporting numerous conference participations, which have been both professionally enriching, and enjoyable by meeting interesting people, and seeing interesting places.

My special thanks and sincere gratitude go to Dr. Joost Rommes, my co-promotor and industry supervisor at NXP Semiconductors. Without Joost's relentless and dedicated support, this work would not have been possible. Thank you, Joost, for answering all my questions, for the patience and attention in understanding all my concerns, for challenging me in every line of thought, and for carefully reviewing papers, presentations and this thesis. I am also very grateful for all your help and advice in many non-work related situations.

For their willingness and time dedicated to evaluate this thesis, I would like to thank all committee members: Prof. Wil Schilders, Prof. Athanasios Antoulas, Prof. Rob Bisseling, Prof. Bob Mattheij, Prof. Michael Günther, Prof. Arjeh Cohen, Dr. Joost Rommes, and Dr. Jan ter Maten together with Dr. Mathias Bode from Jacobs University.

I am also grateful to a number of people from NXP Semiconductors: Giuseppe Grillo and Jos Gobbels, for providing the resources to work on realistic problems, Gerben de Jong, Dennis Jeurissen, Marcel Hanssen, Hans van Walderveen, and Frank Leong for providing the parasitic netlists, assessing reduction results and for giving useful insights on the circuit operations. Thanks also to Theo Beelen for revising part of this thesis, and for his enjoyable company at NXP.

[1]The joint doctorate was financially supported by the COMSON European project, Contract number MRTN-CT 2005-019417, and by the Deutshcen Forschungsgemeinschaft (DFG) research program "AN-1 Modellreduktion".

For special attention to my graph-partitioning related concerns, I greatly appreciate the contribution of Albert-Jan Yzelman and Prof. Rob Bisseling from Utrecht University. Useful insights which helped in understanding fill-reducing ordering methods were also given by Prof. Tim Davis from the University of Florida. For a fruitful knowledge exchange and willingest to perform experiments on request, I wish to thank the Part-MOR [70] authors Pekka Miettinen, Mikko Honkala and Janne Roos, and the Loewner method [60] author Sanda Lefteriu.

Special thanks for his support in various aspects of my work (both technical and financial) are also due to Dr. Jan ter Maten, my supervisor in the COMSON project. As part of the COMSON team, my thanks go to Zoran Ilievski, Alexander Rusakov and Georg Denk for their help in implementation aspects related to the COMSON demonstrator platform.

Coordinating work between two countries may not have always been a smooth process, but thanks to the contribution of several people, all hurdles were easily removed. In this respect, I wish to thank again all my supervisors and Prof. Bob Mattheij of CASA, for the encouragement, trust, and understanding given both on professional, as well as personal level. For their help on several organization matters, I thank Enna van Dijk and Marése Wolfs-van de Hurk at TU/E, and Peter Tsvetkov, Svenja Frischholz, Kaija Grünefeld, Doris Fedler and Ronald Kieschnick from Jacobs University.

Last but not least, I am heartily thankful to my office mates and colleagues in CASA, TU/E and NXP, friends from Eindhoven, Bremen, and further abroad, for the nice moments shared together, for their help in difficult times, and support in various aspects of my life. For unconditional love and support, I thank my devoted family from all my heart.

Roxana Ionuţiu

Eindhoven, August 2011.

Curriculum Vitae

Roxana Ionuţiu was born on July 29, 1984, in Oradea, Romania. After finishing high-school at the "Emanuil Gojdu National College", Oradea, Romania, in 2003, she became a student of Electrical Engineering and Computer Science at Jacobs University (International University Bremen at that time), in Bremen, Germany. She obtained her Bachelor of Science degree from Jacobs in 2006. In 2005 she studied one semester abroad at Rice University, Houston, Texas, USA. She also held two summer internship positions, one at Bremen Online Services GmbH, Bremen, Germany, working on web programming (2004) and a research internship on model reduction at Rice University, under the supervision of Prof.Dr. Athanasios C. Antoulas (2005).

She continued her master's in Electrical Engineering at Rice University and obtained her MSc degree in 2008, with thesis: "Passivity preserving model reduction in the context of spectral zero interpolation", under the supervision of Prof.Dr. Athanasios C. Antoulas. This work was based in part based on the implementation of a specialized eigenvalue solver and so initiated a collaboration with Dr. Joost Rommes at NXP Semiconductors in Eindhoven, the Netherlands.

During her BSc and MSc studies she was a teaching assistant for several courses in electrical engineering and linear dynamical systems (2004-2008) and also worked briefly as a student assistant in the college administration at Jacobs University (2003-2004), and in the multimedia library at Rice University (2005).

From 2008 to 2011, she was a PhD candidate with a joint appointment between Jacobs University, Bremen, Germany (with Prof.Dr. Athanasios C. Antoulas), Eindhoven University of Technology, Eindhoven, the Netherlands (with Prof.Dr. W.H.A. Schilders), and NXP Semiconductors, Eindhoven, the Netherlands (with Dr. Joost Rommes). As a guest at NXP Semiconductors, she implemented the numerical methods presented in this thesis and verified their performance on industrial problems.