POLITECNICO DI TORINO

SCUOLA DI DOTTORATO

Dottorato in Ingegneria Elettronica e delle Comunicazioni - XXV ciclo

Tesi di Dottorato

Macromodels for simulation-based verification of System on Chip and System in Package

Candidato

Salvatore Bernardo Olivadese

Tutore

Prof. Stefano Grivet Talocia

Marzo 2014

to A. M.;
you are like this work...

Abstract

In recent years the focus on electronic integration shifted from high performance microprocessors, whose integration trend is dictated by the famous Moore law, to System on Chip (SoC) and System in Package (SiP) for mobile and embedded applications. The most common example of SoC can be found in smartphones and tablets: multicore CPU (Central Processing Unit) and GPU (Graphics Processing Unit), memory and Radio Frequency (RF) transceivers are often integrated in the same die or package leading to tremendous reduction in size and power consumption of the device. Therefore SoCs/SiPs are by definition heterogeneous electrical systems, in the sense that analog and digital components for RF and Base Band (BB) applications are closely tied together.

To blend such a variety of components in the same electronic package engineers face new difficulties both in design and verification phases. Signal and Power integrity need to be carefully addressed in conjunction with noise levels to address devices constraints. In the context of Analog Mixed Signal (AMS) validation, analog blocks are still the simulation time bottlenecks. The main issues are: the huge complexity of the parasitic networks extracted from components layouts and interconnects, the need of parametric models for non-linear components for what-if analyses, the need of reduced order models for devices having huge ports count like Power Delivery Networks (PDNs) and packages and the lack of low complexity noise complaint synthesis methods for linear macromodels. Although tremendous steps forward were achieved in the last decades in the areas of system identification and model order reduction there are still chances for improvement.

In this thesis the state of the art from system identification of Linear Time Invariant (LTI) systems is revised and improved tailoring the needs of AMS simulations for SoC/SiP applications: a new system identification algorithm to cope with linear components having huge dynamical order and ports count (more than two order of magnitudes) is proposed and passivity constraints are verified and imposed by means of parallel algorithms. The identification of parametric linear models is extended to parameterized small-signal models for non-linear devices. Finally a low-complexity noise compliant synthesis algorithm is introduced in order to export the macromodels in standard SPICE-based solvers. The main contributions of this work are: reduction of simulation time for the verification of modern SoCs/SiPs, introduction of parameterized small-signal models for non-linear RF components enabling a simplified assessment of different project scenarios supporting the widespread

Intellectual Property (IP) reuse pattern, optimization and simplification of the verification flow based on the provision of multi-purpose IP blocks in the form of noise compliant networks.

We are facing the rise of a new era for consumer electronic, and time-to-market is a key feature in the development of new products. Therefore the availability of effective Analog Mixed Signal methodologies becomes a sustainable competitive advantage for companies that are willing to lead these new market segments. The novel algorithms proposed in this work were proved to be of practical relevance in that sense.

Most part of the material presented in this work is based on a research activity carried out at the Munich site of Intel Mobile Communication. As a consequence the methodologies proposed here, arising from practical needs, were tested on several commercial benchmarks demonstrating the importance of melting research activities with industries requirements.

Acknowledgements

I would like to thank the staff of Intel Mobile Communication in Munich for the hospitality and for the tools offered during my PhD. A special thank goes to my Intel supervisor Pietro Brenner and to my manager Alexander Ruehl for making my PhD at Intel possible; to Gianni Signorini (Intel), Michelangelo Bandinu (IdemWorks), Alessandro Chinea (IdemWorks) and Piero Triverio (University of Toronto) for the support and the suggestions during the development of my work.

Last but not least, the most special thank is reserved to my PhD supervisor, Professor Stefano Grivet-Talocia, for the patience and the invaluable guidance during these years.

Contents

Chapter 1

System on Chip for mobile applications

Businesses fail either because they leave their customers or because their customers leave them![1]
Andrew S. Grove, Intel corporation senior advisor

1.1 History and market perspectives

System on Chip (SoC) defines a highly integrated design pattern for Integrated Circuits (IC). Sundry levels of integration are grouped by the SoC definition: starting from a simple chip to memory interconnection up to the integration of a complete transceiver[1] chain for cellphones applications. The SoC paradigm raised naturally in the last decade to meet the requirements of a new fast-growing market segment, i.e. the so called mobile market.

Only a few years ago Personal Computer (PC) users were always demanding for an increase of the computational power. Central Processing Unit (CPU) evolution was well predicted by the famous Moore's law [2] and the outcome nowadays are very complex devices delivering huge computing capabilities. The first step towards mobility was the introduction of Laptops. Thereupon new design constraints appeared: power consumption and form factor.

Telecommunication systems profited from the electronic evolution as well: internet and the world wide web increased in usage and popularity, cellphones evolved delivering a wide range of applications exploiting the potentiality of a fast growing network infrastructure. The standards for mobile communica-

[1] Transceiver: device comprising both a transmitter and a receiver which are combined and share common circuitry or a single housing.

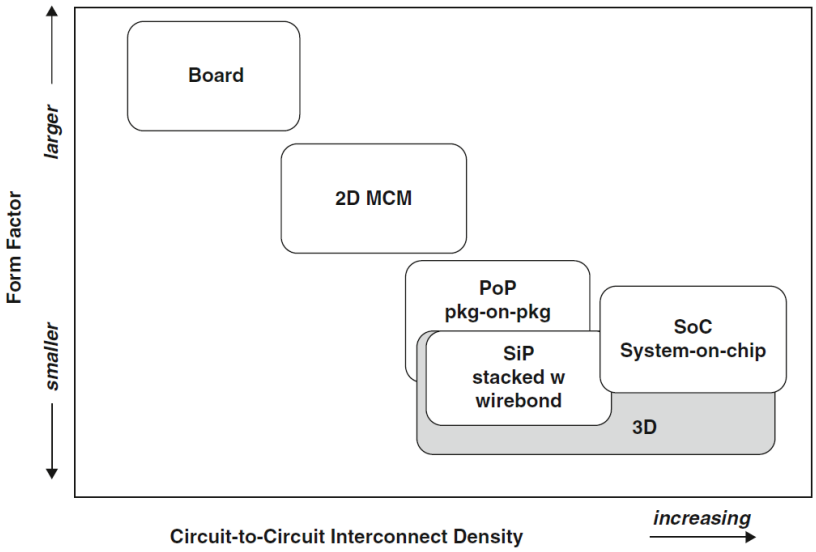

Figure 1.1: The most common system integration technologies are grouped in the figure above as a function of form factor and circuit-to-circuit interconnect density [3, 4].

tion from the third generation (3G) on pushed toward an optimized usage of the communication channel in order to allow the transmission of considerable amounts of data.

In order to combine laptop features with cellphones portability, SiP (System in Package) and SoC are nowadays the integration paradigm for smartphones, tablets and phablets. A nice overview of the most common system integration[2] technologies as a function of form factor and circuit-to-circuit density [3, 4] is depicted in Figure 1.1. Planar integration technologies are becoming more challenging as transistor channel lengths hit the range 20-30 nm. In order to meet the requirements of the market, 3D stacking techniques are emerging as a promising workaround to planar integration limitations [5].

1.2 Design challenges

Compared with the design of nowadays classic ICs, Radio Frequency (RF) SoC design is more involved due to the melt of heterogeneous electronic systems in a small package [6]. Moreover, for RF and mobile applications, Analog Mixed Signal (AMS) methodologies are a must since Digital Signal

[2]System integration is defined as the combination of circuits and Intellectual Property (IP) blocks on the same die.

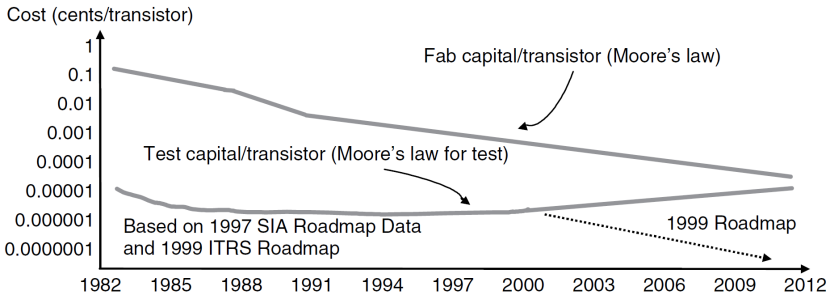

Figure 1.2: Fabrication capital versus test capital based on Semiconductor Industry Association (SIA) and International Technology Roadmap for Semiconductors (ITRS), source [12].

Processing (DSP) blocks are in close connection with analog and RF components [7] thus increasing the overall design complexity.

The main issues arising in RF SoC applications can be divided in two macro groups.

1. Die and package: At this level the growth in transistor count and operating frequency has a direct impact on design complexity leading to

   - poor manufacturability: as the miniaturization process gets closer to the theoretical limits of CMOS (Complementary Metal-Oxide-Semiconductor) technology [8] the design becomes very sensitive to process variation. This affects the throughput yield[3], reliability and testability. In 1999 the Semiconductor Industry Association (SIA) proposed an International Technology Roadmap for Semiconductors (ITRS) showing how the cost of test is going to surpass the cost of silicon manufacturing as depicted in Figure 1.2. As a consequence there is an increasing interest in automatic testing methodologies [10] and adaptive design techniques [11] to stem the drawbacks related with process tolerances;
   - power consumption: four are the main sources of power dissipation in CMOS technology [13]. P_{dyn}: dynamic switching power due to the charging and discharging of circuit capacitances. P_{leak}: due to the leakage current from the reverse-biased diodes and subthreshold conduction. P_{short}: due to the finite signal rise/fall times. P_{bias}: static biasing power. Those issues are addressed by supply power scaling techniques and Low Power (LP) CMOS technologies [14];

[3] The definition of Yield in the context of Integrated Circuits is: the ratio of the number of functional chips to that of the total chips manufactured[9]

- power delivery issues: low-power consumption constraints transformed the design of Power Delivery Networks (PDNs) into a very challenging task in comparison with previous IC technologies [15, 16]. Multi-layer packages and grids are common to supply clean power to the integrated circuits. Two are the figures of merit for PDNs: the target impedance[4] [18] and the voltage IR drop. Both account for two different phenomena: the static IR voltage drop[5] which is introduced by the resistive nature of the PDN conductors, and the inductive di/dt voltage drop which derives from localized power demand and switching patterns [19]. Moreover, large voltage drops in on-chip PDN due to large di/dt may lead to Electro-migration[6] (EM) that is one of the most critical interconnect failure mechanism in ICs [17]. Besides Power Integrity (PI) considerations, PDN should be also designed to afford dynamic power management methodologies meant for power saving modes driven by the control firmware [21];
- heat dissipation: the typical range of operating junction temperature for modern VLSI designs is between 80° and 120° on the silicon substrate [22]. Such boundaries are easily exceeded due to the cumulative power dissipation of the transistors leading to the generation of extreme amounts of heat in a relatively small area. High thermal density has a negative impact on circuit performances by increasing the gate delay and shortening the life of the device. Therefore the packages are carefully designed to remove the heat from the IC substrate;
- on-chip crosstalk: this is mainly introduced by the inter-wire coupling capacitance between adjacent signal lines in on-chip buses [23]. Both hardware (shielding via grounded conductors or particular layout fabrics [24]) and coding signal techniques (crosstalk avoidance codes, CACs [25]) are available to cope with this problem;
- noise: the effect of thermal/white noise due to the increase of temperature becomes always more relevant and needs to be carefully addressed. The flicker (1/f) noise is tightly related with the

[4]The target impedance is calculated from: power supply tolerance, current and switching activity and has to be satisfied by the PDN from DC to at least the first harmonic of the clock frequency [17].

[5]Static IR Voltage drop: is the reduction of the nominal reference voltage for transistors due to the transition of current (I) in the resistive (R) power delivery network.

[6]Electro-migration: flow of metal atoms under the influence of high current densities. May be the cause for increased resistance and reliability problems [20].

CMOS technology and becomes relevant only below a specific corner frequency [13].

2. System and components: considering that portable devices are meant to support different communication standards like: Bluetooth (IEEE 802.15.1), Wi-Fi (IEEE 802.11), GSM, GPRS, UMTS and many more, it is sensible that the same transceiver has to be used for all the communications standards to meet the form-factor constraints of a portable device. As a consequence transceivers and communications systems become more complex due to the advent of new standards and the need to preserve retro-compatibility leading to

   - interconnect delay: for off-chip buses the main bottleneck is represented by the package. Data rate limits are related with the quality of the package. Because of that the performance of the package are crucial for the assessment of Signal and Power Integrity (SI,PI) analysis;
   - off-chip crosstalk: this is mainly due to inter-symbol interference (ISI) and inductive crosstalk [26]. Eye diagram analysis [27] is usually adopted to study such kind of problems.

Exploiting Silicon On Insulator (SOI) technology [28] the future of IC integration goes in the direction of 3D stacking [29]. Integration density, power consumption and form-factor can be effectively addressed by 3D SoC design methodologies[30] while Through Silicon Via (TSV) and Network on Chip (NoC) are the emerging interconnect paradigms [31].

All the design challenges and methodologies described in this section are faced relying on advanced modelling techniques and a well established design flow. Next sections will outline the state of the art on macromodeling and design flow for RF SoC.

1.3 Macromodeling and Design flow

Computer Aided Design (CAD) techniques are well established and widespread in the electronic industries since decades. The introduction of Electronic Design Automation (EDA) dates back to 1980 when it became clear that the gap advances in engineering productivity (P_1) compared with the increase in silicon complexity (P_2) was widening, as depicted in Figure 1.3. This trend, know as *productivity gap* [32], became more relevant due to the advent of SoC designs and stringent time-to-market constraints. The classic design flow, depicted in a simplified version in Figure 1.4, is no longer effective in

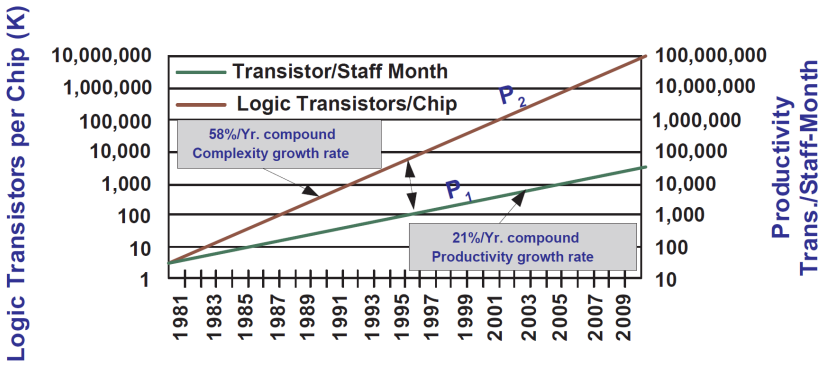

Figure 1.3: The bordeaux line represents the increase for the number of transistors per chip as a function of years (P_2) while the green line indicates the advances in engineering (P_1), source [32].

coping with the *productivity gap* in the context of SoC for mobile applications. The following requirements should be met by an effective SoC design flow:

- rapid development to satisfy time-to-market pressures;
- quality of results: performance, form-factor and power consumption;
- simple verification of complex chips;
- simple to use for teams with different levels and areas of expertise.

To satisfy the constraints listed above modern design flows are heavily relying on the concept of IP (Intellectual Property) reuse [33]: each step in the design flow depicted in Figure 1.4 is now enhanced and supported by well established IP blocks. In a similar fashion to the *code reuse* pattern widely used in Information Technology applications, the main idea behind the IP reuse strategy relies upon the construction of a library of components (generally called IP blocks or macros) to be used in several different projects. More details on this topic are provided in the next section.

Together with IP reuse, as ICs and design flows become more involved, macromodels and related tools must improve and accomplish new features. A macromodel is a high-level mathematical description of the system under analysis that accurately represents its behaviour. The prefix *macro* emphasizes that only the input/output response is described, while no information is retained on the internal structure of the physical system. Besides the typical requirements of accuracy, numerical robustness, physical consistency and efficiency, a few new constraints must be taken into account for macromodels

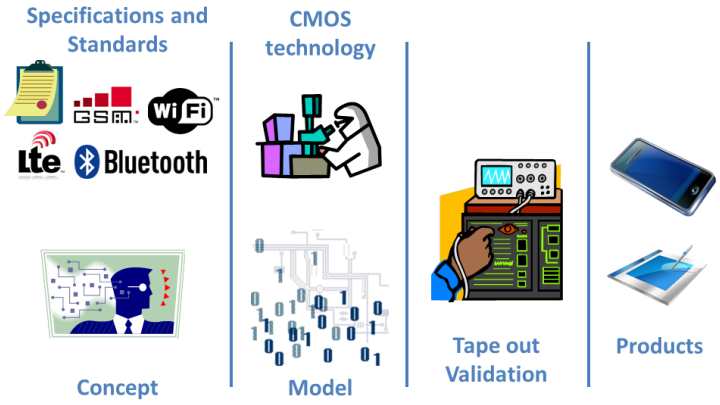

Figure 1.4: The mains steps involved in the design flow of mobile devices are sketched. Starting from specifications and standards the concept of the device is built. A model prototype is created using a peculiar technology in a CAD environment. Several EDA software are used in the pre-tape-out phase to address: functional specifications, manufacturability and physical consistency of the prototype model. In the tape-out phase functional specifications are checked on physical designs. In case of issues the model prototype is used as a test bench. Of course, to reduce production costs, the minimum number of tape-outs should be used to meet all the specifications.

- parameterization: in order to speed up what-if analysis and optimization procedures macromodels should admit some of the most common design parameters (temperature, V_{dd} and geometry) as input variables. With such a feature there is no need to build a new macromodel in case of variations of design parameters;

- usability: macromodels should be available in a standard format, like Spice netlist or HDL (Hardware Descrition Language). The same model must be effective for different type of analyses (time/frequency domain, noise). Inputs, parameters and options must enclose a simple and clear description together with applicability bounds. Thereby independently of user's expertise the model can be used effectively, in a short time and in several different contexts;

- scalability: it is well known that SoCs complexity, intended as dynamical order and elements/interconnections count, grows really fast with time. Modelling techniques must cope with this trend, providing accurate models with low complexity in a short time.

Design complexity and *productivity gap* will further increase with the advent of 3D integration technologies; therefore the availability of accurate models

providing low complexity is the corner stone for a modern design flow intended to meet tight time-to-market constraints. Macromodels associated to the sub-blocks of a complex system can be combined to mimic the behaviour of the whole system leading to a tremendous simplification in the analysis of complex devices. In the following sections the main features of IP reuse and how to deliver adequate macromodels for this new design paradigm is discussed.

1.3.1 IP reuse

Design techniques based on IP reuse were born in the beginning of the 1990's [33]. Two major events are considered as the starting points for the IP reuse diffusion:

- Establishment of the Virtual Socket Interface Alliance (VSIA): in 1996 this cross-industry organization, focused on IP reuse for SoC design, was founded to help foster this new design pattern by combining the skills and knowledge of semiconductor companies, system companies and EDA industry;
- Register Transfer Level (RTL) IP reuse: in 1997 teams from Mentor Graphics and Synopsis proposed the so called *Reuse Methodology* for *soft* IP. The dictates of this design pattern are collected in the widely known Reuse Methodology Manual [34].

The core idea behind IP-oriented SoC design relies upon the availability of reusable IP blocks that support plug-and-play integration in a pre-defined flow. As such IP blocks are the highest level building blocks of an SoC, they are collected in libraries with various timing, area and power configurations providing to designers simple to use IP macros.

The form of a reusable IP core can vary depending on the IP developer/vendor; as a high level classification, three are the following main categories of IP blocks [34]:

- soft IP: blocks defined using RTL or higher level descriptions. They are typically used for digital cores relying on a process-independent hardware description language (HDL) that can be synthesized to gate level. Advantages: flexibility, portability and reusability; while the drawbacks are: lack of timing and power characteristics because performances are tightly related with the technology used to synthesize the HDL. Those macros can be encrypted to hide IP details and prevent the introduction of unreliable features; as a drawback encrypted blocks can not be adapted to fit new design scenarios;

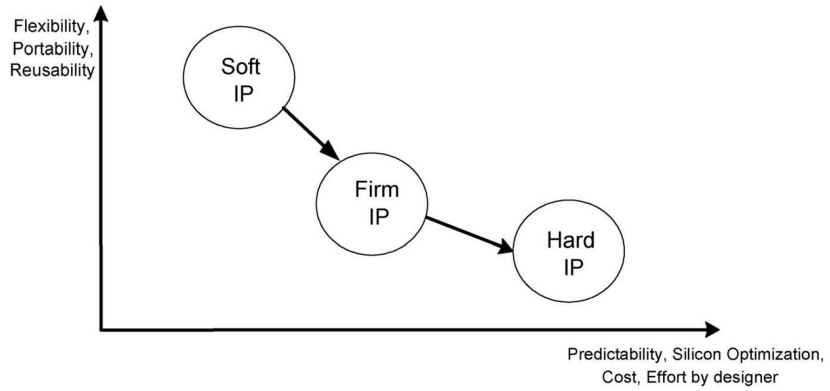

Figure 1.5: The selection of the most suitable form to deliver and IP block should take into account the trade-off depicted in this plot [35].

- hard IP: usually defined by means of faithful layouts tailored for a specific application based on a given technology. For those blocks, performances are predictable but the consequent drawback is the lack of portability;
- firm IP: in the middle between hard and soft blocks, firm IP is delivered as parametrized analog circuit meant to be tailored by designers for a specific application. Block's features can be trimmed leveraging on the available parameters while retaining predictable performances.

As a consequence, selecting the most suited IP form for each block is of paramount importance in order to build an effective and reliable design flow for SoC applications. To drive such an important decision, the plot depicted in Figure 1.5 is suggested as a reference map in [35].

When the IP reuse strategy is applied to the AMS design for RF SoC, one problem arises [36], i.e. the selection of the IP form most suited for an AMS block. Compared to digital design, for which a common design methodology is available [37], AMS design usually relies on specific design process. This issue can be addressed using an effective mixed-signal SoC flow [38, 39] based on the AMS IP blocks in [40, 41, 42].

Currently, due to the complexity of AMS designs, the soft and hard IP forms are used for analog-mixed signal applications [40, 41, 42]. Of course, this choice restricts the scope of applications reducing the overall SoC design flow efficiency [36]. The migration of hard AMS IP blocks to the firm form calls for new features on the macromodels used to derive netlists and schematics. Indeed, as stated in Section 1.3, parametrizability and scalability are the new features required on the macromodel side. Moreover, in

order to provide a high level of usability for such models (consequently for the firm AMS IP blocks) a clear and simple taxonomy is needed; next section introduces such a classification for macromodels.

1.3.2 Macromodels taxonomy

A simple and clear taxonomy for macromodels is needed in order to meet the usability constraints imposed by the IP reuse paradigm detailed in Section 1.3.1. Considering that the main bottleneck in the design of analog-mixed-signal components is represented by the analog blocks, two will be the main criteria behind the proposed taxonomy: all the components are analog (indeed also digital blocks are synthesized via analog elements), and their level of non-linearity is the base for classification. As a consequence of this coarse classification the proposed taxonomy is "orthogonal" to fine technological details attaining the degree of portability required by IP blocks meant to the firm IP form.

In the following for each level of classification the state of the art on macromodeling and system identification will be briefly outlined together with a list of AMS components belonging to each level of the proposed taxonomy.

Linear Time Invariant (LTI) systems

There are several components that can be accurately modelled as Linear Time Invariant systems: packages [43], buses and interconnects [44], Printed Circuit Boards (PCB) [45], Power Distribution Networks (PDNs) [46] and Through Silicon Via (TSV) for 3D SoC [47]. The construction of LTI models for those components is usually based on the work flow depicted in Figure 1.6 from [43]: Scattering parameters are extracted from the layout or 3D model of the component under analysis using a full wave solver. Thus the LTI model can be extracted using the time or frequency raw data leading to a state-space [48] or descriptor representation. Several well assessed techniques are available to construct LTI models from tabulated data:

- Nevanlinna-Pick interpolation [49, 50] is a well known result of complex analysis. Two matrix versions exist for this problem: the matrix Nevanlinna-Pick problem [51] and the tangential Nevanlinna-Pick problem [52]. This method was adopted for the first time in the system identification context by [53] and more recent application can be found in [54]. A comprehensive description of the Nevanlinna-Pick problem, its extensions and variations can be found in [55]. Despite this method

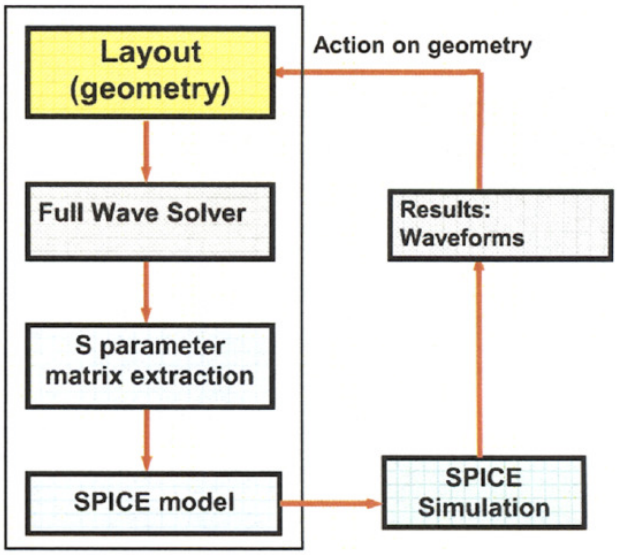

Figure 1.6: The typical work flow used for the creation of LTI models from packages, PCB, TSV and related components is presented. Starting from the layout or the 3D model a full wave solver is applied in order to extract the Scattering parameters. From S-parameters the LTI model is identified via the standard techniques summarized in Section 1.3.2. Once the LTI model is available it can be synthesized as a Spice network and the results from Spice are validated with the results from the 3D full wave solver, source [43].

is theoretically attractive it is seldom used in practice due to computational complexity and numerical stability reasons;

- Löwner interpolation dates back to the work of Löwner for the interpolation of given data on a full arc of the unit circle in the complex plane [56]. It was introduced in the context of control theory and system identification by Kalman and Belevitch [57]. More recent applications of this method can be found in [58, 59];

- The Sanathanan-Koerner iteration was originally proposed in [60] and it is based on the complex curve fitting proposed by Levy in [61]. This is a general strategy to recast a non-linear interpolation problem to the solution of a sequence of linear overdetermined systems. The most popular evolution of the Sanathanan-Koerner iteration is the Vector Fitting (VF) algorithm [62, 63]. Nowadays this is the de facto standard for the identification of linear systems in the EDA context. Despite VF has no guarantees of convergence when dealing with noisy data [64], it offers the best trade-off between computational complexity and robustness [65]. As a consequence the Sanathanan-Koerner iteration and VF are used in this work and are presented in more details in Section 2.2;

- Padé approximation, originally proposed by the mathematician Henri Padé [66], addresses the best approximation of a function under a specific norm by a rational function of a given order. It was introduced in control theory to model exponential delays [67]. Recent applications can be found in system identification literature [68]. This method was quite popular before the introduction of VF and can be still considered a good alternative to the Sanathanan-Koerner iteration for low-order systems [69, 70];

- subspace methods [71] are all composed by three steps: estimation of the predictable subspace from raw data, extraction of the state variable from the predictable subspace and fitting the estimated states to a state space model. Several algorithms are available both for continuous [72] and discrete [73] time models identification. Those techniques are numerically stable and efficient [74]. The lack of a priori physical properties impositions, like stability and passivity, prevents the systematic application of those methods on analog circuits.

It is worth noting that the techniques listed above are meant for electronic devices whose behaviour can be effectively approximated via lumped element

networks, i.e. the propagation delay of the signal can be neglected, otherwise different techniques should be used, like [75, 76].

Parameterized LTI (P-LTI) systems

Although LTI models are helpful and their usage is widespread, the main drawback of the LTI approach lies in the lack of flexibility. Indeed several components like: PCBs, interconnects, packages, RF inductors and TSVs are designed and tested considering different geometrical configurations and working temperatures. As a consequence, a considerable effort was spent in the last years to extend the identification algorithms introduced in Section 1.3.2 to obtain Parameterized-LTI models:

- parameterized Nevanlinna-Pick interpolation was first proposed in [77] but found only few applications in robust control applications [78];
- parameterized Löwner interpolation was introduced by [79]. Due to the major memory consumption this method is not used in practice;
- parameterized Sanathanan-Koerner (SK) iteration was first proposed by Triverio in [80] and then extended by the same author to account for stability [81] and passivity [82]. In a similar fashion VF was used by Ferranti for the P-LTI identification [83] and then with passivity constraints [84]. Currently those are the most diffused techniques for the identification of P-LTI models. Some applications and advances are presented in Section 3.2;
- parameterized Padé approximation can be found in [85]. Being a competitive alternative to VF and SK iteration it found several applications [86, 87];
- parameterized subspace methods were addressed recently [88]. Those methods suffer from a curse of dimensionality leading to an ill-posed parameter estimation problem; a recent attempt to overcome such a limitation can be found in [89].

Despite the theoretical effort, up to now none of the techniques listed above for the identification of P-LTI systems has the robustness and the efficiency to become part of a user-friendly EDA tool.

Small-Signal P-LTI

Using a proper Direct Current (DC) correction strategy [90], presented in Section 3.1, P-LTI models can be also applied to mimic the behaviour of

non-linear devices that behave almost linearly in the neighbourhood of one equilibrium point[7]. This is a common scenario in RF applications, indeed components like: Low Dropout (LDO) regulators, Operational Amplifiers (Op Amp), Low Noise Amplifiers (LNA), buffers and active filters are designed to behave almost linearly under specific working conditions. In the context of RF applications, linear behaviour means that the device does not generate spurious harmonics or that the spurious harmonics are strongly attenuated and thus negligible. For AMS high integration technologies, like in SoC and SiP, the suppression of spurious harmonics is relevant to control coupling noise and undesired mixing effects.

Piecewise linear P-LTI

The P-LTI method can be extended to model strong non-linear devices like drivers, mixers and Phase-Locked-Loops (PLLs) using a piecewise linear interconnection of P-LTI models. The first work dealing with piece wise linear (PWL) networks dates back to Stern in 1956 [92]. A more rigorous study on PWL models for non-linear devices is due to Chua [93], while several PWL techniques are compared in [94]. The idea to use state-space models with PWL states is quite recent and found several applications for the modelling of non-linear devices [87]. In the context of AMS circuits PWL techniques can be found in: formal verification of analog circuits [95], behavioural modelling of nonlinear power amplifiers [96] and mixed-signal circuits [97].

1.4 Proposed advances

Despite the research effort spent in the development of EDA tools and algorithms, design and verification of AMS SoC is still an open issue, which costs to mobile communications companies huge resources [32]. Therefore the main objective of this doctoral work consists in the development of new methodologies to cope with the challenges posed by SoC integration highlighted in Section 1.2. The proposed solutions, while advancing the state of the art for macromodeling of electronic devices, arise from industrial constraints and real design test cases, providing an immediate contribution to practical needs.

Chapter 2 deals with the identification of linear macromodels belonging to the LTI taxonomy class presented in Section 1.3.2. State-space models

[7] In this work by equilibrium or fixed point the Lyapunov definition of local stability is considered [91].

representation and basic identification tools are summarized in Section 2.1 and 2.2. The main contributions of Chapter 2 are:

- a compressed macromodeling algorithm is introduced in Section 2.3 to overcome the limitations of VF when dealing with components having huge ports count, from tens to hundreds. As discussed in Section 1.2, at system level the main bottleneck for interconnections is represented by the package, while at chip level 3D technologies like TSV and NoC are meant to increase the connectivity. The original version of VF [62] and also the more recent advances like [98, 99] are not suited to address such devices because of the excessive memory consumption or due to ill-conditioning. The proposed compressed macromodeling algorithm overcomes those issues relying on a clever reduction of the data set used for the identification of the model. Accuracy and physical properties like causality and passivity can be imposed directly on the compressed macromodel, as presented in Section 2.3.4, leading to a tremendous speedup on the overall identification procedure (see Section 2.4.4) in comparison with state of the art techniques [100];

- a parallel algorithm to verify the passivity of linear macromodels is introduced in Section 2.5. Since the most common algorithms for system identification (VF and SK) do not guarantee the generation of passive models, passivity needs to be addressed independently [101]. Moreover, passivity characterization is of course the first step for the passivity enforcement [102], and needs to be repeated several times. Several algorithms are available for the passivity characterization [101, 103, 104]. Some of them are already available for parallel architectures [105]. The algorithm proposed in Section 2.5 is an efficient parallel implementation of [104];

Chapter 3 discusses the identification of parameterized LTI (P-LTI) models. The availability of parameterized models is the cornerstone for the development of a modern and effective design and verification flow. Considering that several methodologies for the identification of P-LTI models are available as discussed in Section 1.3.2, the main contributions of Chapter 3 are

- a Direct Current (DC) correction strategy for small-signal models of non-linear circuits, presented in Section 3.1. This simple but effective idea is the link between linear and small-signal models for non-linear devices. RF devices like LDO and OpAmp are designed to behave almost linearly under appropriate biasing. The so called small-signal LTI models of those devices are accurate around a specified operating point

but fail to reproduce the DC response of the real non-linear device. The proposed DC correction can be used to overcome this issue;

- parameterized small-signal models are proposed in Section 3.2. According to the taxonomy proposed in Section 1.3.2 models are sorted depending on the level of non-linearity. The combination of P-LTI models with a parameterized DC correction strategy makes it possible to model fairly non-linear devices using a smooth combination of linear models parameterized by the operating point. The effectiveness of this strategy is demonstrated in Section 3.3 by analysing some test cases of practical relevance.

Chapter 4 presents the synthesis of State-Space models as linear lumped networks. As already noted in Section 1.3.1, the first step for the migration of AMS IP blocks towards the IP firm description relies on the availability of flexible and efficient implementations of the macromodels. Therefore canonical synthesis[8] algorithm in Spice compatible format are described. The main contributions of Chapter 4 are:

- modern presentation of canonical synthesis methods for static and dynamic networks discussed in Section 4.2 and 4.3. For each synthesis method: complexity of the network and practical relevance are detailed. In particular: stability, accuracy and noise analysis compliance are considered. Statical network synthesis techniques are considered in their own because of practical relevance for connectivity, static IR drop [106] and power distribution analysis;

- a new synthesis method for dynamic networks, based on Darlington resistance extraction framework, is presented in Section 4.3.4. Each step of this new algorithm is described focusing on numerical robustness and noise compliance of the resulting Spice netlist.

Finally, conclusions are summarized in the last Chapter, highlighting both theoretical and practical relevance of results and methodologies discussed in this work.

[8] As explained in Chapter 4, a network synthesis is defined as *canonical* when it requires the theoretical minimum number of primitive network elements.

Chapter 2

Linear Time Invariant macromodels

Macromodeling techniques have become a standard practice in system design and verification flows. Such methods allow to convert external characterizations of linear and time-invariant structures such as passive devices and electrical interconnects into compact closed-form mathematical expressions or circuit equivalents. This conversion is needed to allow system-level transient simulations and verifications starting from a native characterization that is typically available in the frequency domain in form of tabulated scattering responses, the latter being determined from direct measurements or full-wave numerical solutions.

This Chapter introduces some advances to the state of the art of Linear Time Invariants (LTI) macromodeling techniques. The necessary background on state-space models and system identification is discussed in Section 2.1, while two of the most popular algorithms for linear systems identification are described in Section 2.2, i.e. the Sanathanan-Koerner iteration and Vector Fitting. Extensions and improvements for those identification methods are the main contributions of this Chapter. In Section 2.3, the Compressed macromodeling algorithm is introduced as a clever system identification procedure based on Vector Fitting for systems having a large port count. In Section 2.5, a highly efficient parallel passivity verification method is presented.

2.1 State-space macromodels

The state-space representation was introduced in control engineering and circuit theory by Bashkow [48] and nowadays is the most common descrip-

tion for dynamical systems. State-space equations constitute a mathematical model of the physical system under analysis as a set of input, output and internal state variables related by coupled first-order differential equations. Dealing with linear time-invariant systems the associated state-space equations read

$$\dot{\boldsymbol{x}}(t) = \mathbf{A}\boldsymbol{x}(t) + \mathbf{B}\boldsymbol{u}(t), \tag{2.1}$$

$$\boldsymbol{y}(t) = \mathbf{C}\boldsymbol{x}(t) + \mathbf{D}\boldsymbol{u}(t). \tag{2.2}$$

with $\mathbf{A} \in \mathbb{R}^{N\times N}$, $\mathbf{B} \in \mathbb{R}^{N\times P}$, $\mathbf{C} \in \mathbb{R}^{P\times N}$ and $\mathbf{D} \in \mathbb{R}^{P\times P}$ constant matrices. Inputs are collected in vector $\boldsymbol{u}$, outputs in vector $\boldsymbol{y}$ while the internal states are in vector $\boldsymbol{x}$. Two features are of paramount importance for a state-space system

- observability: defined as the ability to always reconstruct the initial state $\boldsymbol{x}(0)$ observing the outputs of the system for $t \geq 0$ provided that also the input evolution is known for $t \geq 0$;
- controllability: defined as the possibility to always design an input sequence that steers the system to a desired final state.

Both conditions are guaranteed when the model (2.1)-(2.2) has minimal dynamic order, defined as the McMillan degree of the system [107]. If the state-space is not minimal, it can be converted to a minimal one by means of standard techniques [108].

Taking now the Laplace transform of (2.1) and (2.2) and assuming $x(0) = 0$, it follows

$$s\mathbf{X}(s) = \mathbf{A}\mathbf{X}(s) + \mathbf{B}\mathbf{U}(s), \tag{2.3}$$

$$\mathbf{Y}(s) = \mathbf{C}\mathbf{X}(s) + \mathbf{D}\mathbf{U}(s), \tag{2.4}$$

which leads to the transfer function matrix relating $\mathbf{U}(s)$ and $\mathbf{Y}(s)$

$$\mathbf{H}(s) = \mathbf{D} + \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B}. \tag{2.5}$$

The transfer function (2.5) is rational. In case of poles (eigenvalues of $\mathbf{A}$) with unit multiplicity, $\mathbf{H}(s)$ can also be written in the so called pole-residue form, i.e.

$$\mathbf{H}(s) = \mathbf{D} + \sum_{n=1}^{N} \frac{\mathbf{R}_n}{s - p_n}, \tag{2.6}$$

where p_n are the simple poles, $\mathbf{R}_n$ are the associated residue matrices, and $\mathbf{D}$ is the direct coupling term. Please note that the three descriptions (2.1)-(2.5)

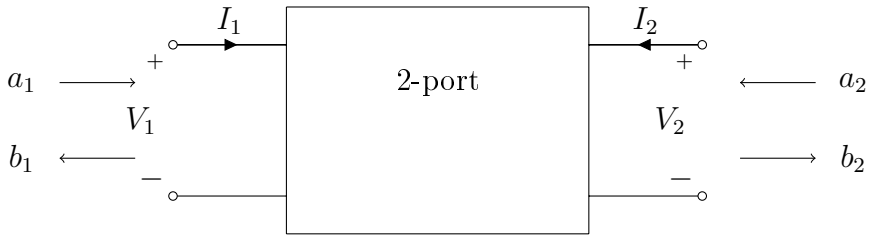

Figure 2.1: Incident a_i and reflected b_i power waves for a two-port network.

and (2.6) associated to the minimal state-space (2.1) system are equivalent to each other and one is preferred to the others depending on the application context.

The identification work flow described in Section 1.3 (Figure 1.4) heavily relies on the availability of accurate models in the form of (2.1). Such models can be converted to linear lumped networks to be solved using SPICE based solvers using the synthesis techniques discussed in Chapter 4. In order to extract accurate state-space models using the raw data obtained from measurement or full-wave solvers an identification algorithm must be used. In the following the raw data used for the identification are supposed to be Scattering (S)-parameters [109]. Recall that the S-parameters for a 2-port (the extension to P-port is straightforward) linear time-invariant network, as depicted in Figure 2.1, are defined as

$$\begin{bmatrix} b_1 \\ b_2 \end{bmatrix} = \begin{bmatrix} S_{11} & S_{12} \\ S_{21} & S_{22} \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \end{bmatrix} \rightarrow \boldsymbol{b} = \mathbf{S}\boldsymbol{a}\,, \tag{2.7}$$

where Z_0 is a prescribed real reference impedance and the travelling waves a_i and b_i are defined as

$$a_1 = \frac{V_1 + Z_0 I_1}{2\sqrt{Z_0}}\,, \quad a_2 = \frac{V_2 + Z_0 I_2}{2\sqrt{Z_0}}\,, \tag{2.8}$$

$$b_1 = \frac{V_1 - Z_0 I_1}{2\sqrt{Z_0}}\,, \quad b_2 = \frac{V_2 - Z_0 I_2}{2\sqrt{Z_0}}\,. \tag{2.9}$$

The main constraint common to all identification procedures consists in the minimization of the difference between the linear identified model response and the reference raw data-set. Working with S-parameters (2.7) the raw data for an LTI network is composed of matrices $\mathbf{S}_l = \mathbf{S}(s_l) \in \mathbb{R}^{P\times P}$, with $l = 1, \ldots, L$ number of discrete frequency samples $s_l = \jmath\omega_l$. In this context the identification problem can be formulated as: find a state-space model $\mathbf{S}(s)$ such that

$$\min \sum_l \|\mathbf{S}(s_l) - \mathbf{S}_l\|^2 \tag{2.10}$$

for a given norm.

Among all the identification algorithms, listed in Section 1.3.2 of Chapter 1, for the extraction of accurate state-space models starting from raw data the two most used in practice are the Sanathanan-Koerner iteration and the Vector Fitting procedure. Those two methods are presented in the following section.

2.2 Sanathanan-Koerner and Vector Fitting

The minimization constraint (2.10) associated to the identification problem was addressed by Sanathanan and Koerner in [60], the resulting Sanathanan-Koerner (SK) Iteration is briefly summarized in this section together with his most popular extension, i.e. the Vector Fitting (VF) algorithm [62].

The identification of a scalar transfer function $h(s)$ is considered, instead of the matrix case (2.10), in order to simplify and focus the presentation on the algorithm. The extension to multi-port devices is straightforward [60]. In the basic SK Iteration framework a set of frequency-domain samples (s_l, h_l) for $l = 1, \ldots, L$ is used to identify a rational model of the form

$$h(s; \boldsymbol{x}) = \frac{n(s; \boldsymbol{x})}{d(s; \boldsymbol{x})} = \frac{a_0 + a_1 s + \cdots + a_m s^m}{b_0 + b_1 s + \cdots + b_{n-1} s^{n-1} + s^n} \tag{2.11}$$

where $n(s; \boldsymbol{x})$ and $d(s; \boldsymbol{x})$ are respectively numerator and denominator polynomials of degree m and n. The unknown coefficients are collected in the vector

$$\boldsymbol{x} = (a_0, a_1, \ldots, a_m, b_0, b_1, \ldots, b_{n-1})^T. \tag{2.12}$$

The general identification problem requires to find coefficients $\boldsymbol{x}$ which minimize in some norm the residual error $\boldsymbol{r}(\boldsymbol{x})$, whose components are

$$r_l(\boldsymbol{x}) = h_l - \frac{n(s_l; \boldsymbol{x})}{d(s_l; \boldsymbol{x})}. \tag{2.13}$$

To avoid the solution of a non-linear interpolation problem the strategy proposed by Levy in [61] can be used, i.e. instead of minimizing the non-linear residual (2.13), the following modified residual is minimised

$$e_l(\boldsymbol{x}) = d(s_l; \boldsymbol{x}) r_l(x) = d(s_l; \boldsymbol{x}) h_l - n(s_l; \boldsymbol{x}) \tag{2.14}$$

by solving the linear least square problem

$$\mathbf{F}\boldsymbol{x} \simeq \boldsymbol{g} \tag{2.15}$$

where $g_l = h_l s_l^n$, $\mathbf{F} = (\mathbf{V}_{m+1}, -\tilde{\mathbf{H}}\mathbf{V}_n)$ with $\tilde{\mathbf{H}} = \mathrm{diag}(h_i)$, $i = 1, \ldots, L$ and $\mathbf{V}_n$ Vandermonde matrix [110]

$$\mathbf{V}_n = \begin{bmatrix} 1 & s_1 & s_1^2 & \ldots & s_1^n \\ 1 & s_2 & s_2^2 & \ldots & s_2^n \\ \vdots & \vdots & \vdots & \ddots & \\ 1 & s_L & s_L^2 & \ldots & s_L^n \end{bmatrix} \tag{2.16}$$

based on the available frequency points s_l with $n+1$ columns. Minimizing $\|\boldsymbol{e}(\boldsymbol{x})\|$ or $\|\boldsymbol{r}(\boldsymbol{x})\|$ is not equivalent due to the weight $d(s_l; \boldsymbol{x})$, therefore the SK iteration [60] tries to overcome this limitation using an iteration-dependent residual, defined as

$$r_l^{\nu}(\boldsymbol{x}_\nu) = \frac{d(s_l; \boldsymbol{x}_\nu) h_l - n(s_l; \boldsymbol{x}_\nu)}{d(s_l; \boldsymbol{x}_{\nu-1})} \tag{2.17}$$

where the normalization weight $d(s_l; \boldsymbol{x}_{\nu-1})$ is known from the previous iteration $\nu - 1$. The iteration-dependent vector x_ν which minimizes the iteration-dependent residual (2.17) can be found solving the overdetermined linear system

$$\mathbf{M}_{\nu-1}\mathbf{F}\boldsymbol{x}_\nu \simeq \mathbf{M}_{\nu-1}\boldsymbol{g} \tag{2.18}$$

where $\mathbf{M}_{\nu-1} = \mathrm{diag}(m_i^{(\nu-1)})$ with $i = 1, \ldots, L$ and $m_i^{(\nu-1)} = d^{-1}(s_i; \boldsymbol{x}_{\nu-1})$. In case of convergence, as $\nu \to \infty$ the minimization of (2.17) is equivalent to minimizing (2.13). In practice some numerical issues arise: it is well known that Vandermonde matrices and their compositions are very ill-conditioned [111], moreover raw input data can be affected by noise thus making the identification problem more difficult.

In order to avoid those issues a general basis expansion can be used for the numerator and denominator in (2.11), i.e.

$$h(s; \boldsymbol{x}) = \frac{n(s; \boldsymbol{x})}{d(s; \boldsymbol{x})} = \frac{\sum_{j=0}^{m} c_j \phi_j(s)}{\sum_{j=0}^{n} d_j \phi_j(s)} \tag{2.19}$$

with $\boldsymbol{x}$ collecting the unknown coefficients c_j, d_j leading to the so-called Generalized-SK iteration [112]. A typical choice is to use partial fraction basis functions associated to a set of prescribed poles $q_j, j = 1, \ldots, n$, i.e.

$$\phi_0(s) = 1, \quad \text{and} \quad \phi_j(s) = \frac{1}{s - q_j}, \quad j = 1, \ldots, n. \tag{2.20}$$

Substituting (2.20) into (2.19) leads to

$$h(s;\boldsymbol{x}) = \frac{n(s;\boldsymbol{x})}{d(s;\boldsymbol{x})} = \frac{c_0 + \sum_{j=1}^{n} \frac{c_j}{s-q_j}}{1 + \sum_{j=1}^{n} \frac{d_j}{s-q_j}} \tag{2.21}$$

which is equivalent to the model in (2.11). Indeed supposing that c_j, d_j and the basis poles q_j are known, (2.21) can be converted in a standard rational form with the zeros of the numerator z_j, and the zeros of the denominator p_j such that

$$h(s;\boldsymbol{x}) = \frac{n(s;\boldsymbol{x})}{d(s;\boldsymbol{x})} = c_0 \frac{\prod_{j=1}^{n} \frac{s-z_j}{s-q_j}}{\prod_{j=1}^{n} \frac{s-p_j}{s-q_j}} = c_0 \frac{\prod_{j=1}^{n} (s-z_j)}{\prod_{j=1}^{n} (s-p_j)} \tag{2.22}$$

where it is clear that the poles q_j cancel out being common to both numerator and denominator. The GSK iteration is thus obtained by replacing the monomials s_l^j in (2.16) with $\phi_j(s_l)$.

A simple update on the basis poles and functions of each iteration leads to the VF algorithm: starting from an arbitrary guess of the model poles used to define the basis functions (2.20), the non-linear problem (2.13) is solved using one GSK; then the initial basis poles are improved by using, at the second iteration, the set p_j defined in (2.22) to construct the partial fraction basis functions. The process is then iterated until convergence. A more detailed description of VF algorithm can be found in Appendix B or in [62]. No more details are provided here since in the following VF is used as an identification engine, the main focus will be in preprocessing of the data and post-processing of the model.

One drawback of VF appears when dealing with devices with large ports counts like TSV, packages and PDNs. Since the complexity of VF in the most advanced formulation [98] scales as $\mathcal{O}(P^2LN^2)$ per iteration, the identification of devices having more than one hundred ports (P) and requiring several frequency samples (L) for an accurate characterization will run out-of-memory on commodity hardware, and will take a long time on high performance servers. Therefore a clever reformulation of the identification problem aimed at reducing the impact of ports (P) count and number of samples (L) on the overall complexity of Vector Fitting (VF) is of great interest. Next section introduces an innovative technique, the so called *compressed macromodeling*. This new methodology allows to perform the identification

of large ports count devices, accurately sampled in frequency, on commodity hardware (laptop) and in a short time compared to standard identification procedures.

2.3 Compressed macromodeling

In this Section an approach for improving the efficiency of rational fitting and passivity enforcement for medium and large-scale structures is presented. Problems characterized by possibly hundreds of ports and requiring thousands of internal states for their models are addressed. Requirements for problems of such complexity arise, as discussed in Section 1.2, in power bus modeling and optimization, chip-package co-design, TSV and NoC interconnects for 3D packages and mixed-signal system design.

The basic idea behind the proposed strategy can be easily understood considering a generic P-port electrical interconnect structure characterized through tabulated scattering frequency samples $\mathbf{S}_l \in \mathbb{C}^{P\times P}$ at frequencies ω_l, with $l = 1, \ldots, L$. This raw data is usually available from field simulations or direct measurements. The VF algorithm from Section 2.2 is routinely used to fit these data samples with a rational model

$$\mathbf{S}(s) = \mathbf{S}_\infty + \sum_{n=1}^{N} \frac{\mathbf{R}_n}{s - p_n}, \tag{2.23}$$

where p_n are the poles of the macromodel, $\mathbf{R}_n$ are the associated residue matrices, and $\mathbf{S}_\infty$ is the direct coupling term. Standard formulations of the VF algorithm [62] minimize the global model error (2.10) through an iterative sequence of linear least squares solutions. Since the compression strategy presented here is complementary to the VF implementation, a detailed description of VF algorithm is not reported here, more details can be found in Appendix B or [62].

The main idea of the compression scheme is presented through an example. Figure 2.2 depicts several scattering responses of a high-speed connector. As it can be seen the various responses that are depicted look very similar, with only marginal differences. Of course, these differences may be important, so they should be preserved in the final macromodel. However, it is conceivable that all these responses can be represented as a linear superposition of selected "representative" responses or, more formally, "basis functions". Therefore expansions of the form

$$S_{ij}(s) \simeq \sum_{q=1}^{\rho} \alpha_q^{(i,j)} w_q(s), \tag{2.24}$$

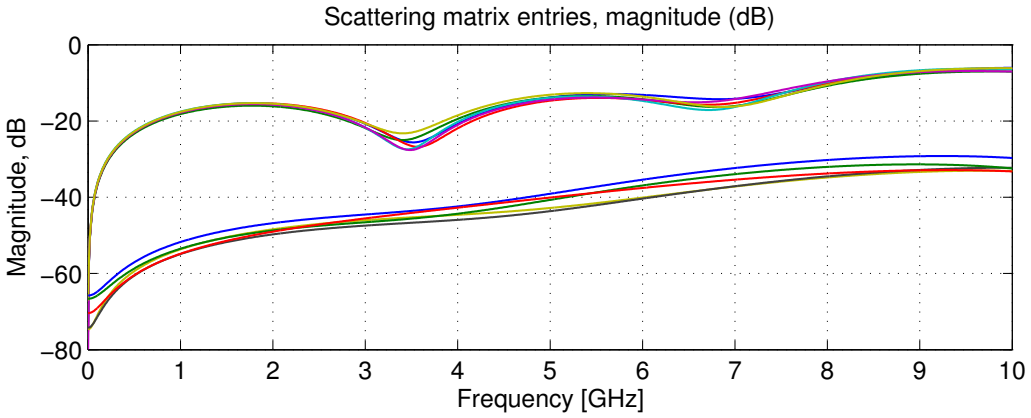

Figure 2.2: Various scattering responses of a high-speed connector (top curves: reflection coefficients, bottom curves: crosstalks).

with constant coefficients $\alpha_q^{(i,j)}$ and frequency-dependent "basis functions" $w_q(s)$, are suited for a clever reduction of the dataset. It is clear that if the number of required basis functions $w_q(s)$ is much smaller than the total number of responses, $\rho \ll P^2$, it is possible to achieve a significant computational cost reduction by applying VF to the few functions $w_q(s)$, rather than to the complete set of P^2 raw scattering responses. This idea is developed in the following Subsection 2.3.1, relying on the well known Singular Values Decomposition [110].

2.3.1 SVD-based compression

Consider the set of raw scattering samples $\mathbf{S}_l$, $\forall l$. For each selected frequency ω_l, all elements of the scattering matrix are stacked into a single row-vector $\boldsymbol{x}_l \in \mathbb{C}^{P^2}$, constructed as $\boldsymbol{x}_l = \text{vec}(\mathbf{S}_l)^T$. The vec() operator stacks all columns of its matrix element into a single column vector. More precisely, element $(\mathbf{S}_l)_{ij}$ with $1 \leq i, j \leq P$ corresponds to element $(\boldsymbol{x}_l)_k$ for $1 \leq k \leq P^2$ through

$$\begin{aligned} k &= i + (j-1)P \\ i &= 1 + \text{mod}(k-1, P) \\ j &= \lceil k/P \rceil \end{aligned} \tag{2.25}$$

where $\text{mod}(a, b)$ returns the remainder of the integer division a/b and $\lceil c \rceil$ is the ceil operator that returns the smallest integer not less than c. The mapping $(i, j) \leftrightarrow k$ in (2.25) will be used consistently during the presentation. All the vectors $\boldsymbol{x}_l$ corresponding to different frequencies ω_l are now collected

as rows in a matrix $\mathbf{X} \in \mathbb{C}^{L \times P^2}$, i.e.

$$\mathbf{X} = \begin{bmatrix} \longleftarrow & \boldsymbol{x}_1 & \longrightarrow \\ \vdots & \vdots & \vdots \\ \longleftarrow & \boldsymbol{x}_L & \longrightarrow \end{bmatrix} = \begin{bmatrix} \uparrow & \cdots & \uparrow \\ \boldsymbol{z}_1 & \cdots & \boldsymbol{z}_{P^2} \\ \downarrow & \cdots & \downarrow \end{bmatrix}. \tag{2.26}$$

Each row $\boldsymbol{x}_l$ of this matrix corresponds to a single frequency ω_l, while each column $\boldsymbol{z}_k$ collects all frequency samples of a single scattering response $(\boldsymbol{z}_k)_l = S_{ij}(\jmath\omega_l)$.

Assume that the P^2 scattering responses can be represented as an approximate sum of few basis functions. This implies that the column span of matrix $\mathbf{X}$ can be safely approximated by projection onto a subspace $\mathcal{W}$ having a dimension $\rho \ll P^2$. Several alternatives are available for constructing this subspace. In this work, the Singular Value Decomposition (SVD) is used since it provides a full control over the approximation error [113].

A direct application of SVD to matrix $\mathbf{X}$ leads to

$$\mathbf{X} = \widetilde{\mathbf{U}}\widetilde{\mathbf{\Sigma}}\widetilde{\mathbf{V}}^H = \widetilde{\mathbf{W}}\widetilde{\mathbf{V}}^H \tag{2.27}$$

where $\widetilde{\mathbf{U}}$ and $\widetilde{\mathbf{V}}$ are complex unitary matrices collecting the left and right singular vectors and $\widetilde{\mathbf{\Sigma}}$ collects the sorted real and positive singular values $\widetilde{\sigma}_q$ on its main diagonal. Matrix $\widetilde{\mathbf{W}} = \widetilde{\mathbf{U}}\widetilde{\mathbf{\Sigma}}$ is orthogonal with each column $\widetilde{\boldsymbol{w}}_q$ scaled by the corresponding singular value, $\|\widetilde{\boldsymbol{w}}_q\| = \widetilde{\sigma}_q$. The k-th column of $\mathbf{X}$ is thus represented, using (2.27), as

$$\boldsymbol{z}_k = \sum_q \widetilde{v}^*_{kq} \widetilde{\boldsymbol{w}}_q \,. \tag{2.28}$$

This expression is exact, with no approximation error, if all singular values/vectors are considered in the expansion. Each sampled scattering response is thus represented as a superposition of "basis vectors" $\widetilde{\boldsymbol{w}}_q$, whose norm decreases uniformly with increasing q.

The coefficients $\widetilde{v}^*_{kq}$ are complex-valued constants. Since a real expansion coefficient is needed in order to guarantee the causality and the realness of each element in the expansion (2.24), the SVD is slightly modified by splitting real and imaginary parts $\mathbf{X} = \mathbf{X}' + \jmath\mathbf{X}''$ where $\mathbf{X}', \mathbf{X}'' \in \mathbb{R}^{L \times P^2}$, or equivalently

$$\mathbf{X} = \begin{bmatrix} \mathbf{I}_L & \jmath\mathbf{I}_L \end{bmatrix} \begin{bmatrix} \mathbf{X}' \\ \mathbf{X}'' \end{bmatrix} \tag{2.29}$$

where $\mathbf{I}_L$ is the identity matrix of size L. Then, a truncated SVD decomposition is performed, based on the optimized implementation for large matrices [114], where only the first ρ singular values are retained

$$\begin{bmatrix} \mathbf{X}' \\ \mathbf{X}'' \end{bmatrix} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T \simeq \bar{\mathbf{U}}\bar{\mathbf{\Sigma}}\bar{\mathbf{V}}^T \,, \tag{2.30}$$

where $\bar{\mathbf{U}} \in \mathbb{R}^{2L\times\rho}$, $\bar{\mathbf{\Sigma}} \in \mathbb{R}^{\rho\times\rho}$, $\bar{\mathbf{V}} \in \mathbb{R}^{P^2\times\rho}$ with $\rho \ll r = \min\{2L, P^2\}$, and $\bar{\mathbf{V}}$ is orthonormal, $\bar{\mathbf{V}}^T\bar{\mathbf{V}} = \mathbf{I}$. Defining now

$$\bar{\mathbf{W}} = \begin{bmatrix}\mathbf{I}_L & \jmath\mathbf{I}_L\end{bmatrix}\bar{\mathbf{U}}\bar{\mathbf{\Sigma}} \tag{2.31}$$

leads the low-rank approximation

$$\mathbf{X} \simeq \bar{\mathbf{X}} = \bar{\mathbf{W}}\bar{\mathbf{V}}^T. \tag{2.32}$$

Equivalently,

$$\boldsymbol{z}_k \simeq \sum_{q=1}^{\rho} v_{kq}\bar{\boldsymbol{w}}_q\,, \tag{2.33}$$

which is similar to (2.28) but has guaranteed real coefficients v_{kq}. The q-th column $\bar{\boldsymbol{w}}_q \in \mathbb{C}^L$ of $\bar{\mathbf{W}}$, collects all frequency samples that define the q-th basis function. Sharp bounds, in different norms, can be provided for the error between the original matrix $\mathbf{X}$ collecting all scattering data and its low-rank approximation $\bar{\mathbf{X}}$. Using the spectral norm, defined in Appendix A, leads to

$$\begin{aligned}\mathcal{E}_2 = \left\|\bar{\mathbf{X}} - \mathbf{X}\right\|_2 &= \left\|\begin{bmatrix}\mathbf{I}_L & \jmath\mathbf{I}_L\end{bmatrix}\left[\bar{\mathbf{U}}\bar{\mathbf{\Sigma}}\bar{\mathbf{V}}^T - \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T\right]\right\|_2\\ &\le \left\|\begin{bmatrix}\mathbf{I}_L & \jmath\mathbf{I}_L\end{bmatrix}\right\|_2\left\|\bar{\mathbf{U}}\bar{\mathbf{\Sigma}}\bar{\mathbf{V}}^T - \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T\right\|_2\\ &\le \sqrt{2}\sigma_{\rho+1}\,,\end{aligned} \tag{2.34}$$

where the last row follows from standard properties of the SVD decomposition. It follows that the accuracy of the approximation is fully controlled by the first neglected singular value $\sigma_{\rho+1}$. Using the Frobenius norm the error bound becomes

$$\mathcal{E}_F = \left\|\bar{\mathbf{X}} - \mathbf{X}\right\|_F \le \sqrt{2L}\sqrt{\sum_{n=\rho+1}^{r}\sigma_n^2}\,, \tag{2.35}$$

in terms of the cumulative energy of the neglected singular values in (2.30). The accuracy of the proposed compression strategy is demonstrated in Figure 2.3: The top panel depicts two scattering responses of the same connector already considered in Figure 2.2, together with the corresponding low-rank approximation. The difference is hardly visible; while the bottom panel reports the first three basis vectors $\bar{\boldsymbol{w}}_q$ in the corresponding expansion (2.33).

2.3.2 Fitting the basis functions

Once expansion (2.33) is available, a rational approximation of each basis vector $\bar{\boldsymbol{w}}_q$ is performed. Consider a row-vector of scalar functions of frequency

$$\boldsymbol{w}(s) = \begin{pmatrix}w_1(s) & w_2(s) & \dots & w_\rho(s)\end{pmatrix}\,, \tag{2.36}$$

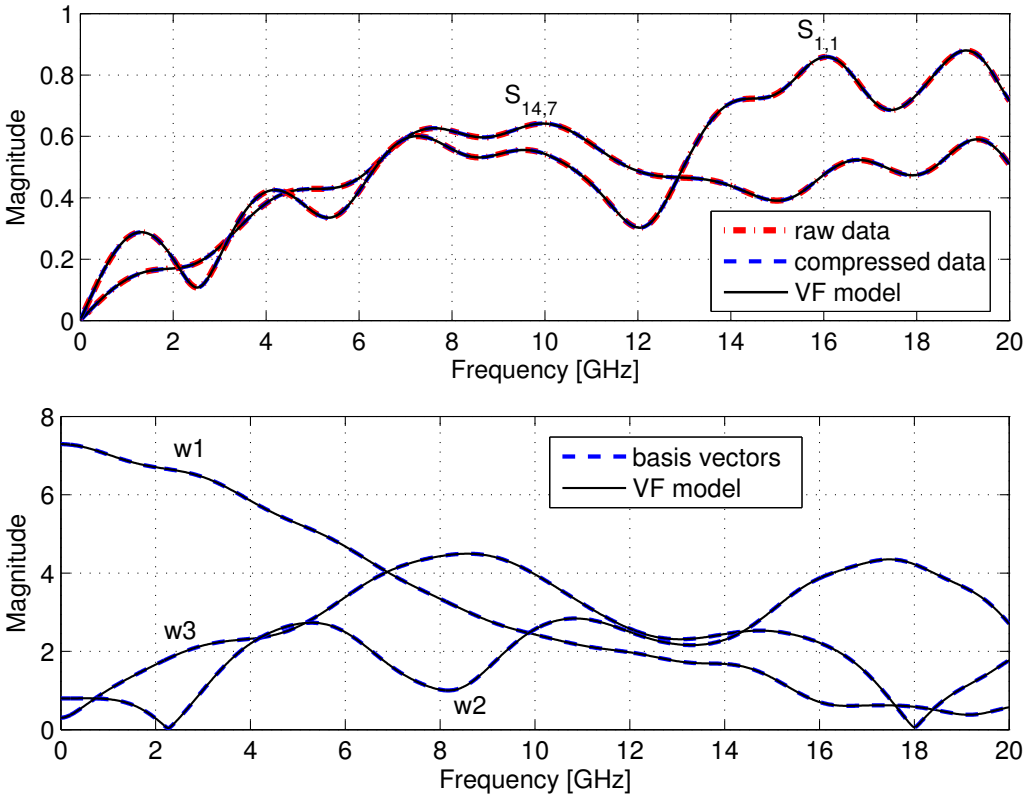

Figure 2.3: Top: raw scattering responses of a high-speed connector before compression (red dashed line), its compressed ($\rho = 30$) approximation (blue dashed line), and its low-rank rational approximation computed by VF (black line). Bottom: first three vectors $\bar{\boldsymbol{w}}_q$ (blue dashed lines) in expansion (2.33) and corresponding VF approximation (black line).

with each element assumed in rational form

$$w_q(s) = w_{q,\infty} + \sum_{n=1}^{N_w} \frac{r_{q,n}}{s - p_n} \,. \tag{2.37}$$

The unknown poles p_n, residues $r_{q,n}$ and direct coupling constants $w_{q,\infty}$ are computed by applying a standard VF run. Since only ρ independent responses are concurrently fitted instead ot P^2, it is expected that the runtime of the VF process is drastically reduced. This is indeed the case, as it is shown in Section 2.3.3. Note that a set of common poles p_n for all basis functions is used in $\boldsymbol{w}(s)$, since these will be used to reconstruct the original scattering matrix through (2.33), thus obtaining a global rational macromodel in form (2.23).

A successful fitting process with stable poles is guaranteed by the realness of the expansion coefficients in (2.33). In fact, post-multiplying (2.32) by $\bar{\mathbf{V}}$, since $\bar{\mathbf{V}}^T\bar{\mathbf{V}} = \mathbf{I}$, it follows

$$\bar{\boldsymbol{w}}_q \simeq \sum_{k=1}^{P^2} v_{kq}\boldsymbol{z}_k \,, \tag{2.38}$$

which shows that each basis vector can be represented as a linear combination of the raw scattering responses with real coefficients. This is sufficient to conclude that if the original responses are causal, each of the basis functions will be causal. Therefore, the rational approximation (2.37) is guaranteed to have stable poles p_n, see [115].

A state-space realization can be constructed from (2.37) using standard techniques. For later convenience, this realization is constructed for the transpose system, which has a Single-Input Multiple-Output structure, as

$$\boldsymbol{w}(s)^T \leftrightarrow \left(\begin{array}{c|c} \mathbf{A}_w & \mathbf{B}_w \\ \hline \mathbf{C}_w & \mathbf{D}_w \end{array}\right) \tag{2.39}$$

with $\mathbf{A}_w \in \mathbb{R}^{N_w \times N_w}$, $\mathbf{B}_w \in \mathbb{R}^{N_w \times 1}$, $\mathbf{C}_w \in \mathbb{R}^{\rho \times N_w}$, $\mathbf{D}_w \in \mathbb{R}^{\rho \times 1}$. A reshaped global rational macromodel is defined according to the expansion (2.32), as

$$\begin{aligned} \mathbf{X}^T(s) &= \bar{\mathbf{V}}\boldsymbol{w}^T(s) = \\ &= \bar{\mathbf{V}}\mathbf{D}_w + \bar{\mathbf{V}}\mathbf{C}_w(s\mathbf{I}_{N_w} - \mathbf{A}_w)^{-1}\mathbf{B}_w \,, \end{aligned} \tag{2.40}$$

where $\mathbf{X}^T(s)$ is a column vector of P^2 rational responses. Finally, a global rational macromodel for the original scattering representation is obtained with a simple reshape operation

$$\mathbf{S}(s) = \mathrm{mat}(\mathbf{X}^T(s)) \,, \tag{2.41}$$

where the mat$(\cdot)$ operator reconstructs a $P \times P$ matrix starting from the corresponding $P^2 \times 1$ vector vec$(\mathbf{S})$. It is easy to show that a state-space realization of $\mathbf{S}(s)$ can be obtained as

$$\mathbf{S}(s) \leftrightarrow \left(\begin{array}{c|c} \mathbf{A} & \mathbf{B} \\ \hline \mathbf{C} & \mathbf{D} \end{array}\right) \tag{2.42}$$

with

$$\begin{array}{lll} \mathbf{A} & = \mathbf{I}_P \otimes \mathbf{A}_w\,, & \mathbf{B} = \mathbf{I}_P \otimes \mathbf{B}_w\,, \\ \mathbf{C} & = \mathbf{\Psi}(\mathbf{I}_P \otimes \mathbf{C}_w)\,, & \mathbf{D} = \mathbf{\Psi}(\mathbf{I}_P \otimes \mathbf{D}_w)\,, \end{array} \tag{2.43}$$

where $\otimes$ denotes the Kronecker matrix product [116] and

$$\mathbf{\Psi} = \left(\bar{\mathbf{V}}_1\ \bar{\mathbf{V}}_2\ \cdots \bar{\mathbf{V}}_P\right) \tag{2.44}$$

with $\bar{\mathbf{V}}_j \in \mathbb{R}^{P\times\rho}$ collecting the P rows $\{j(P-1)+1,\ldots,jP\}$ of matrix $\bar{\mathbf{V}}$

$$\bar{\mathbf{V}} = \begin{bmatrix} \bar{\mathbf{V}}_1 \\ \vdots \\ \bar{\mathbf{V}}_P \end{bmatrix} \tag{2.45}$$

In (2.43) the size of the various matrices is $\mathbf{A} \in \mathbb{R}^{N\times N}$, $\mathbf{B} \in \mathbb{R}^{N\times P}$, $\mathbf{C} \in \mathbb{R}^{P\times N}$, $\mathbf{D} \in \mathbb{R}^{P\times P}$, where $N = N_w P$ denotes the global dynamic order of the realization. The transfer matrix of the compressed macromodel associated to (2.43) reads

$$\mathbf{S}(s) = \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B} + \mathbf{D}\,. \tag{2.46}$$

Once the rational approximation (2.37) is available, $\boldsymbol{w}(s)$ is evaluated at each raw frequency point ω_l and the results are collected as rows in matrix $\widehat{\mathbf{W}} \in \mathbb{C}^{L\times\rho}$, which in turn is used to reconstruct the samples of the global rational macromodel, collected in matrix $\widehat{\mathbf{X}} = \widehat{\mathbf{W}}\bar{\mathbf{V}}^T$. Due to the orthonormality of the columns of $\bar{\mathbf{V}}$ it follows

$$\left\|\bar{\mathbf{X}} - \widehat{\mathbf{X}}\right\|_2 = \left\|\bar{\mathbf{W}}\bar{\mathbf{V}}^T - \widehat{\mathbf{W}}\bar{\mathbf{V}}^T\right\|_2 = \left\|\bar{\mathbf{W}} - \widehat{\mathbf{W}}\right\|_2. \tag{2.47}$$

This implies that the construction of a global rational model starting from the rational basis functions is well-behaved, since it results in a fitting error that is identical to the fitting error achieved in the construction of the low-rank system $\boldsymbol{w}(s)$. The global approximation error between raw scattering samples and global rational macromodel can thus be characterized as

$$\begin{aligned} \delta_2 = \left\|\mathbf{X} - \widehat{\mathbf{X}}\right\|_2 &\le \left\|\mathbf{X} - \bar{\mathbf{X}}\right\|_2 + \left\|\bar{\mathbf{X}} - \widehat{\mathbf{X}}\right\|_2 \\ &\le \sqrt{2}\sigma_{\rho+1} + \left\|\bar{\mathbf{W}} - \widehat{\mathbf{W}}\right\|_2, \end{aligned} \tag{2.48}$$

where the individual contributions of SVD truncation and VF approximation are explicit.

Table 2.1: Benchmark structures: L is the number of raw frequency samples, P the number of ports, ρ the number of basis functions; N_x and N_w denote the number of poles used for full and compressed fitting, respectively.

Case	L	P	P^2	ρ	N_w	N_x
1	471	12	144	17	20	22
2	690	48	2304	24	27	28
3	1001	56	3136	30	30	30
4	572	25	625	5	5	5
5	71	92	8464	22	22	23
6	570	34	1156	40	57	58
7	1001	24	576	13	12	12
8	1228	83	6889	31	30	31
9	100	8	64	6	29	29
10	197	245	60025	14	45	29
11	13	52	2704	3	3	3
12	40	800	640000	8	8	8
13	572	41	1681	10	11	11
14	141	542	293764	16	21	-
15	1000	34	1156	10	10	15
16	501	28	784	9	12	16
17	364	20	400	40	58	59
18	367	181	32761	6	24	39

2.3.3 Compressed fitting examples

Here are introduced some benchmark cases of practical interest. Table 2.1 lists a total of 18 interconnect structures, characterized by different number of ports P and raw frequency samples L. These structures include high-speed connectors (cases 2, 3, 7), PCB interconnects (cases 9, 17), package interconnects (cases 5, 8, 13, 15, 16), power or mixed signal/power distribution networks (cases 1, 4, 6, 10, 11, 14, 18), and Through Silicon Via (TSV) fields (case 12). All raw frequency samples were obtained from 2D or 3D field characterizations.

The last column in Table 2.1 shows the number of poles N_x that were required by Vector Fitting to fit the full set of responses with a global model-vs-data deviation $\|\delta\mathbf{X}\|_2 = \delta_2$ defined in (2.48). The publicly available VF code [117] was used for these tests, by iteratively increasing the number of poles until the above accuracy condition was met.

In this subsection, the performance of standard and compressed VF are

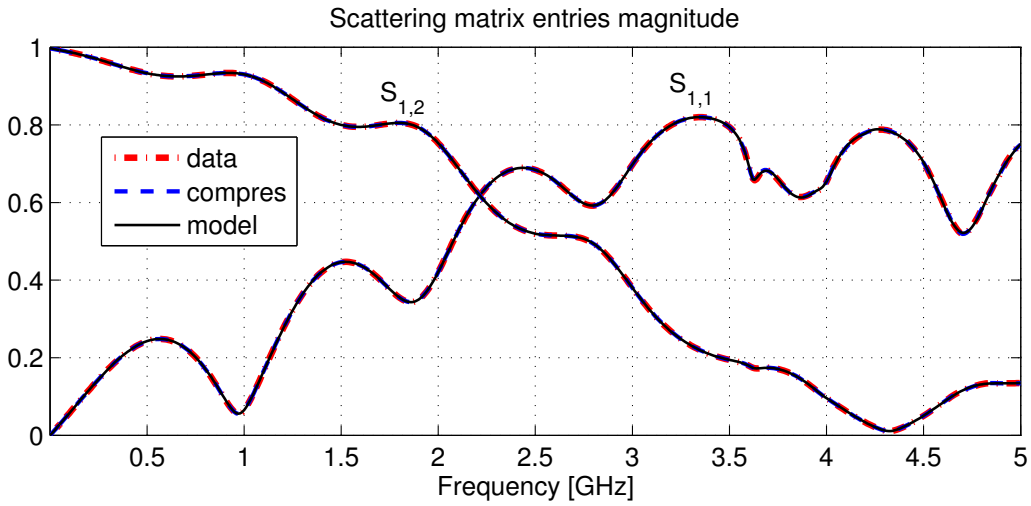

Figure 2.4: Two sample scattering responses for case 6 before (dash-dotted lines) and after (dashed lines) compression, compared to the compressed rational fitted model responses (solid lines).

compared. To this end, the compression error $\mathcal{E}_2$, defined in (2.34), and VF approximation error are fixed to a constant value defined later (usually close to 0.1). This choice results in a number of basis functions ρ and in a number of poles for the basis functions N_w, also reported in Table 2.1. These results show collectively that

- the number of basis functions always results $\rho \ll P^2$, therefore the computational complexity of the compressed VF run always results much less than the standard full VF;
- the number of poles required for the compressed and the full macromodels is comparable, $N_w \simeq N_x$, showing that the compression strategy does not create spurious or artificial components in the basis functions that would require an excessive number of poles for their fitting.

Figure 2.4 compares the results of full and compressed macromodels to the raw scattering responses for benchmark case 6, showing that an excellent accuracy is obtained using both techniques. Figure 2.5 shows some of the corresponding basis functions together with their rational fitted models.

Table 2.2 reports the execution time in seconds that was required for compression, denoted as T_{SVD} (based on [114]), for fitting the ρ basis functions and constructing the compressed macromodel, denoted as T_{VFW}, and for applying standard VF to the full set of raw responses, denoted as T_{VFX}. The

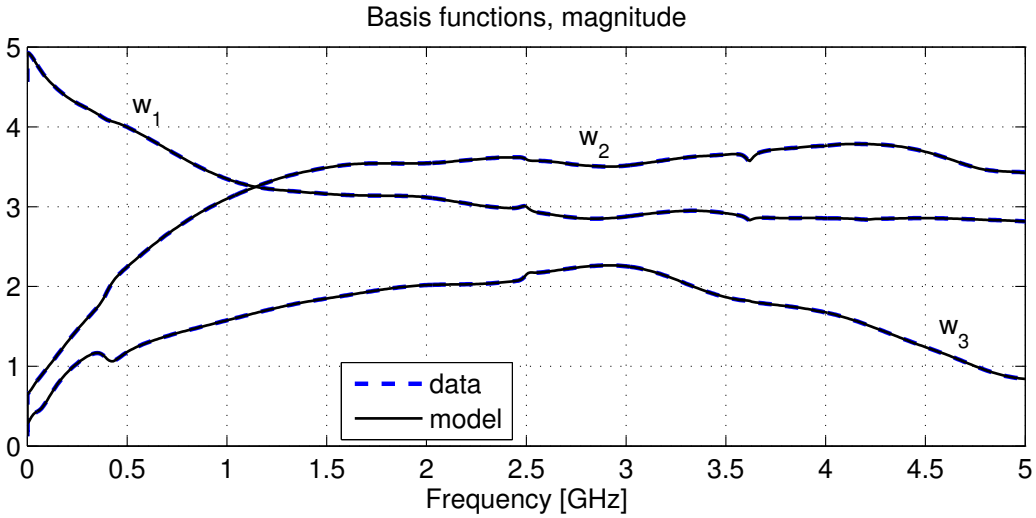

Figure 2.5: First three basis functions for case 6. Original frequency samples $\bar{\boldsymbol{w}}_q$ (dashed lines) and rational model $w_q(s)$ (solid lines).

Table 2.2: CPU time in seconds required for data compression (T_{SVD}), based on the SVD optimized implementation for large matrices [114], and compressed fitting (T_{VFW}) compared to full fitting (T_{VFX}).

Case	T_{SVD} [s]	T_{VFW} [s]	T_{VFX} [s]	Speedup
1	0.03	0.66	4.2	6.03
2	0.8	1.7	183.5	70.5
3	1.3	3.7	419.7	82.4
4	0.28	0.02	1.42	4.6
5	0.7	0.23	59.4	63
6	0.33	10.6	355.2	32.1
7	0.37	0.28	11.6	17.8
8	3.2	4.6	1273	160
9	0.004	0.2	0.94	4.44
10	2.4	1.2	1609	437.1
11	0.01	0.006	0.2	12
12	12.8	0.04	592	45.8
13	1.7	0.3	17.8	8.8
14	9.2	0.8	-	-
15	4.8	1.5	39	6.1
16	0.3	0.154	12	24.2
17	0.15	8.05	77.3	9.4
18	2.2	0.4	2074	760.4

overall speedup reported in the last column demonstrates how effective can the compressed macromodeling approach be for those cases that are characterized by a large port count or a large number of frequency samples. For case 14, standard VF could not even be applied due to an excessive memory requirement.

2.3.4 Passivity of compressed macromodels

There is no guarantee that the global macromodel (2.46) with state-space matrices (2.43) is passive. It is however possible to explicitly enforce model (asymptotic) stability by constraining the poles p_n to have a strictly negative real part, a standard practice in VF applications [62]. The fundamental condition under which a scattering transfer matrix $\mathbf{S}(s)$ (2.102) represents a passive macromodel is bounded realness [118, 119, 120, 115]. A transfer matrix $\mathbf{S}(s)$ is Bounded Real (BR) if

- each element of $\mathbf{S}(s)$ is defined and analytic in $\mathbb{R}\mathrm{e}\{s\} > 0$;
- $\mathbf{S}^*(s) = \mathbf{S}(s^*)$;
- $\mathbf{\Theta}(s) = \mathbf{I} - \mathbf{S}(s)^H\mathbf{S}(s) \succeq 0$ for $\mathbb{R}\mathrm{e}\{s\} > 0$.

The first two conditions are guaranteed if the state-space realizations (2.102) is real-valued and asymptotically stable [115]. Under these assumptions the condition on $\mathbf{\Theta}(s)$ can be relaxed and checked only on the imaginary axis $s = \jmath\omega$

$$\mathbf{\Theta}(\jmath\omega) \succeq 0\,, \quad \forall\omega\,, \tag{2.49}$$

which in turn is equivalent to requiring that all singular values of $\mathbf{S}(\jmath\omega)$ must be uniformly bounded by one at any frequency

$$\sigma_i \leq 1\,, \quad \forall\sigma_i \in \sigma(\mathbf{S}(\jmath\omega))\,, \quad \forall\omega\,. \tag{2.50}$$

Considering that $\sigma_i = \sqrt{1-\lambda_i}$, where $\lambda \in \lambda(\mathbf{\Theta}(\jmath\omega))$ are the eigenvalues of $\mathbf{\Theta}(\jmath\omega)$, it follows that (2.49) is equivalent to

$$\lambda_i \geq 0\,, \quad \forall\lambda_i \in \lambda(\mathbf{\Theta}(\jmath\omega))\,, \quad \forall\omega. \tag{2.51}$$

The passivity condition (2.51), which can be checked either via adaptive frequency sampling [104], see Section 2.5, or through identification of imaginary eigenvalues of the associated Hamiltonian matrix [121], can be violated over finite or infinite frequency bands. In particular, this second case occurs if the model is not asymptotically passive, i.e. $\min\lambda\{\mathbf{\Theta}(\infty)\} < 0$. In this situation, asymptotic passivity can be recovered by perturbing just the direct

coupling matrix $\mathbf{D}$; this is the subject of Section 2.3.5. Then, in Section 2.4 a global passivity enforcement scheme for enforcing (2.51) at all frequencies $\omega \in \mathbb{R}$ is presented.

2.3.5 Asymptotic passivity enforcement

The macromodel (2.46) is asymptotically (strictly) passive if

$$\|\mathbf{D}\|_2 \leq \delta_p < 1\,, \tag{2.52}$$

where δ_p is some desired passivity threshold. In case (2.52) is not verified, matrix $\mathbf{D}$ is modified so that this condition is met. Of course it is more efficient to operate directly on the compressed macromodel (2.42), therefore a perturbation vector Δ_w is added to the corresponding direct coupling vector $\boldsymbol{d}_w$, preserving the projection coefficients in matrix $\boldsymbol{\Psi}$. The perturbed matrix results

$$\mathbf{D}_p = \boldsymbol{\Psi}[\mathbf{I}_P \otimes (\boldsymbol{d}_w + \Delta_w)]\,, \tag{2.53}$$

with

$$\mathbf{D}_p - \mathbf{D} = \boldsymbol{\Psi}(\mathbf{I}_P \otimes \Delta_w)\,. \tag{2.54}$$

The minimal perturbation of (2.54), in the standard 2-norm, should be used to achieve asymptotic passivity. This leads to the following formulation

$$\min_{\Delta_w} \|\boldsymbol{\Psi}(\mathbf{I}_P \otimes \Delta_w)\|_2 \quad \text{s.t.} \quad \|\mathbf{D}_p\|_2 \leq \delta_p\,. \tag{2.55}$$

The solution of (2.55) is now addressed using various different approaches, with results presented and compared in Section 2.3.6.

Once a solution Δ_w of (2.55) is available, an asymptotically passive macromodel is constructed by

1. constructing the vector $\boldsymbol{d}_p = \boldsymbol{d}_w + \Delta_w$;

2. subtracting the q-th component $d_{p,q}$ of this vector from the frequency samples of the q-th basis function $\bar{\boldsymbol{w}}_q$ by redefining

$$\bar{\boldsymbol{w}}_q \leftarrow \bar{\boldsymbol{w}}_q - d_{p,q} \tag{2.56}$$

3. fitting the resulting frequency samples with a strictly proper rational function

$$w_q(s) = \sum_{n=1}^{N_w} \frac{r_{q,n}}{s - p_n}\,, \tag{2.57}$$

where the poles p_n are kept fixed to the poles of the original unperturbed macromodel (2.37);

4. defining the state-space realization of the compressed macromodel as in (2.42), but with $\boldsymbol{d}_w$ replaced by $\boldsymbol{d}_p$.

The following methods can be used to identify the perturbation vector Δ_w and the corresponding $\bar{\boldsymbol{w}}_q$.

Direct scaling

The easiest way to enforce the asymptotic passivity is through the direct rescaling

$$\boldsymbol{d}_p = \boldsymbol{d}_w \frac{\delta_p}{\|\mathbf{D}\|_2}, \quad \mathbf{D}_p = \boldsymbol{\Psi}(\mathbf{I}_P \otimes \boldsymbol{d}_p). \tag{2.58}$$

This definition imposes asymptotic passivity by construction, but does not guarantee that the asymptotic model perturbation $\|\boldsymbol{\Psi}(\mathbf{I}_P \otimes \Delta_w)\|_2$ is minimized, as required by (2.55). However, since the compressed macromodel will be re-generated via a new constrained vector fitting run (2.57), the asymptotic perturbation will have a significant effect only at high frequencies, resulting in a quite acceptable accuracy within the modeling band. These statements will be validated through numerical examples in 2.3.6. Therefore, this scaling method is actually quite competitive due to its simplicity with respect to the more precise approaches that follow.

Linearization

The method described in this section is based on two simplifications of (2.55). First, the norm of Δ_w is minimized instead of the norm of $\mathbf{D}_p - \mathbf{D}$. Second, the constraint $\|\mathbf{D}_p\|_2 \leq \delta_p$ is replaced by an approximate constraint on Δ_w based on a linearization process. These two conditions lead to a problem of smaller size with respect to (2.55), which should require less computational effort for its solution.

Start with a SVD decomposition of $\mathbf{D} = \mathbf{L}\boldsymbol{\Sigma}_D\mathbf{R}^T$. Denoting the singular values as ς_i, $i = 1, \ldots, P$ with the associated left and right singular vectors $\boldsymbol{l}_i$ and $\boldsymbol{r}_i$ it follows

$$\varsigma_i = \boldsymbol{l}_i^T \mathbf{D} \boldsymbol{r}_i. \tag{2.59}$$

Apply now the same projection to the perturbed direct coupling matrix $\mathbf{D}_p$ obtaining

$$\boldsymbol{l}_i^T \mathbf{D}_p \boldsymbol{r}_i = \varsigma_i + \boldsymbol{l}_i^T \boldsymbol{\Psi}(\mathbf{I}_P \otimes \Delta_w)\boldsymbol{r}_i. \tag{2.60}$$

Note that this quantity is not equal to the i-th singular value $\varsigma_{p,i}$ of $\mathbf{D}_p$, but it provides only a first-order approximation. Thus, condition

$$\boldsymbol{l}_i^T \mathbf{D}_p \boldsymbol{r}_i \leq \delta_p \tag{2.61}$$

corresponds to a linearised projection of constraint $\|\mathbf{D}_p\|_2 \leq \delta_p$. Using (2.60), after some straightforward algebraic manipulations leads to

$$(\boldsymbol{r}_i^T \otimes \boldsymbol{l}_i^T)\bar{\mathbf{V}}\Delta_w \leq \delta_p - \varsigma_i \,. \tag{2.62}$$

Collecting the various constraints (2.62) for all i leads to the linear underdetermined system

$$\mathbf{M}\Delta_w = \boldsymbol{b} \,, \tag{2.63}$$

where the number of rows in $\mathbf{M}$ defines the number of singular values of $\mathbf{D}$ being perturbed. Among all vectors Δ_w satisfying (2.63), the minimum-norm solution is needed, which is available in closed form as

$$\Delta_w = \mathbf{M}^\dagger \boldsymbol{b} \,, \tag{2.64}$$

with $\mathbf{M}^\dagger$ denoting the Moore-Penrose pseudoinverse of $\mathbf{M}$.

Due to the approximate nature of (2.62), the solution (2.64) of (2.63) does not guarantee that $\|\mathbf{D}_p\|_2 \leq \delta_p$. Therefore, the process can be iterated until this condition is achieved. At each iteration, two slightly different constraints can be used, leading to different numerical schemes

1. system (2.63) si formed by collecting all P singular values, setting at the right hand side

$$b_i = \begin{cases} \delta_p - \varsigma_i & \varsigma_i > \delta_p \,, \\ 0 & \varsigma_i \leq \delta_p \,. \end{cases} \tag{2.65}$$

   This choice tries to explicitly preserve those singular values that are already below the threshold δ_p.

2. only constraints with $\varsigma_i > \delta_p$ are formed, so that only the singular value terms exceeding the threshold δ_p are explicitly perturbed.

Linear Matrix Inequalities

The problem stated in (2.55) can be cast as a Linear Matrix Inequality (LMI) [122, 123]. In fact, introducing the slack variable γ, minimization of the objective function in (2.55) can be restated as

$$\min \gamma \quad \text{s.t.} \quad \begin{bmatrix} \gamma \mathbf{I}_P & \boldsymbol{\Psi}(\mathbf{I}_P \otimes \Delta_w) \\ (\mathbf{I}_P \otimes \Delta_w^T)\boldsymbol{\Psi}^T & \gamma \mathbf{I}_P \end{bmatrix} \succ 0 \,, \tag{2.66}$$

whereas the asymptotic passivity constraint is equivalent to

$$\begin{bmatrix} \delta_p \mathbf{I}_P & \mathbf{D} + \boldsymbol{\Psi}(\mathbf{I}_P \otimes \Delta_w) \\ \mathbf{D}^T + (\mathbf{I}_P \otimes \Delta_w^T)\boldsymbol{\Psi}^T & \delta_p \mathbf{I}_P \end{bmatrix} \succ 0 \,. \tag{2.67}$$

Expressions (2.66) and (2.67) form a system of LMI's. This formulation is based on convex constraints with a convex objective function. Therefore, its solution can be achieved numerically within arbitrary precision and with a finite number of steps using some specialized software. All results documented in the following were obtained with the SeDuMi package [124].

2.3.6 Numerical Results

Table 2.3 compares the asymptotic passivity enforcement results obtained by the various schemes presented in Sections 2.3.5 for those cases that resulted non-asymptotically passive after the compressed fitting stage. The maximum singular value $\|\mathbf{D}\|_2$ of the direct coupling matrix is reported for convenience in the second column. The four schemes are compared in terms of direct coupling perturbation amount $\mathbf{\Delta} = \mathbf{D}_p - \mathbf{D}$ measured in the spectral norm, number of iterations (when applicable), and total runtime. The latter includes not only the direct coupling perturbation, but also the computation of the perturbed residues and the construction of the global state-space realization, as described in Sec. 2.3.5.

The direct scaling method requires no iterations. Only the computation of the norm $\|\mathbf{D}\|_2$ is required. Scaling requires negligible time, so that the total runtime is practically used for recomputing the updated residue matrices. The linarization and the LMI methods instead require several iterations and require significantly larger runtime. These three methods fail for the largest cases 12 and 14 due to excessive memory occupation (LMI) or lack of convergence (linearisation) within a maximum number of 600 iterations. If converging, the linearization methods are faster than the LMI approach. However, the linearisation methods are not guaranteed to attain the optimal solution, as does the LMI approach. This is confirmed by the amount of perturbation, which is smallest for the LMI case among all other methods. It is worth noting that the simplistic direct scaling approach provides final perturbation errors that are comparable with the LMI scheme. Due to its efficiency, the direct scaling approach appears as the most competitive. Of course, in case the resulting perturbation is excessive, one can resort to the LMI scheme, which is guaranteed to be optimal though slow.

2.4 Global passivity enforcement

This section addresses the enforcement of global passivity for the macromodel (2.46) characterized by the state-space realization (2.43), assumed to be asymptotically stable and asymptotically passive. It is assumed that (2.51)

Table 2.3: Comparison of asymptotic passivity enforcement methods in terms of perturbation amount $\|\mathbf{\Delta}\|_2$, number of iterations ♯ it, and runtime in seconds. See text for additional details.

		Scaling		Linearization 1			Linearization 2			LMI		
Case	$\|\mathbf{D}\|_2$	$\|\mathbf{\Delta}\|_2$	Time [s]	$\|\mathbf{\Delta}\|_2$	♯ it	Time [s]	$\|\mathbf{\Delta}\|_2$	♯ it	Time [s]	$\|\mathbf{\Delta}\|_2$	♯ it	Time [s]
5	1.02	0.03	1	0.11	154	12	1.9	424	14	0.03	24	26.2
9	1.041	0.05	0.04	0.051	3	0.007	0.05	3	0.005	0.05	11	0.17
11	1.26	0.28	0.15	0.29	5	0.3	0.29	5	0.01	0.28	17	2.7
12	2.74	0.3	54.3	–	–	–	–	–	–	–	–	–
14	1.04	0.07	114	–	–	–	–	–	–	–	–	–
18	1	0.01	5	1.2	600	146	1.45	136	13.6	0.01	32	434

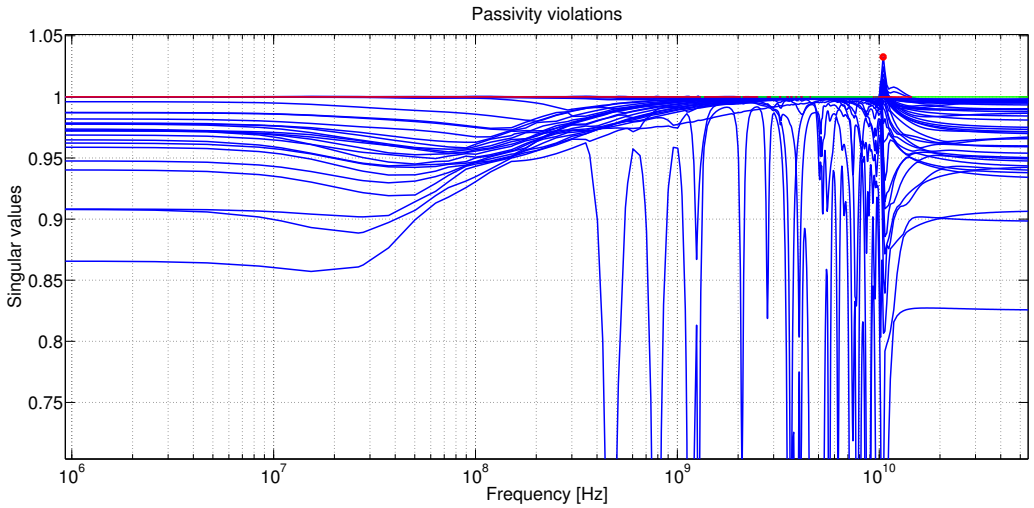

Figure 2.6: Singular value plot before passivity enforcement for the S-parameter state-space model of a PCB. Passivity violations are highlighted near $10GHz$ by a small red circle.

is violated at some frequencies $\omega \in \Omega$, where Ω is the union of finite-width frequency bands like in Figure 2.6, which refers to the state-space model of a PCB.

In order to enforce passivity, one of the standard perturbation approaches can be followed [121, 104]. Passivity violations can be identified via standard techniques [104] or using the parallel algorithm presented later in Section 2.5. The main difference in the present framework with respect to published results is that the system perturbation should not be arbitrary but structured, according to the form of (2.43). In the following only the state-to-output map is perturbed, i.e.

$$\mathbf{C}_p = \mathbf{C} + \mathbf{\Delta}_C \,, \tag{2.68}$$

where the perturbation term $\mathbf{\Delta}_C$ is defined as

$$\mathbf{\Delta}_C = \mathbf{\Psi}(\mathbf{I}_P \otimes \mathbf{\Delta}_{C_w}) \,. \tag{2.69}$$

As for the asymptotic passivity enforcement of Sec. 2.3.5, the expansion coefficients in matrix $\mathbf{\Psi}$ are preserved and only the lower-dimensional compressed macromodel (2.42) is perturbed.

2.4.1 Passivity enforcement

Consider a single frequency ω_0 at which condition (2.51) is violated by some negative eigenvalue $\lambda_i < 0$, and let the corresponding eigenvector of $\mathbf{\Theta}(\jmath\omega_0)$

(defined in Section 2.3.4) be $\boldsymbol{\zeta}_i$, normalized such that $\|\boldsymbol{\zeta}_i\|_2 = 1$. Applying (2.68) leads to a first-order approximation of the perturbed eigenvalue [125]

$$\lambda_{p,i} \simeq \lambda_i + \boldsymbol{\zeta}_i^H \boldsymbol{\Delta}_{\boldsymbol{\Theta}} \boldsymbol{\zeta}_i \,, \tag{2.70}$$

where

$$\boldsymbol{\Delta}_{\boldsymbol{\Theta}} \simeq -\mathbf{K}_0^H \boldsymbol{\Delta}_C \mathbf{S}_0 - \mathbf{S}_0^H \boldsymbol{\Delta}_C^T \mathbf{K}_0 \tag{2.71}$$

and

$$\mathbf{S}_0 = \mathbf{D} + \mathbf{C}\mathbf{K}_0 \,, \quad \mathbf{K}_0 = (\jmath\omega_0 \mathbf{I} - \mathbf{A})^{-1}\mathbf{B} \,. \tag{2.72}$$

Standard manipulations lead to

$$\lambda_{p,i} \simeq \lambda_i + \boldsymbol{t}_i \,\mathrm{vec}(\boldsymbol{\Delta}_C) \,, \tag{2.73}$$

where the row-vector $\boldsymbol{t}_i$ is defined as

$$\boldsymbol{t}_i = -2\mathrm{Re}\{(\mathbf{K}_0\boldsymbol{\zeta}_i)^T \otimes (\mathbf{S}_0\boldsymbol{\zeta}_i)^H\} \,. \tag{2.74}$$

Enforcing now $\lambda_{p,i} \geqslant 0$ leads to the following linear inequality constraint

$$\boldsymbol{t}_i \,\mathrm{vec}(\boldsymbol{\Delta}_C) \geqslant -\lambda_i \,. \tag{2.75}$$

Also an additional constraint is included, i.e.

$$\boldsymbol{t}_i \,\mathrm{vec}(\boldsymbol{\Delta}_C) \leqslant 1 - \lambda_i \tag{2.76}$$

to guarantee that the perturbed eigenvalue remains in [0,1], as required by the assumed scattering representation. The above constraints are built for all M eigenvalues λ_i to be perturbed, possibly at multiple frequencies, and formulated as

$$\begin{cases} \min \theta \\ \|\mathrm{vec}(\boldsymbol{\Delta}_C)\|_2^2 < \theta \\ \mathbf{T}\,\mathrm{vec}(\boldsymbol{\Delta}_C) \geqslant \boldsymbol{b} \end{cases} \tag{2.77}$$

where θ is a slack variable. The last row collects in a compact form all constraints (2.75)-(2.76).

Now the perturbation structure (2.69) is imposed. Using (2.44), it is easy to show that

$$\boldsymbol{\Delta}_C = \left(\bar{\mathbf{V}}_1 \boldsymbol{\Delta}_{C_w}, \ldots, \bar{\mathbf{V}}_P \boldsymbol{\Delta}_{C_w}\right) \,. \tag{2.78}$$

Applying the vec$(\cdot)$ operator to the i-th column block in (2.78) leads to

$$\mathrm{vec}(\bar{\mathbf{V}}_i \boldsymbol{\Delta}_{C_w}) = (\mathbf{I}_{N_w} \otimes \bar{\mathbf{V}}_i)\,\mathrm{vec}(\boldsymbol{\Delta}_{C_w}) \,, \tag{2.79}$$

so that (2.78) can be written in "vectorized" form as

$$\mathrm{vec}(\boldsymbol{\Delta}_C) = \boldsymbol{\Pi}\,\mathrm{vec}(\boldsymbol{\Delta}_{C_w}) \,, \tag{2.80}$$

where $\mathbf{\Pi} \in \mathbb{R}^{PN \times \rho N_w}$ is defined as

$$\mathbf{\Pi} = \begin{bmatrix} \mathbf{I}_{N_w} \otimes \bar{\mathbf{V}}_1 \\ \vdots \\ \mathbf{I}_{N_w} \otimes \bar{\mathbf{V}}_P \end{bmatrix} \tag{2.81}$$

Finally, defining $\mathbf{T}_w \in \mathbb{R}^{2M \times \rho N_w}$ as

$$\mathbf{T}_w = \mathbf{T}\mathbf{\Pi}\,, \tag{2.82}$$

the structured and compressed passivity enforcement problem reads

$$\begin{cases} \min \theta \\ \|\mathrm{vec}(\mathbf{\Delta}_{C_w})\|_2^2 < \theta \\ \mathbf{T}_w \,\mathrm{vec}(\mathbf{\Delta}_{C_w}) \geqslant \boldsymbol{b} \end{cases} \tag{2.83}$$

Note that matrix $\mathbf{\Pi}$ is never constructed in practice, since all constraints in (2.83) and in particular matrix $\mathbf{T}_w$ can be built directly using optimized code.

Comparing the standard formulation (2.77) with the compressed and structured formulation (2.83), it is evident that the latter is much more convenient, since the number of decision variables is reduced by a factor

$$\frac{\#\{\mathbf{\Delta}_{C_w}\}}{\#\{\mathbf{\Delta}_C\}} = \frac{\rho N_w}{PN} = \frac{\rho}{P^2} \ll 1\,. \tag{2.84}$$

This makes the cost for the solution of (2.83) practically negligible with respect to all other macromodeling steps. Note that the converse is typically the case in standard macromodeling, since passivity enforcement is usually the most demanding part of state of the art schemes. This big advantage is due to the particular state-space structure in (2.43).

2.4.2 Accuracy control

The formulations in (2.77) and (2.83) aim at finding the minimum norm of the perturbation terms $\mathbf{\Delta}_C$ or $\mathbf{\Delta}_{C_w}$ that are compatible with the passivity constraints. This condition however does not ensure that the energy (squared $\mathscr{L}^2$-norm) of the transfer matrix perturbation is minimized. To this end, the minimum of

$$\|\mathbf{\Delta}_\mathbf{S}\|_{\mathscr{L}^2}^2 = \frac{1}{2\pi}\int_{-\infty}^{\infty} \mathrm{tr}\{\mathbf{\Delta}_\mathbf{S}(\jmath\omega)\mathbf{\Delta}_\mathbf{S}^H(\jmath\omega)\}d\omega \tag{2.85}$$

should be found. However, it is well known [126] that this norm can be characterized as

$$\|\mathbf{\Delta}_\mathbf{S}\|_{\mathscr{L}^2}^2 = \mathrm{tr}\{\mathbf{\Delta}_C\mathbf{P}_C\mathbf{\Delta}_C^T\} \tag{2.86}$$

where $\mathbf{P}_C$ is the controllability Gramian associated to (2.43), found as the unique symmetric and positive definite solution of the Lyapunov equation

$$\mathbf{A}\mathbf{P}_C + \mathbf{P}_C\mathbf{A}^T = -\mathbf{B}\mathbf{B}^T\,. \tag{2.87}$$

Computing the Cholesky factorization $\mathbf{P}_C = \mathbf{Q}_C^T\mathbf{Q}_C$ and defining

$$\mathbf{\Xi} = \mathbf{\Delta}_C\mathbf{Q}_C^T\,, \quad \boldsymbol{\xi} = \mathrm{vec}(\mathbf{\Xi}) = (\mathbf{Q}_C \otimes \mathbf{I}_P)\,\mathrm{vec}(\mathbf{\Delta}_C)\,, \tag{2.88}$$

it follows

$$\|\mathbf{\Delta_S}\|^2_{\mathscr{L}^2} = \mathrm{tr}\{\mathbf{\Xi}\mathbf{\Xi}^T\} = \|\boldsymbol{\xi}\|_2^2\,. \tag{2.89}$$

Therefore, problem (2.77) can be cast as a minimum $\mathscr{L}^2$-norm formulation by performing the change of variable (2.88), obtaining

$$\begin{cases} \min\theta \\ \|\boldsymbol{\xi}\|_2^2 < \theta \\ \mathbf{\Gamma}\boldsymbol{\xi} \geqslant \boldsymbol{b} \end{cases} \tag{2.90}$$

where $\mathbf{\Gamma} = \mathbf{T}(\mathbf{Q}_C^{-1} \otimes \mathbf{I}_P)$.

Apply now the same procedure to (2.83). The controllability Gramian associated to the compressed state-space realization (2.42) reads

$$\mathbf{A}_w\mathbf{P}_{C_w} + \mathbf{P}_{C_w}\mathbf{A}_w^T = -\mathbf{B}_w\mathbf{B}_w^T\,, \tag{2.91}$$

together with its Cholesky factorization $\mathbf{P}_{C_w} = \mathbf{Q}_{C_w}^T\mathbf{Q}_{C_w}$. Note that the numerical solution of (2.42) requires only $\mathcal{O}(N_w)$ operations due to the sparse (diagonal or tridiagonal) realization of $\boldsymbol{w}(s)^T$. This cost is negligible with respect to all other macromodeling steps in the proposed framework. Defining

$$\begin{aligned} \mathbf{\Xi}_w &= \mathbf{\Delta}_{C_w}\mathbf{Q}_{C_w}^T\,, \\ \boldsymbol{\xi}_w &= \mathrm{vec}(\mathbf{\Xi}_w) = (\mathbf{Q}_{C_w} \otimes \mathbf{I}_\rho)\,\mathrm{vec}(\mathbf{\Delta}_{C_w})\,, \end{aligned} \tag{2.92}$$

and denoting as $\mathbf{\Delta}_{\boldsymbol{w}^T}(s)$ the induced perturbation on the compressed macromodel, it follows

$$\|\mathbf{\Delta}_{\boldsymbol{w}^T}\|^2_{\mathscr{L}^2} = \|\boldsymbol{\xi}_w\|_2^2\,, \tag{2.93}$$

so that substitution into (2.83) leads to

$$\begin{cases} \min\theta \\ \|\boldsymbol{\xi}_w\|_2^2 < \theta \\ \mathbf{\Gamma}_w\boldsymbol{\xi}_w \geqslant \boldsymbol{b} \end{cases} \tag{2.94}$$

where $\mathbf{\Gamma}_w = \mathbf{T}_w(\mathbf{Q}_{C_w}^{-1} \otimes \mathbf{I}_\rho)$. The solution of (2.94) thus provides the minimum $\mathscr{L}^2$-norm perturbation of the compressed macromodel $\boldsymbol{w}^T(s)$.

It follows that

Lemma 1. *Define* $\mathbf{P}_C$ *and* $\mathbf{P}_{C_w}$ *as in* (2.87) *and* (2.91). *Then*

$$\mathbf{P}_C = \mathbf{I}_P \otimes \mathbf{P}_{C_w} . \tag{2.95}$$

Proof. Suppose that $\mathbf{P}_{C_w}$ is the solution of (2.91), then $\mathbf{P}_C$ defined in (2.95) is a solution of (2.87) by direct substitution. Using (2.43),

$$\begin{aligned}
&\mathbf{A}\mathbf{P}_C + \mathbf{P}_C\mathbf{A}^T \\
&= (\mathbf{I}_P \otimes \mathbf{A}_w)(\mathbf{I}_P \otimes \mathbf{P}_{C_w}) + (\mathbf{I}_P \otimes \mathbf{P}_{C_w})(\mathbf{I}_P \otimes \mathbf{A}_w^T) \\
&= \mathbf{I}_P \otimes (\mathbf{A}_w\mathbf{P}_{C_w} + \mathbf{P}_{C_w}\mathbf{A}_w^T) \\
&= \mathbf{I}_P \otimes (-\mathbf{B}_w\mathbf{B}_w^T) \\
&= -(\mathbf{I}_P \otimes \mathbf{B}_w)(\mathbf{I}_P \otimes \mathbf{B}_w^T) \\
&= -\mathbf{B}\mathbf{B}^T .
\end{aligned}$$

Since both $\mathbf{A}$ and $\mathbf{A}_w$ are strictly negative definite, $\mathbf{P}_C$ and $\mathbf{P}_{C_w}$ are the unique solutions of Lyapunov equations (2.87) and (2.91), which implies (2.95). □

It is now possible to state an important result.

Theorem 1. *Defining the compressed macromodel perturbation*

$$\boldsymbol{\Delta}_{\boldsymbol{w}^T} \leftrightarrow \left(\begin{array}{c|c} \mathbf{A}_w & \mathbf{B}_w \\ \hline \boldsymbol{\Delta}_{C_w} & \mathbf{0} \end{array}\right) \tag{2.96}$$

and the corresponding global macromodel perturbation

$$\boldsymbol{\Delta}_{\mathbf{S}} \leftrightarrow \left(\begin{array}{c|c} \mathbf{A} & \mathbf{B} \\ \hline \boldsymbol{\Delta}_C & \mathbf{0} \end{array}\right) , \tag{2.97}$$

with state-space matrices constructed as in (2.43), *if follows*

$$\|\boldsymbol{\Delta}_{\mathbf{S}}\|^2_{\mathscr{L}^2} = \|\boldsymbol{\Delta}_{\boldsymbol{w}^T}\|^2_{\mathscr{L}^2} \tag{2.98}$$

Proof. As a preliminary result, consider matrix $\bar{\mathbf{V}}$ in (2.30). Using (2.45), the orthogonality condition $\bar{\mathbf{V}}^T\bar{\mathbf{V}} = \mathbf{I}$ can be rewritten in terms of its constituent blocks $\bar{\mathbf{V}}_i$ as

$$\sum_{i=1}^{P}\sum_{m=1}^{P}(\bar{\mathbf{V}}_i)_{m\ell}(\bar{\mathbf{V}}_i)_{mn} = \delta_{n\ell} , \quad n, \ell = 1, \ldots, \rho , \tag{2.99}$$

where $\delta_{n\ell} = 1$ if $n = \ell$ and 0 otherwise. Considering now (2.78) and using (2.95), a straightforward algebraic manipulation leads to

$$\boldsymbol{\Delta}_C\mathbf{P}_C\boldsymbol{\Delta}_C^T = \sum_{i=1}^{P}\bar{\mathbf{V}}_i\boldsymbol{\Upsilon}_w\bar{\mathbf{V}}_i^T , \tag{2.100}$$

where $\boldsymbol{\Upsilon}_w = \boldsymbol{\Delta}_{C_w} \mathbf{P}_{C_w} \boldsymbol{\Delta}_{C_w}^T$. The $\mathscr{L}^2$-norm of the global macromodel perturbation is characterized as

$$\begin{aligned}
\|\boldsymbol{\Delta}_{\mathbf{S}}\|_{\mathscr{L}^2}^2 &= \mathrm{tr}\{\boldsymbol{\Delta}_C \mathbf{P}_C \boldsymbol{\Delta}_C^T\} \\
&= \sum_{m=1}^{P} \left(\sum_{i=1}^{P} \bar{\mathbf{V}}_i \boldsymbol{\Upsilon}_w \bar{\mathbf{V}}_i^T \right)_{mm} \\
&= \sum_{m=1}^{P} \sum_{i=1}^{P} \sum_{n=1}^{\rho} \sum_{\ell=1}^{\rho} (\bar{\mathbf{V}}_i)_{m\ell} (\boldsymbol{\Upsilon}_w)_{\ell n} (\bar{\mathbf{V}}_i)_{mn} \\
&= \sum_{n=1}^{\rho} \sum_{\ell=1}^{\rho} (\boldsymbol{\Upsilon}_w)_{\ell n} \sum_{m=1}^{P} \sum_{i=1}^{P} (\bar{\mathbf{V}}_i)_{m\ell} (\bar{\mathbf{V}}_i)_{mn} \\
&= \sum_{\ell=1}^{\rho} (\boldsymbol{\Upsilon}_w)_{\ell\ell} \\
&= \|\boldsymbol{\Delta}_{\boldsymbol{w}^T}\|_{\mathscr{L}^2}^2 ,
\end{aligned}$$

which completes the proof. □

The practical relevance of this theorem is that the solution of the small-size optimization problem (2.94), in addition to providing the minimum-energy perturbation of the compressed macromodel, will also provide as a by-product the minimum-energy solution of the full-size passivity enforcement problem, which is the main objective. Global passivity enforcement is thus achieved with optimal accuracy and negligible cost through (2.94).

2.4.3 Passivity enforcement examples

In this subsection, the performance of the passivity enforcement schemes (2.90) and (2.94) are compared for each of the benchmark cases of Table 2.1. The results are summarized in Table 2.4, where the total execution time and number of iterations for both schemes are grouped in columns 2 and 3 for convenience. It can be seen that the number of iterations for the compressed scheme is practically always less than for the full scheme. This implies that, independent on the runtime required for a single iteration, the compressed scheme performs generally better. This consideration should be taken into account when interpreting the total runtime, reported in the second column. Note that a dramatic reduction is achieved by the compressed scheme, which is able to complete the passivity enforcement also for those large cases (12, 14, and 18) for which the full scheme requires excessive memory.

Two different speedup factors are reported in the fourth column of Table 2.4. The first is the overall speedup factor, obtained as the ratio of the

Table 2.4: Comparison of full and compressed passivity enforcement schemes in terms of number of iterations ♯ it, runtime, and accuracy $\|\delta\mathbf{X}\|_2$. Last two columns report the overall speedup (SU) and the speedup per iteration (SUit).

Case	Full / Compressed			SU	SUit
	♯ it	Time [s]	$\|\delta\mathbf{X}\|_2$		
1	6 / 7	2.42 / 1.52	0.22 / 0.26	1.6	1.8
2	2 / 1	9.63 / 1.85	0.22 / 0.11	5.2	2.6
3	12 / 7	255 / 3.87	2.61 / 2.61	66.1	38.5
4	2 / 1	3.7 / 0.36	0.04 / 0.04	10.2	5.1
5	12 / 9	687.5 / 22.6	0.16 / 0.21	30.4	22.8
6	50 / 30	324.3 / 12.1	0.53 / 0.41	26.8	16.1
7	2 / 2	1.45 / 0.36	0.05 / 0.06	4.1	4.1
8	28 / 10	510 / 15.9	1.43 / 1.26	32.1	11.4
9	2 / 26	5.83 / 1.64	4.15 / 4.21	3.5	3.1
10	9 / 8	3865 / 145	3.31 / 3.32	26.6	23.6
11	2 / 4	9.34 / 1.81	0.04 / 0.05	5.2	10.4
12	- / 32	-.- / 17344	-.- / 1.21	-.-	-.-
13	8 / 7	24.7 / 4.86	0.16 / 0.21	5.1	4.2
14	- / 13	-.- / 5049	-.- / 1.21	-.-	-.-
15	1 / 2	5.85 / 3.17	0.08 / 0.08	1.8	3.6
16	10 / 8	13.1 / 1.95	0.21 / 0.25	6.8	5.4
17	10 / 6	13.7 / 1.26	0.51 / 0.51	11.4	6.8
18	- / 5	-.- / 1621	-.- / 6.79	-.-	-.-

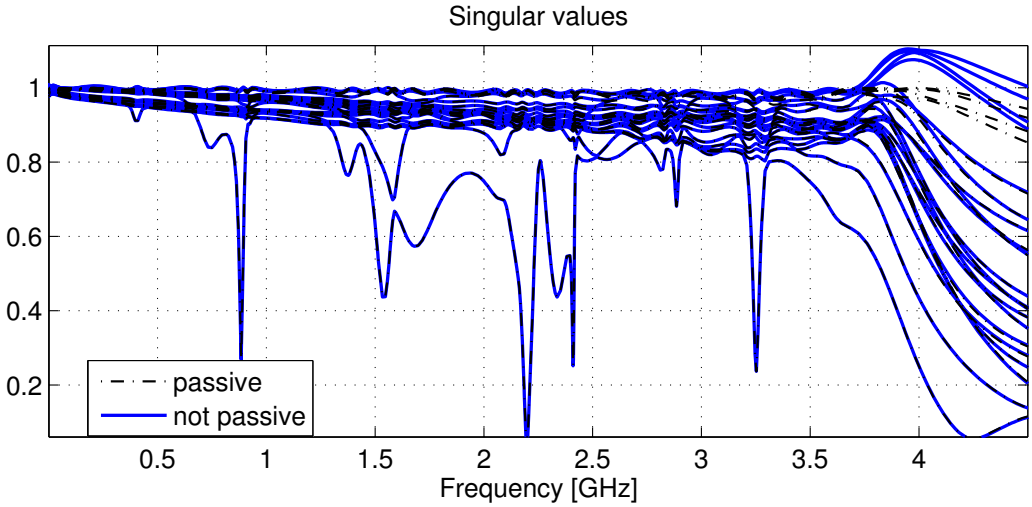

Figure 2.7: Singular value plot before and after passivity enforcement for case 17.

total runtime required by the full and compressed schemes. The second is the average runtime per iteration, which provides a more precise metric for assessing the enhancement in efficiency that can be achieved with proposed approach. In any case, both speedup per iteration and overall speedup are between 1 and 2 orders of magnitude for the most challenging cases, except for the largest cases for which only the compressed scheme could achieve its goal.

Finally, the last column of Table 2.4 reports the deviation of the obtained passive models with respect to the original raw data, showing that the accuracies of both full and compressed schemes are comparable. Figure 2.7 reports as an example the singular value plot for case 17, showing all singular values before and after compressed passivity enforcement. As expected, the singular values of the passive model are uniformly unitary bounded.

2.4.4 A summary of numerical results

The main results for all benchmark cases are now summarized. Table 2.5 provides a detailed report on the accuracy of all intermediate steps of the proposed compressed passive macromodeling approach. The second column reports the thresholds ϵ_{SVD} and ϵ_{VF} that were used, respectively, to bound the approximation error for SVD truncation and compressed VF. Note that these thresholds are used to bound the spectral 2-norm of error matrices $\|\delta\mathbf{X}\|_2$ collecting all responses at all frequencies. Since the relationship of

Table 2.5: Accuracy with respect to raw data of compressed data ($\delta\mathbf{X}_{\text{SVD}}$) and compressed macromodel before ($\delta\mathbf{X}_{\text{VF}}$) and after ($\delta\mathbf{X}_{\text{PAS}}$) passivity enforcement.

Case	ϵ	$\delta\mathbf{X}_{\text{SVD}}$	$\delta\mathbf{X}_{\text{VF}}$	$\delta\mathbf{X}_{\text{PAS}}$
	SVD / VF	$\|\cdot\|_2$ / $\|\cdot\|_{\max}$	$\|\cdot\|_2$ / $\|\cdot\|_{\max}$	$\|\cdot\|_2$ / $\|\cdot\|_{\max}$
1	0.1 / 0.1	0.07 / 0.0039	0.09 / 0.006	0.26 / 0.014
2	0.1 / 0.1	0.06 / 0.0045	0.09 / 0.007	0.11 / 0.007
3	0.1 / 0.1	0.06 / 0.0029	0.08 / 0.003	2.61 / 0.064
4	0.1 / 0.1	0.04 / 0.0015	0.04 / 0.002	0.04 / 0.002
5	0.1 / 0.1	0.06 / 0.0105	0.09 / 0.051	0.23 / 0.057
6	0.1 / 0.1	0.07 / 0.0041	0.09 / 0.006	0.42 / 0.015
7	0.1 / 0.1	0.01 / 0.0005	0.04 / 0.001	0.06 / 0.002
8	0.1 / 0.5	0.08 / 0.0027	0.48 / 0.016	1.05 / 0.014
9	0.1 / 0.1	0.05 / 0.0084	0.05 / 0.008	4.12 / 0.632
10	0.1 / 3.0	0.07 / 0.0061	2.21 / 0.048	2.53 / 0.048
11	0.1 / 0.1	0.01 / 0.0012	0.01 / 0.001	0.18 / 0.016
12	0.1 / 0.1	0.02 / 0.0002	0.05 / 0.001	1.22 / 0.011
13	0.1 / 0.1	0.04 / 0.0046	0.05 / 0.011	0.21 / 0.011
14	0.1 / 0.1	0.07 / 0.0213	0.08 / 0.021	1.26 / 0.031
15	0.1 / 0.1	0.06 / 0.0018	0.08 / 0.002	0.08 / 0.002
16	0.1 / 0.1	0.04 / 0.0147	0.08 / 0.015	0.25 / 0.015
17	0.1 / 0.4	0.07 / 0.0241	0.39 / 0.315	0.43 / 0.315
18	0.1 / 6.8	0.07 / 0.0055	6.79 / 0.212	6.91 / 0.218

these thresholds to the actual deviation that is achieved at a given frequency for a given response is not obvious, the results in terms of the worst-case norm, defined as

$$\|\delta\mathbf{X}\|_{\max} = \max_{\ell k} |(\delta\mathbf{X})_{\ell k}| \ . \tag{2.101}$$

are also reported.

The last three columns of Table 2.5 report the spectral and worst-case accuracies (with respect to raw data) of compressed data $\delta\mathbf{X}_{\mathrm{SVD}}$, compressed fitted model $\delta\mathbf{X}_{\mathrm{VF}}$, and final model after compressed passivity enforcement $\delta\mathbf{X}_{\mathrm{PAS}}$, respectively. The table clearly shows that accuracy is well preserved through all modeling steps. For illustration, in Figures 2.8 and 2.9, respectively, the responses characterized by the worst-case absolute error can be found for case 17, and the responses characterized by the worst-case relative error for case 2. Similar results were obtained for all other cases and are not reported here.

2.5 Parallel passivity check

In order to use the passivity enforcement scheme previously introduced in Section 2.4.1, passivity violations of the state-space model

$$\mathbf{H}(s) = \mathbf{D} + \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B} \tag{2.102}$$

must be properly identified. While causality and stability are guaranteed by the unique condition that all model poles should have negative real part, passivity is more difficult to guarantee since a special set of constraints are necessary according to Section 2.3.4. It is important to note that condition (2.51) must be checked for each frequency $\omega \in \mathbb{R}$. The first idea is then to use a frequency sampling process to extract a significant set of frequency points ω_l and to check condition (2.51) on these samples only. Of course for the sake of reliability the set of samples ω_l must be determined adaptively according to the dynamic features of the macromodel. As a consequence, dealing with models having large port count and high dynamical order makes this strategy computationally expensive. Therefore the main objective of this section is to introduce an highly efficient parallel implementation of the available adaptive sampling scheme proposed in [104]. The core idea of the adaptive sampling scheme is presented in the rest of this section and in Section 2.5.1, while the new parallel implementation strategy is detailed in Sections 2.5.2, 2.5.3 and 2.5.4.

The main objective of the proposed Parallel Adaptive Sampling (PAS) scheme is to determine a partition of the frequency axis $\Omega = [0, \infty)$ into

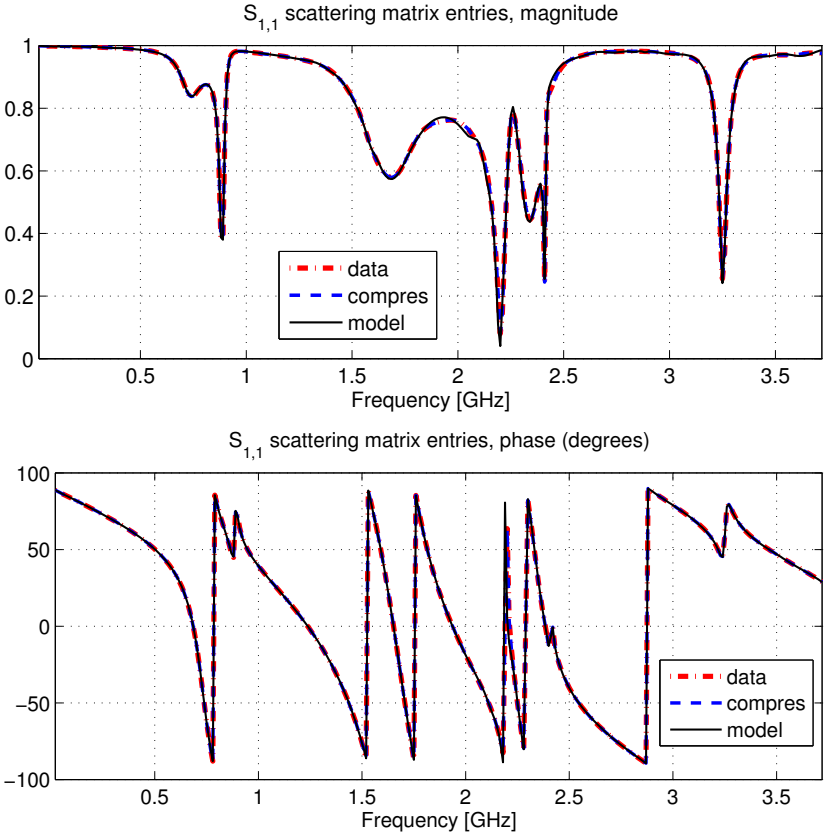

Figure 2.8: A scattering response of a PCB interconnect (case 17) before (red dashed line) and after (blue dashed line) compression. The black line represents the response of the passive compressed macromodel.

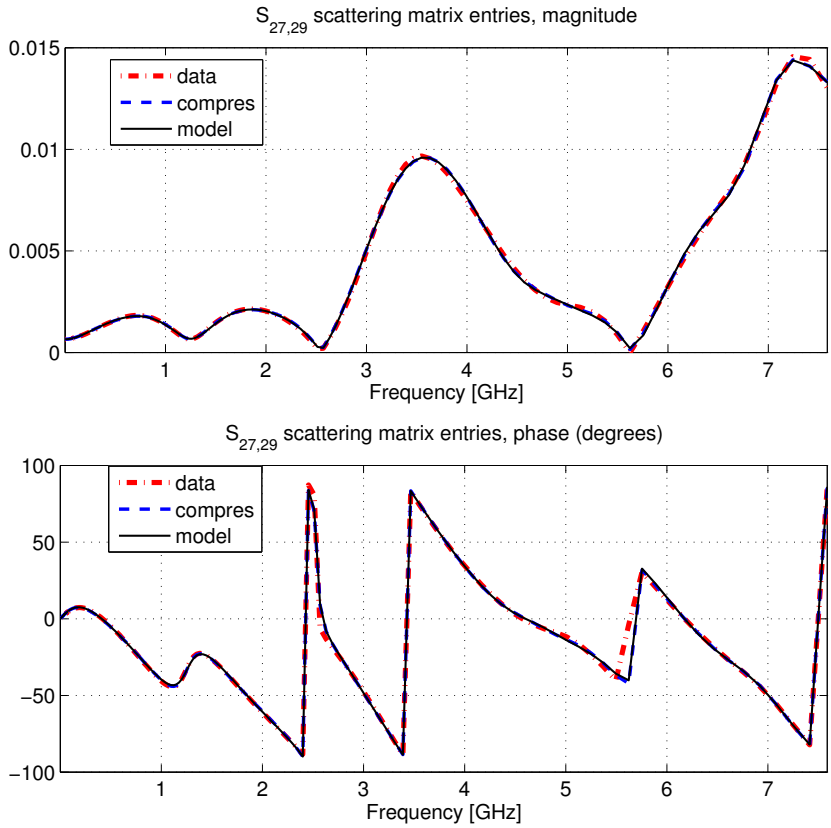

Figure 2.9: As in Figure 2.8, but for a high-speed connector (case 2).

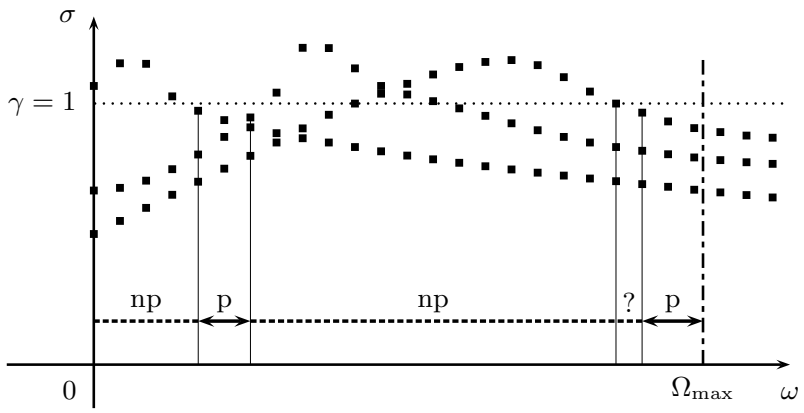

Figure 2.10: Graphical illustration passivity characterization via frequency sampling. See text for details.

disjoint sub-bands

$$\Omega = \bigcup_{q=1}^{Q} \Omega_q \,, \qquad \Omega_q = [\omega_{q-1}, \omega_q) \tag{2.103}$$

with $\omega_0 = 0$ and $\omega_Q = +\infty$. Defining the interior of each sub-band as

$$\check{\Omega}_q = (\omega_{q-1}, \omega_q) = \Omega_q - \{\omega_{q-1}\} \,, \tag{2.104}$$

the partition (2.103) is determined such that one of the following conditions will hold for each sub-band $\check{\Omega}_q$

- $\max_i \sigma_i(\jmath\omega) > 1$, $\forall \omega \in \check{\Omega}_q$: in this case, passivity condition (2.49) is violated at any point within the sub-band, which is thus flagged as "non-passive" with the superscript $^{\mathrm{np}}$.
- $\max_i \sigma_i(\jmath\omega) < 1$, $\forall \omega \in \check{\Omega}_q$: in this case, (2.49) holds at any point within the sub-band, which is thus flagged as "passive" with the superscript $^{\mathrm{p}}$.
- $\max_i \sigma_i(\jmath\omega) \approx 1$, $\forall \omega \in \check{\Omega}_q$: in this case, the maximum singular value will be too close to the threshold $\gamma = 1$ in order to qualify the system as locally passive or non-passive in $\check{\Omega}_q$. It should be guaranteed that this last case is such that $|\Omega_q| = \omega_q - \omega_{q-1}$ is small. This undetermined case will be flagged with the superscript $^{?}$.

Passive, non-passive, and undetermined bands will be collected as

$$\begin{array}{rcll} \Omega^{\mathrm{np}} & = & \bigcup_q \Omega_q : & \max_i \sigma_i(\jmath\omega) > 1, \, \forall \omega \in \check{\Omega}_q \\ \Omega^{\mathrm{p}} & = & \bigcup_q \Omega_q : & \max_i \sigma_i(\jmath\omega) < 1, \, \forall \omega \in \check{\Omega}_q \\ \Omega^{?} & = & \bigcup_q \Omega_q : & \Omega_q \not\subseteq \Omega^{\mathrm{np}} \cup \Omega^{\mathrm{p}} \end{array} \tag{2.105}$$

In addition, for each non-passive sub-band $\Omega_q \subseteq \Omega^{\mathrm{np}}$, all local maxima $\hat{\sigma}_l$ and the corresponding frequencies $\hat{\omega}_l$ at which these maxima are attained are needed. See Figure 2.10 for a graphical illustration.

2.5.1 Accuracy-controlled sampling via eigenvector tracking

Recall that, when Scattering models are used the state-space matrix $\mathbf{A}$ has no purely imaginary poles, as a consequence of VF implementation [117, 62], thus the singular values $\sigma_i(\jmath\omega)$ are continuous and differentiable functions of frequency [125]. However, when computing these singular values numerically over a prescribed discrete set of frequencies $\{\omega_l\}$, there is no guarantee that

each $\sigma_i(\jmath\omega_l)$ for fixed i collects samples from the *same* singular value trajectory. The computation at each frequency ω_l is in fact independent, and the adopted singular value or eigenvalue solver may return its results with an order that may differ from one sample to the next.

The first objective is thus to dynamically determine a set of frequencies $\{\omega_l\}$ that is sufficient to track the individual smooth singular value trajectories by a suitable reordering. This reordering can be achieved by a mode tracking scheme [127], such as the one presented in [104]. Given two available (adjacent) frequency samples ω_m and ω_{m+1}, the eigendecomposition of $\mathbf{\Theta}(\jmath\omega_m)$ and $\mathbf{\Theta}(\jmath\omega_{m+1})$ is computed, then the eigenvalues are collected into matrices $\mathbf{\Lambda}_m$ and $\mathbf{\Lambda}_{m+1}$ while the (orthogonal and unit-normalized) eigenvectors are stored into matrices $\mathbf{V}_m$ and $\mathbf{V}_{m+1}$. Note that these matrices coincide with the right singular vectors of $\mathbf{S}(\jmath\omega)$. Then, all possible mutual scalar products among all these eigenvectors are computed as

$$\widetilde{\boldsymbol{p}}_{m,m+1} = \mathbf{V}_m^H \mathbf{V}_{m+1} \,. \tag{2.106}$$

If the two frequencies are sufficiently close so that the direction of the eigenvectors undergoes a small change from ω_m to ω_{m+1}, then $\widetilde{\boldsymbol{p}}_{m,m+1}$ will have approximately the structure of a permutation matrix, with one single element per row and column with magnitude close to 1, and with all other elements nearly 0. If this is true, the permutation matrix $\boldsymbol{p}_{m,m+1}$ that reorders the eigenvectors and eigenvalues from sample m to sample $m+1$ is obtained by rounding the magnitude of each element of $\widetilde{\boldsymbol{p}}_{m,m+1}$ towards 0 or 1. A numerical test whether this tracking/permutation is successful can be obtained by checking

$$\max_{i,i'} \left\{ \left| \left(|\boldsymbol{p}_{m,m+1}^T \widetilde{\boldsymbol{p}}_{m,m+1}| - \mathbf{I} \right) \right|_{i,i'} \right\} < \varepsilon \tag{2.107}$$

for a suitable threshold $\varepsilon \ll 1$. Refer to [104] for more details. If condition (2.107) is fulfilled, it follows that the behaviour of the system transfer function and its singular values is well resolved within $[\omega_m, \omega_{m+1}]$. Otherwise, a new sample $\omega_{m+1/2} = (\omega_m + \omega_{m+1})/2$ is added and the check is applied again to the two subintervals $[\omega_m, \omega_{m+1/2}]$ and $[\omega_{m+1/2}, \omega_{m+1}]$. Binary subdivision of each pair of adjacent samples drawn from an initial distribution is applied recursively until (2.107) is met everywhere.

2.5.2 Parallel Adaptive Sampling

Consider in more details the above described adaptive refinement scheme. Formally, the refinement check is expressed as

$$\nu = \mathcal{R}(\omega_m, \omega_{m+1}) \,, \tag{2.108}$$

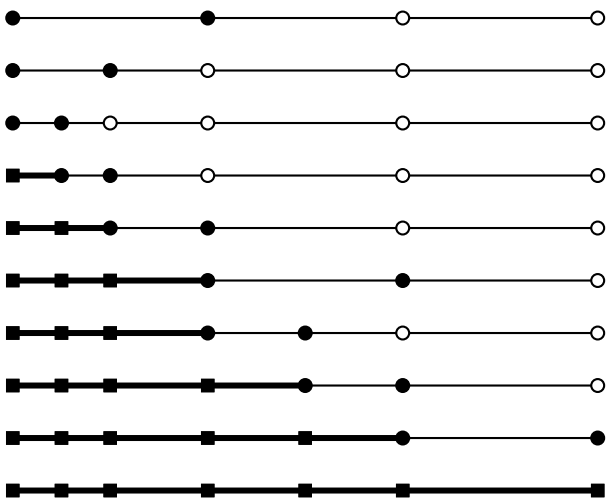

Figure 2.11: Adaptive frequency sampling via local refinement (serial implementation). Each row from top to bottom corresponds to one application of the $\mathcal{R}$ check (2.108). White dots denote samples still to be processed. Black dots denote samples being used by current $\mathcal{R}$ check. Black squares denote samples that do not need any more processing. A thick line highlights a frequency band that is finalized and which does not need further refinement.

where the input arguments define the local band to be checked, and the output ν can be either $\omega_{m+1/2}$ or the empty set $\emptyset$, in which case no further refinement is required. Evaluation of (2.108) requires the computation of transfer matrix $\mathbf{S}(\jmath\omega)$ at the two frequencies ω_m, ω_{m+1}, together with its right singular vector matrices $\mathbf{V}_m$ and $\mathbf{V}_{m+1}$. As part of the $\mathcal{R}$ check, the following computations are included: if ν is empty, the resulting permutation matrix $\mathbf{P}_{m,m+1}$ is immediately applied to reorder the singular values at ω_{m+1}; otherwise, the new sample $\omega_{m+1/2}$ is computed together with its associated transfer matrix $\mathbf{S}(\jmath\omega_{m+1/2})$ and singular vector matrix $\mathbf{V}_{m+1/2}$, which are stored for the next check.

Iterative application of (2.108) determines a binary subdivision tree of the frequency axis, where each node in the tree denotes a frequency sample. Figure 2.11 illustrates the order in which the $\mathcal{R}$ check is applied in a serial implementation, where it is assumed that the leftmost local sub-band that is still to be refined is processed first. Figure 2.11 shows that the sub-bands are finalized starting from the left edge of the initial frequency interval. This consideration leads to a simple strategy for the parallelization of this refinement scheme using T concurrent threads, based on the following steps and rules.

Startup

At startup, a set of initial frequency samples $\mathcal{S}^0$ is determined. Here, this set is constructed as the union of samples obtained independently through different strategies:

- an upper frequency $\Omega_{\max}$ is determined following the procedure in [102], with the guarantee that no passivity violations occur for $\omega > \Omega_{\max}$; therefore, only the interval $[0, \Omega_{\max}]$ needs to be checked instead of the full imaginary axis;
- a set $\mathcal{S}_{\text{lin}}$ of l_{lin} uniformly spaced samples are determined in $[0, \Omega_{\max}]$, including edges;
- a set $\mathcal{S}_{\log}$ of logarithmically spaced samples with l_d samples per decade are computed from $\omega_{\min}$ to $\omega_{\max}$, where l_d, $\omega_{\min}$ and $\omega_{\max}$ depend on the particular application and structure of interest;
- a set $\mathcal{S}_p$ of samples is obtained as in [104] from the model poles $p_i = \alpha_i \pm \jmath\beta_i$ by sampling uniformly with $2R+1$ points the phase of the associated resonance curve, as

$$\mathcal{S}_p = \bigcup_{i,r} \left\{ \omega_{i,r} = \beta_i + \alpha_i \tan \frac{r\pi}{2(R+1)} \right\} \tag{2.109}$$

  with $r = -R, \ldots, R$.

As a result, the set of initial samples that will be subject to the $\mathcal{R}$ iteration is defined as

$$\mathcal{S}^0 = \mathcal{S}_{\text{lin}} \cup \mathcal{S}_{\log} \cup \mathcal{S}_p \,, \tag{2.110}$$

with all samples reordered for increasing values.

Initial workload allocation

Supposing that T concurrent threads are available, the set of initial samples is partitioned as

$$\mathcal{S}^0 = \bigcup_{t=1}^{T} \mathcal{S}_t^0 \,, \tag{2.111}$$

where the number of elements of each subset is $\#\{\mathcal{S}_t^0\} = \lfloor \#\{\mathcal{S}^0\}/T \rfloor$ for $t = 0, \ldots, T-1$. The remaining samples are assigned to $\mathcal{S}_T^0$. The subdivision is ordered, such that for $t_1 < t_2$,

$$\forall \omega_i \in \mathcal{S}_{t_1}^0 \quad \text{and} \quad \forall \omega_j \in \mathcal{S}_{t_2}^0 \quad \Rightarrow \quad \omega_i \leq \omega_j \,, \tag{2.112}$$

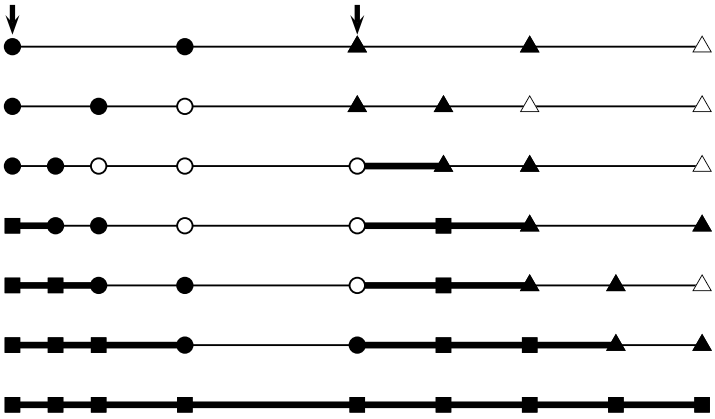

Figure 2.12: Parallel adaptive frequency sampling via local refinement using $T = 2$ threads. Samples assigned to thread $t = 1$ ($t = 2$) are depicted with circles (triangles). Arrows indicate start points (leftmost sample) for the two threads. White fill denotes samples still to be processed, whereas black fill denotes samples used by current iteration. Black squares denote samples that do not need any more processing. A thick line highlights a frequency band that is finalized and which does not need further refinement.

with each pair of adjacent sub-bands $\mathcal{S}_{t_i}^0$ and $\mathcal{S}_{t_{i+1}}^0$ sharing the single sample

$$\tilde{\omega}_i = \max \mathcal{S}_{t_i}^0 = \min \mathcal{S}_{t_{i+1}}^0 \,. \tag{2.113}$$

Each subset $\mathcal{S}_t^0$ is allocated statically to thread t, which iteratively applies the $\mathcal{R}$ refinement check until the entire sub-band is covered, as in Figure 2.11. This initial allocation ensures that, if no refinement is required, approximately the same amount of work is allocated for each thread. Figure 2.12 illustrates this process, showing the evolution of each subset of samples $\mathcal{S}_t^\nu$ at few iterations ν. In the following, the iteration count ν will be dropped.

Dynamic thread reallocation

As the iterative refinement check proceeds and each sub-band is processed independently by each thread, it may happen that some bands require more adaptive refinement steps than others. Therefore, it may happen that one thread t_j completes its refinement task when the other threads are still working. In this case, the thread should not be left inactive, since this would compromise parallel efficiency. In order to find some work to do for the idle thread t_j, the remaining threads t_i are scanned for $i \neq j$ and the number of sample pairs in set $\mathcal{S}_{t_i}$ that at current iteration are still to be processed is

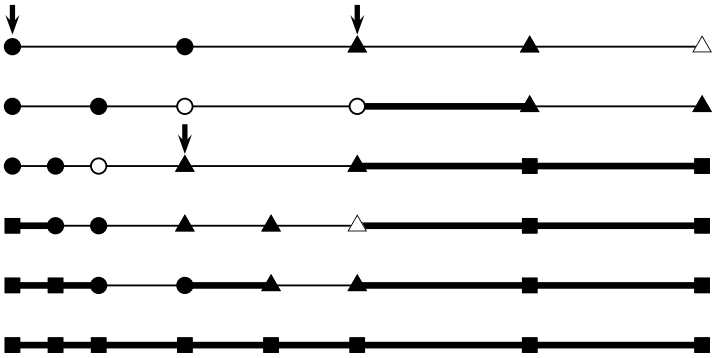

Figure 2.13: Parallel adaptive frequency sampling via local refinement using $T = 2$ threads and dynamic rescheduling (same notation as in Figure 2.12). Note that thread $t = 2$ is restarted at the third iteration after completing its initially assigned workload.

found by the $\mathcal{R}$ check. Although it is not guaranteed that the work for these threads will coincide with the corresponding number of unchecked sub-bands, the number of expected $\mathcal{R}$ iterations will not certainly be smaller. Therefore, the thread t_ℓ that requires the largest amount of estimated $\mathcal{R}$ checks is identified and thread t_j is restarted by assigning to it one half of the samples still to be processed by t_ℓ. More precisely

$$\mathcal{S}_\ell \rightarrow \hat{\mathcal{S}}_{t_\ell} \cup \hat{\mathcal{S}}_{t_j} \tag{2.114}$$

is divided with the constraint

$$\forall \omega_i \in \hat{\mathcal{S}}_{t_\ell} \quad \text{and} \quad \forall \omega_l \in \hat{\mathcal{S}}_{t_j} \quad \Rightarrow \quad \omega_i \leq \omega_l \,, \tag{2.115}$$

with the two sets $\hat{\mathcal{S}}_{t_\ell}$, $\hat{\mathcal{S}}_{t_j}$ sharing only one sample. This strategy guarantees an initially equal subdivision of the workload between t_j and t_ℓ. Figure 2.13 provides a graphical illustration of this thread reallocation. Then, the thread reallocation process is repeated any time some thread becomes idle, by rescheduling it to help the most busy thread at that time.

End of refinement pass

The above described multi-thread adaptive refinement process stops when all threads have completed their tasks. Due to the proposed optimized dynamic scheduling, the algorithm is automatically load balanced, except for the last iteration during which a group of threads might remain idle while the other threads are completing their last task. The maximum total duration of this last step is the time required for a single $\mathcal{R}$ iteration.

In addition to the natural stopping condition for the $\mathcal{R}$ iteration, which occurs when $\emptyset$ is returned by (2.108) and in which case all singular value trajectories are tracked based on their singular vector perturbation, an additional stopping condition is added in terms of the maximum number of nested refinements $I_{\max}$. This parameter intervenes when tracking is not possible, e.g., in the case of singular values with higher multiplicity, whose singular vectors cannot be defined uniquely. In all numerical tests in this paper $I_{\max} = 6$ was used, providing a good compromise between accuracy and efficiency.

2.5.3 Local passivity check

The final result of the above refinement scheme is a set of frequency samples ω_l and a reordered sequence (through the above-defined permutation matrices $\boldsymbol{p}_{m,m+1}$) of singular values samples. For fixed i, the reordered samples $\sigma_i(\jmath\omega_l)$ can thus be considered to be drawn from a continuous and differentiable trajectory $\sigma_i(\jmath\omega)$. Exploitation of this smoothness leads to various straightforward ways of checking passivity between each pair of adjacent frequencies. One can define a worst-case linear prediction error at sample ω_m based on a first-order eigenvalue perturbation from the adjacent left and right samples [104]

$$\Delta_m^{\pm} = \max_i \left\{ \left| \left(\boldsymbol{v}_{m\pm1}^H \boldsymbol{\Theta}_m \boldsymbol{v}_{m\pm1} \right)_{ii} - (\boldsymbol{\Lambda}_m)_{ii} \right| \right\} , \tag{2.116}$$

and infer that the model is locally passive in a neighbourhood of ω_m if

$$\max_i \sigma_i(\jmath\omega_m) + \beta \max\{\Delta_m^-, \Delta_m^+\} < 1 , \tag{2.117}$$

where $\beta > 1$ is a parameter used to compensate for the missing higher order terms in the linear prediction. This local check at ω_m can be formally expressed as

$$\vartheta_m = \mathcal{C}(\omega_{m-1}, \omega_m, \omega_{m+1}) , \tag{2.118}$$

where ϑ_m is either 0 (flagging locally non-passive samples) or 1 (locally passive samples), since a symmetric check is performed using both samples at the left and right of current sample. The only exception is when the check is performed at the edge of the bandwidth of interest, in which case only two samples are used to construct a one-sided linear prediction error Δ_m^- or Δ_m^+.

Performing this local passivity check using T computational threads is straightforward, since a direct static scheduling is sufficient. In fact, since the $\mathcal{C}$ check is performed on a prescribed set of samples which remains fixed

and does not grow through iterations, the static work allocation discussed in Sec 2.5.2 is already optimal. Therefore, this aspect is not further discussed.

As a result from above procedure, the model is concluded to be passive in (ω_m, ω_{m+1}) if (2.117) is satisfied at both ω_m and ω_{m+1}. Conversely, the model is concluded to be non-passive in (ω_m, ω_{m+1}), or at least in some portion of it, if any of the maximum singular values at sample m and $m+1$ is larger than one,

$$\max_i \sigma_i(\jmath\omega_m) > 1 \quad \text{or} \quad \max_i \sigma_i(\jmath\omega_{m+1}) > 1\,. \tag{2.119}$$

For all other cases in which

$$\max_i \sigma_i(\jmath\omega_m) \leq 1 \quad \text{and} \quad \max_i \sigma_i(\jmath\omega_{m+1}) \leq 1\,, \tag{2.120}$$

but (2.117) is not satisfied at ω_m and ω_{m+1}, the sub-band is flagged as undetermined since the singular value trajectories are too close to the threshold.

Once all sub-bands are flagged, adjacent passive (non-passive or undetermined) bands are merged to form the subdivision (2.103). Finally, the local maxima $(\hat{\omega}_l, \hat{\sigma}_l)$ of the singular value trajectories for each non-passive sub-band are determined by constructing a local quadratic polynomial that interpolates three adjacent samples and by taking its peak value. All these operations require negligible time and are performed as a serial post processing in the actual implementation.

2.5.4 Optimizations

The local passivity check $\mathcal{C}$ as described above is performed after the adaptive refinement iteration $\mathcal{R}$ is completed. This strategy presents some critical aspects related to memory use and management. In fact, the $\mathcal{C}$ check requires to store, for each sample ω_m to be checked, the matrix $\mathbf{\Theta}(\jmath\omega_m)$, the eigenvalue matrix $\mathbf{\Lambda}_m$, and the eigenvector matrices at the left and right samples $\mathbf{V}_{m\pm1}$. As a consequence, until a sub-band (ω_m, ω_{m+1}) is definitely flagged as passive/non-passive/undetermined, all the above quantities need to be stored for each of the two samples m, $m+1$. For a $P \times P$ transfer function resulting into a number L of final frequency samples, the overall storage requirement scales as $\mathcal{O}(2P^2L)$. For instance, a 100-port structure with 10000 frequency samples requires more than 1.6 GB of storage using complex double-precision arithmetic.

This large storage requirement can be relaxed and significantly reduced with a modified scheduling approach that interleaves the application of $\mathcal{R}$ and $\mathcal{C}$ iterations. In fact, after each sub-band (ω_m, ω_{m+1}) is flagged after running the $\mathcal{C}$ check at both its endpoints, only the P eigenvalues along the

Table 2.6: Peak memory usage during parallel adaptive sampling and local passivity check for a test case ($L = 4392, P = 56$) with (M_2) and without (M_1) memory optimization. Results are shown for different number of threads T.

T	M_1, MB	M_2, MB
1	442	21
2	446	24
3	451	28
4	455	34
5	461	32
6	471	39
7	480	41
8	491	50

diagonal of $\mathbf{\Lambda}_m$ need to be stored for the final identification of local singular value maxima. The idea is then, during the $\mathcal{R}$ refinement loop, to

- apply a $\mathcal{C}$ check whenever a triplet of adjacent samples $(\omega_{m-1}, \omega_m, \omega_{m+1})$ is finalized by the $\mathcal{R}$ check;
- flag sub-band (ω_m, ω_{m+1}) as soon as both samples are processed by a $\mathcal{C}$ check;
- free the memory from data that is not required by later $\mathcal{R}$ or $\mathcal{C}$ checks, and reuse it to store new samples data, as required by local refinement.

The actual implementation does not free or allocate any memory during the main refinement loop, since this would dramatically impact performance (memory management operations require exclusive access to resources and are not thread-safe). A preallocated pool (buffer) of elementary memory cells is used, whose dimension is based on some heuristic criterion depending on the number of concurrent threads T. These cells are reused by suitable linking through pointer reassignment. If the preallocated memory pool is full, then another block is allocated at once, thus limiting impact on parallel performance. Table 2.6 illustrates the memory savings obtained for a significant test case. Note that this memory optimization is achieved with no loss of performance or parallel efficiency.

Table 2.7: Test cases A-I: L, P and N denote the number of frequency samples in the raw data, the number of ports and the dynamic order of the obtained model, respectively.

Case	L	P	N
A	511	18	4572
B	4096	36	4968
C	2000	36	8064
D	2043	18	2952
E	4096	18	3600
F	145	35	700
G	990	155	10540
H	282	164	6888
I	348	172	5504

2.5.5 Parallel passivity check results

The performance of the proposed passivity check scheme is discussed in this section. From the test cases listed in Table 2.1 the most relevant examples are selected for this section, i.e. cases 5 and 6. Cases A-J are high order models, whose details can be found in Table 2.7, specifically selected to test and challenge the proposed algorithm. Those test cases are very challenging in term of execution time, therefore are very good benchmarks for the parallelization strategy proposed in Section 2.5. The first set of results in Table 2.8 reports the number of frequency samples required by a continuous smooth tracking of the singular values/vectors. The set of initial samples $\mathcal{S}^0$ was generated using the guidelines of Section 2.5.2, with $l_{\text{lin}} = 300$ linearly spaced samples, $l_d = 4$ samples per decade over 9 decades of frequency, and $2R + 1 = 7$ samples per pole. Since this number of initial samples is quite limited, it is expected that the PAS scheme will add many samples in order to track unambiguously the singular value trajectories. This is confirmed by the number of final samples $\#\{\mathcal{S}^{\text{end}}\}$ reported in Table 2.8, which is always in the order of several thousands. Figure 2.14 reports few selected singular value trajectories for case 5 within a restricted frequency band, showing how the final set of samples is able to resolve all fine variations of the curves, which are sampled too coarsely by the initial sample distribution.

The passivity violations detected by the PAS scheme are depicted in Figure 2.15 while Table 2.9 reports the timing results and the parallel speedup for $T = 8$ and $T = 16$ concurrent threads obtained the proposed PAS scheme,

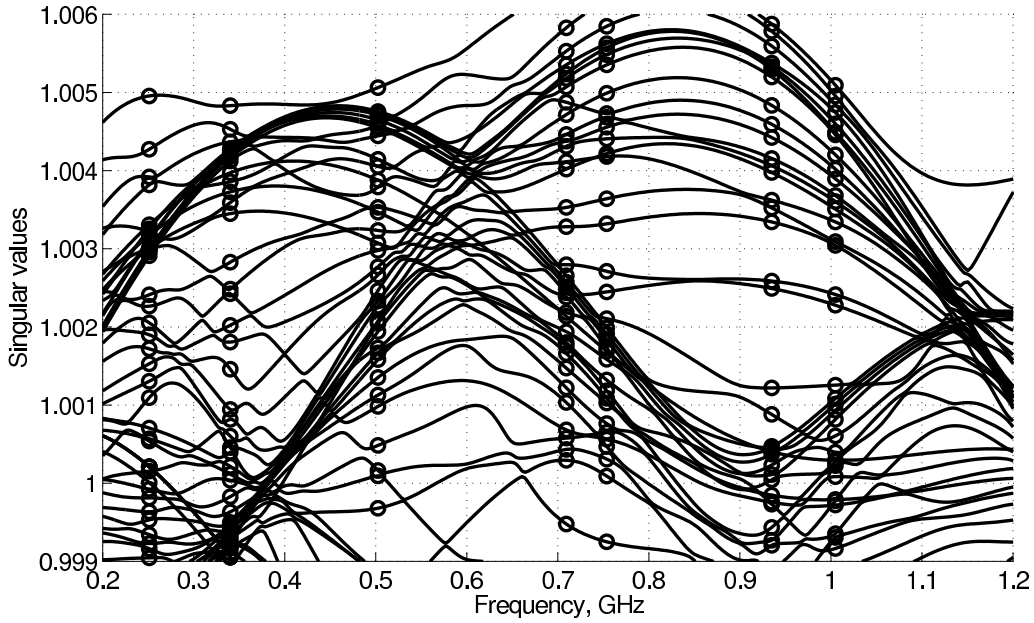

Figure 2.14: Starting (circles) and final tracked frequency samples of few selected singular values for case 5.

Table 2.8: Passivity check: number of initial $\#\{\mathcal{S}^0\}$ and final $\#\{\mathcal{S}^{\text{end}}\}$ frequency samples obtained by the proposed adaptive frequency sampling scheme.

Case	$\#\{\mathcal{S}^0\}$	$\#\{\mathcal{S}^{\text{end}}\}$
5	376	5229
6	451	3129
A	1187	13216
B	766	6568
C	1093	16049
D	873	6932
E	1007	10112
F	348	1969
G	558	12712
H	467	11229
I	429	10128

Table 2.9: Timing results for the Parallel Adaptive Sampling and local passivity check scheme for $T = 1, 8$ and 16 threads, with corresponding speedup factors.

Case	τ_1, s	τ_8, s	τ_{16}, s
5	112.76	14.31 (7.88×)	7.11 (15.87×)
6	5.69	0.73 (7.80×)	0.37 (15.39×)
A	444.01	59.02 (7.52×)	30.90 (14.37×)
B	4088.17	533.90 (7.66×)	277.28 (14.74×)
C	104.72	13.89 (7.54×)	7.35 (14.25×)
D	373.64	49.65 (7.53×)	25.66 (14.56×)
E	628.17	82.56 (7.61×)	41.82 (15.02×)
F	1.39	0.19 (7.15×)	0.11 (13.21×)
G	601.55	77.83 (7.73×)	40.54 (14.84×)
H	569.44	72.77 (7.83×)	36.66 (15.53×)
I	374.20	48.10 (7.78×)	24.05 (15.56×)

inclusive of both adaptive sampling refinement and local passivity check. It can be seen that the scalability of this passivity check scheme with the number of cores is excellent, with a speedup superior to 15× in almost all cases.

Finally the average speedup on several test cases is depicted in Figure 2.16.

2.6 Conclusions

In this Chapter, a comprehensive framework for compressed passive macromodeling of large-scale interconnect structures was presented. The main enabling factor for this new approach is the observation that the whole set of P^2 scattering responses of P-port large-scale systems can be expressed through a much lower-dimensional set of $\rho \ll P$ basis functions. A singular value truncation is able to determine both the number of such basis functions and the corresponding expansion coefficients, with full control over the approximation error.

The above compressed data representation was used to derive reduced-complexity Vector Fitting and passivity enforcement schemes. The former generates a rational macromodel for the set of basis functions. The latter enforces global passivity constraints using a restricted set of perturbation variables and relying on a robust and efficient parallel implementation of

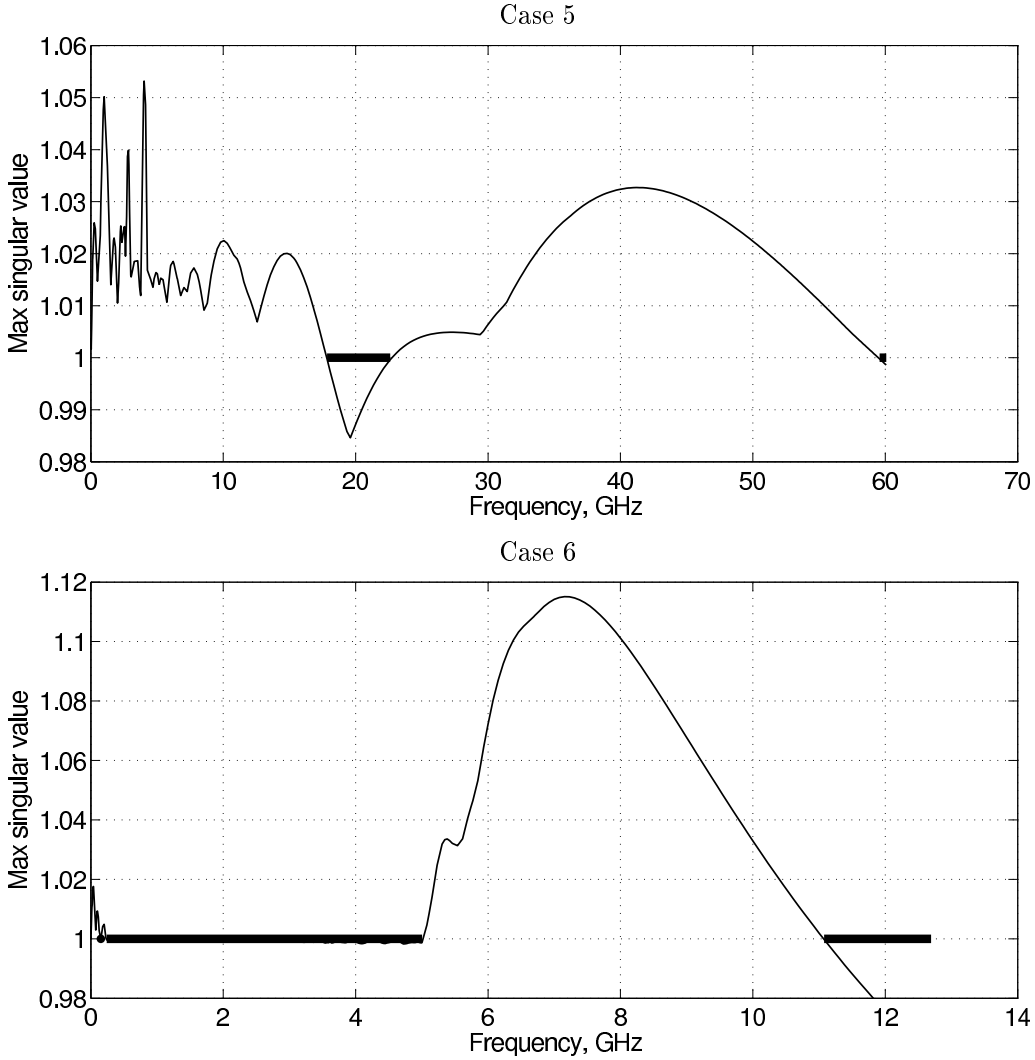

Figure 2.15: Maximum singular value (thin line) and frequency bands Ω_q^{p} that are flagged as passive after the adaptive sampling check (thick line). Top panel: case 5; bottom panel: case 6.

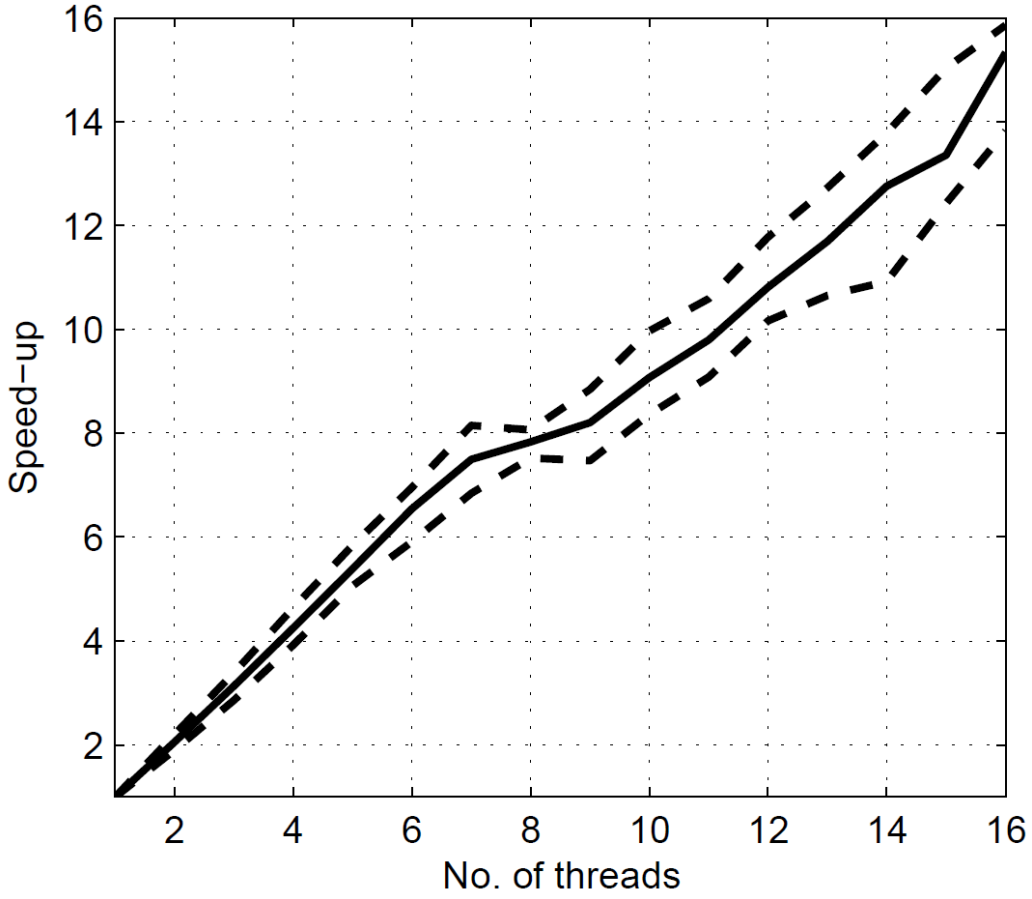

Figure 2.16: The plots report the ratio τ_T/τ_1 versus the number T of computational threads for the best and worst cases (dashed lines), and the average (solid lines) among all analysed benchmarks.

the passivity check algorithm. The overall result is a passive macromodeling scheme that has the potential to outperform state-of-the-art methods in terms of scalability, memory occupation, and CPU requirements, as illustrated through several challenging benchmark cases.

Chapter 3

Small-signal and parameterized macromodels

According to the design flow described in Section 1.3 once a prototype design is available, extensive numerical simulations are required using suitable models for all Circuit Blocks (CBs), in order to verify the proper functioning of the entire system under realistic operating conditions. It is clear that the adoption of full transistor-level models for such verifications is not viable due to excessive overall complexity. In several situations, however, the dynamic behaviour of individual CB's can be approximated by suitable reduced-order behavioural macromodels. This is in fact true for those devices, such as Low Noise Amplifiers (LNA), Operational Amplifiers, Low Dropout regulator (LDO), or programmable filters, that operate almost linearly when suitably biased around a specified operating points. Once validated against the full transistor-level netlist models, such behavioural equivalents offer an excellent solution for drastically reducing the overall runtime of system-level simulations.

In this Chapter, a new parameterized behavioural modelling approach is described that is able to: i) reliably compute a reduced order small-signal macromodel of linearized CB; ii) enforce the DC response of the reduced equivalent to match exactly the DC response of the original CB; iii) include in the macromodel's coefficients a closed-form parameterization in terms of both biasing conditions, e.g. the nominal V_{dd} applied to the CB, and even additional design or operation parameters, e.g., the temperature. The approach presented here builds on existing parameterized macromodeling approaches that are available in the literature [112, 115, 128, 82, 129, 130, 131, 132, 133, 134]. The focus here is to show what modifications are needed in these approaches in order to guarantee at the same time a good parameterization and full DC compliance.

For the sake of clarity, Section 3.1 is dedicated to the presentation of the DC correction strategy being such a methodology of interest by itself when applied to Linear Transfer Function Models (LTFMs); the extension to the parameterized case is detailed in Section 3.2 together with applications to real designs.

3.1 DC-corrected small-signal models

Non-linear and causal systems, for which the wavelength associated to the operating frequency is much larger than the circuits physical dimensions, can be modelled via finite-order non-linear state space equations [135]

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}(t)) \tag{3.1}$$

$$\boldsymbol{y}(t) = \boldsymbol{g}(\boldsymbol{x}(t), \boldsymbol{u}(t)) \tag{3.2}$$

where $\boldsymbol{u}(t), \boldsymbol{y}(t) \in \mathbb{R}^P$ denote system inputs and outputs, $\boldsymbol{x}(t) \in \mathbb{R}^N$ is an internal state vector, and $\dot{\boldsymbol{x}}(t) = \frac{d\boldsymbol{x}(t)}{dt}$.

When (3.1)-(3.2) represent a non-linear circuit block for AMS and RF applications, like LNA's (Low Noise Amplifiers), OPA's (Operational Amplifiers) and programmable active filters, a significant complexity reduction of these non-linear state equations is possible. In fact, since these devices are designed to operate almost linearly when driven below maximum allowed input power or signal magnitude, the input, output and state vectors can be represented as a superposition of a constant DC term $(\boldsymbol{u}_{DC}, \boldsymbol{x}_{DC}, \boldsymbol{y}_{DC})$ on all the ports and a small-signal time dependent term $(\tilde{\boldsymbol{u}}(t), \tilde{\boldsymbol{x}}(t), \tilde{\boldsymbol{y}}(t))$ as

$$\boldsymbol{u}(t) = \boldsymbol{u}_{DC} + \tilde{\boldsymbol{u}}(t)\,, \tag{3.3}$$

$$\boldsymbol{x}(t) = \boldsymbol{x}_{DC} + \tilde{\boldsymbol{x}}(t)\,, \tag{3.4}$$

$$\boldsymbol{y}(t) = \boldsymbol{y}_{DC} + \tilde{\boldsymbol{y}}(t)\,, \tag{3.5}$$

where vectors $\boldsymbol{u}_{DC}, \boldsymbol{x}_{DC}, \boldsymbol{y}_{DC}$ collect the constant DC contributions. If only constant inputs are applied (DC conditions), it follows

$$\boldsymbol{u}(t) = \boldsymbol{u}_{DC} \quad \text{and} \quad \dot{\boldsymbol{x}}(t) = \boldsymbol{0}\,, \tag{3.6}$$

which applied to (3.1) and (3.2) leads to the definition of the DC operation point as the solution of

$$\boldsymbol{f}(\boldsymbol{x}_{DC}, \boldsymbol{u}_{DC}) = \boldsymbol{0}\,, \tag{3.7}$$

$$\boldsymbol{y}_{DC} = \boldsymbol{g}(\boldsymbol{x}_{DC}, \boldsymbol{u}_{DC})\,. \tag{3.8}$$

The triplet $\boldsymbol{u}_{DC}, \boldsymbol{x}_{DC}, \boldsymbol{y}_{DC}$ is available from a direct DC simulation of the transistor-level circuit block.

Using (3.3)-(3.5) into (3.1)-(3.2) leads to

$$\dot{\tilde{\boldsymbol{x}}}(t) = \boldsymbol{f}(\boldsymbol{x}_{DC} + \tilde{\boldsymbol{x}}(t), \boldsymbol{u}_{DC} + \tilde{\boldsymbol{u}}(t)) \,, \tag{3.9}$$

$$\tilde{\boldsymbol{y}}(t) + \boldsymbol{y}_{DC} = \boldsymbol{g}(\boldsymbol{x}_{DC} + \tilde{\boldsymbol{x}}(t), \boldsymbol{u}_{DC} + \tilde{\boldsymbol{u}}(t)) \,, \tag{3.10}$$

which, under small-signal excitation, can be approximated by a first-order Taylor expansion of both state and output equations

$$\dot{\tilde{\boldsymbol{x}}}(t) \approx \mathbf{A}\tilde{\boldsymbol{x}}(t) + \mathbf{B}\tilde{\boldsymbol{u}}(t) \,, \tag{3.11}$$

$$\tilde{\boldsymbol{y}}(t) \approx \mathbf{C}\tilde{\boldsymbol{x}}(t) + \mathbf{D}\tilde{\boldsymbol{u}}(t) \,, \tag{3.12}$$

where $\mathbf{A} \in \mathbb{R}^{N\times N}$, $\mathbf{B} \in \mathbb{R}^{N\times P}$, $\mathbf{C} \in \mathbb{R}^{P\times N}$ and $\mathbf{D} \in \mathbb{R}^{P\times P}$ denote constant state-space matrices defining the small-signal Linear Transfer Function Model (LTFM) of the Circuit Block (CB) around the specified bias conditions, with frequency-dependent input-output response

$$\mathbf{H}(s) = \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B} + \mathbf{D}. \tag{3.13}$$

The elements of these state matrices are formally defined as partial derivatives of the various components of (3.1)-(3.2) evaluated at the current DC point. However, as discussed in [136], it is also possible to obtain the LTFM by first extracting a set of frequency-dependent small-signal Scattering $\mathbf{S}_l$, Admittance $\mathbf{Y}_l$ or Impedance $\mathbf{Z}_l$ parameters, in the following collectively denoted as $\mathbf{H}_l$ with $l = 1, \ldots, L$, by exploiting standard features of state of the art circuit solvers, namely a set of small-signal AC analyses. Then, this data is fed to a macromodeling algorithm, e.g. Vector Fitting [62], to directly obtain the reduced-order macromodel (3.11)-(3.12) by minimizing the macromodel error $\|\mathbf{H}(\jmath\omega_l) - \mathbf{H}_l\|$ in the desired norm.

The LTFM usually attains a very good accuracy for the small-signal characterization of the CB in the frequency domain [136]. Unfortunately, similar good results can not be obtained from time domain (transient) simulation. In fact, a direct replacement of the non-linear CB with the LFTM in a transient simulation setup leads to possibly incorrect biasing, since the small-signal macromodel does not include any information of the underlying DC operation point. When excited by constant inputs $\boldsymbol{u}(t) = \boldsymbol{u}_{DC}$, the LTFM provides its closed form DC output solution

$$\check{\boldsymbol{y}}_{DC} = \mathbf{H}(0)\boldsymbol{u}_{DC} = (\mathbf{D} - \mathbf{C}\mathbf{A}^{-1}\mathbf{B})\boldsymbol{u}_{DC}, \tag{3.14}$$

which has no relationship with the true DC operation point of the original CB. This information is not embedded in the LTFM, which only represents

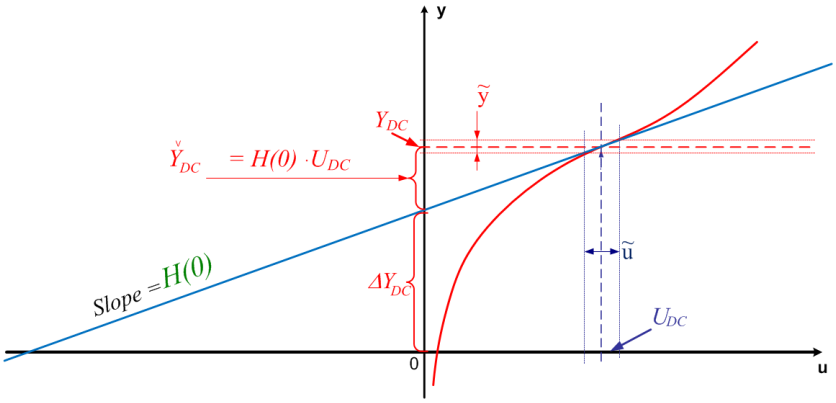

Figure 3.1: Graphical illustration of the DC point correction for a static one-port case. The LTFM (blue line) provides a good (first order) approximation near the operating point of the non-linear characteristic (red curve), but the DC solution of the LTFM $\check{\boldsymbol{y}}_{DC}$ from (3.14) has no relation with the correct DC solution $\boldsymbol{y}_{DC}$.

the dynamics of the small variations around the bias point. This issue is summarized graphically in Figure 3.1.

Considering the case of several CB's modelled as LTFM's and connected together in a long chain to realize a low complexity model of an RF transceiver path, it is clear that the DC solution of all individual simplified models must comply with the exact bias conditions, especially when some non-linear components are still present in the testbench. An example is provided by the system level schematic of a simple receiver stage in Figure 3.2 [137], which shows how a circuit block driven by the outputs of the previous LTFM could receive as an input the wrong DC bias and could therefore be operating incorrectly.

3.1.1 DC correction strategy

To overcome the intrinsic DC-OP accuracy limitation of the LTFM at DC, the following correction strategy can be implemented. Assume that the correct bias conditions provided by the input-output pair $(\boldsymbol{u}_{DC}, \boldsymbol{y}_{DC})$ are known as a solution of (3.8) for the original non-linear system. Then, once the small-signal macromodel (3.11)-(3.12) is available, its closed-form DC solution $\check{\boldsymbol{y}}_{DC}$ driven by the same nominal biasing inputs $\boldsymbol{u}_{DC}$ is computed as in (3.14). Compute the difference

$$\Delta \boldsymbol{y}_{DC} = \boldsymbol{y}_{DC} - \check{\boldsymbol{y}}_{DC}\,, \tag{3.15}$$

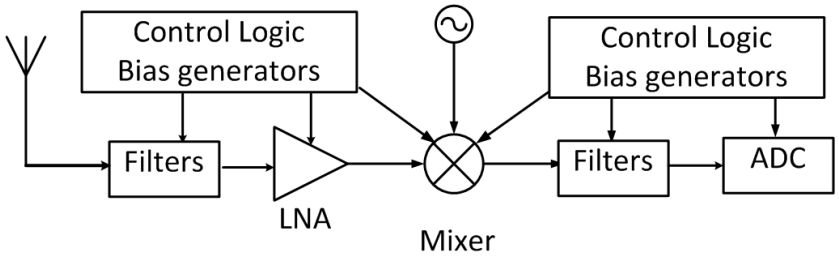

Figure 3.2: Top level schematic of a basic receiver chain [137]. For the simulation of such a CB chain it is essential that each block in the chain biases the following CB correctly. Even a small error in the DC-OP modelling of some CB, like the LNA, will corrupt the performance of the following CB's.

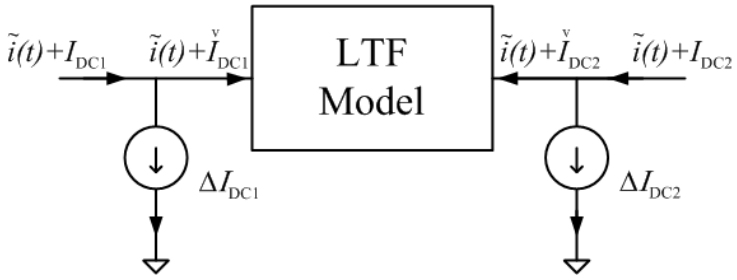

Figure 3.3: DC point correction for a two port LTFM. The correct DC bias is set via constant current sources ΔI_{DC1} and ΔI_{DC2} applied at the input ports of the LTFM. The current source values are provided by the elements of the correction vector (3.15).

which represents the correction that must be applied to the DC solution of the LTFM in order to obtain the nominal CB bias level.

The correction terms $\Delta \boldsymbol{y}_{DC}$ are applied by defining an enlarged DC-corrected small-signal macromodel which embeds the original LTFM and adds at its interface ports suitable constant sources, whose values are the components of $\Delta \boldsymbol{y}_{DC}$. In case the k-th port input $\boldsymbol{u}_k$ is a voltage and the corresponding k-th output $\boldsymbol{y}_k$ is a current, the correction is applied as a shunt current source with value $\Delta \boldsymbol{y}_{DCk}$. Conversely, if $\boldsymbol{u}_k$ is a current and $\boldsymbol{y}_k$ is a voltage, a series constant voltage source $\Delta \boldsymbol{y}_{DCk}$ is applied. The basic idea is depicted in Figure 3.3 for a two-port voltage-controlled device. It should be noted that using constant correction sources will affect and fix the DC point only, without any effect on the accuracy of the LTFM around the OP point under small-signal excitation.

The proposed strategy for the extraction of a low-complexity DC-compliant small-signal linear macromodel can be summarized in the following steps:

1. create a suitable CB characterization test bench and apply there the desired DC operation point setting to each CB pin;

2. extract $\boldsymbol{y}_{DC}$ and the small-signal frequency-dependent $\mathbf{S}_l$, $\mathbf{Y}_l$, or $\mathbf{Z}_l$ parameters from a circuit simulation of the non-linear system, here represented by (3.1)-(3.2);

3. perform a rational curve fitting of the $\mathbf{S}_l$, $\mathbf{Y}_l$, or $\mathbf{Z}_l$ parameters, e.g. using VF [62], and obtain a state-space realization of the LTFM;

4. compute $\check{\boldsymbol{y}}_{DC}$ from (3.14) and $\Delta\boldsymbol{y}_{DC}$ from (3.15);

5. synthesize a circuit netlist using one of the standard macromodel realization described in Chapter 4, complemented by DC correction sources $\Delta\boldsymbol{y}_{DC}$ at its external ports.

3.1.2 Results

This section presents some results to illustrate the effectiveness of the proposed method. The following test cases are considered.

- A two-stage buffer: this is a simple non-linear example whose netlist is depicted in Figure 3.7. The accuracy of the extracted LTFM can be seen in Figure 3.4.

- A Low-Drop Out (LDO) regulator: the corresponding CB is taken from a real 3G transceiver design. This is basically a DC voltage regulator, controlled by external biases and a logic unit. LDO's can operate with a very small input-output differential voltage. The high level schematic of this component is depicted in Figure 3.8. The accuracy of the extracted LTFM can be seen in Figure 3.5.

- A Low Noise Amplifier (LNA): the corresponding CB was also taken from a real 3G transceiver design. LNA's are widely used in receiver chains like the one depicted in Figure 3.2. A high level schematic for the LNA is depicted in Figure 3.9. The accuracy of the extracted LTFM can be seen in Figure 3.6.

For each test case, the relative error between the raw and DC-corrected LTFM responses is considered under constant excitation by the nominal bias inputs. These errors are defined, respectively, as

$$\epsilon_{\check{y}} = \left| \frac{\check{\boldsymbol{y}}_{DC} - \boldsymbol{y}_{DC}}{\boldsymbol{y}_{DC}} \right| , \tag{3.16}$$

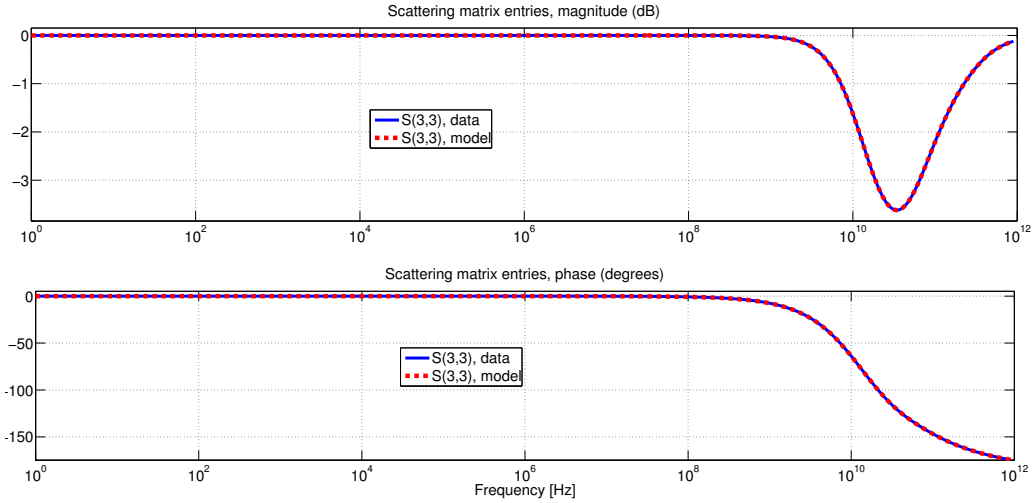

Figure 3.4: $S_{3,3}$ from the small-signal model of the Buffer. The red dashed line is the model response while the blue lines are the data used for the identification.

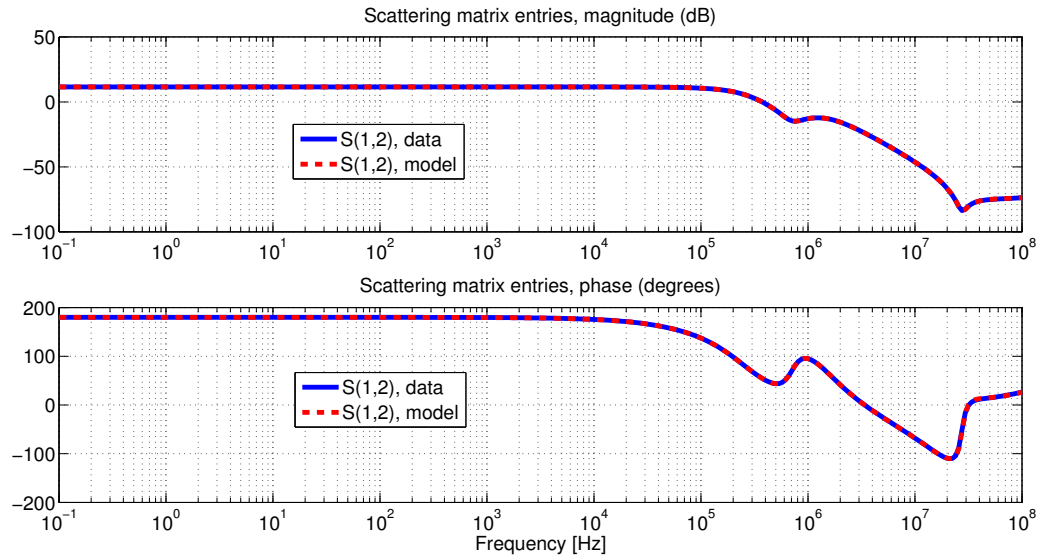

Figure 3.5: $S_{1,2}$ from the small-signal model of the LDO. The red dashed line is the model response while the blue lines are the data used for the identification.

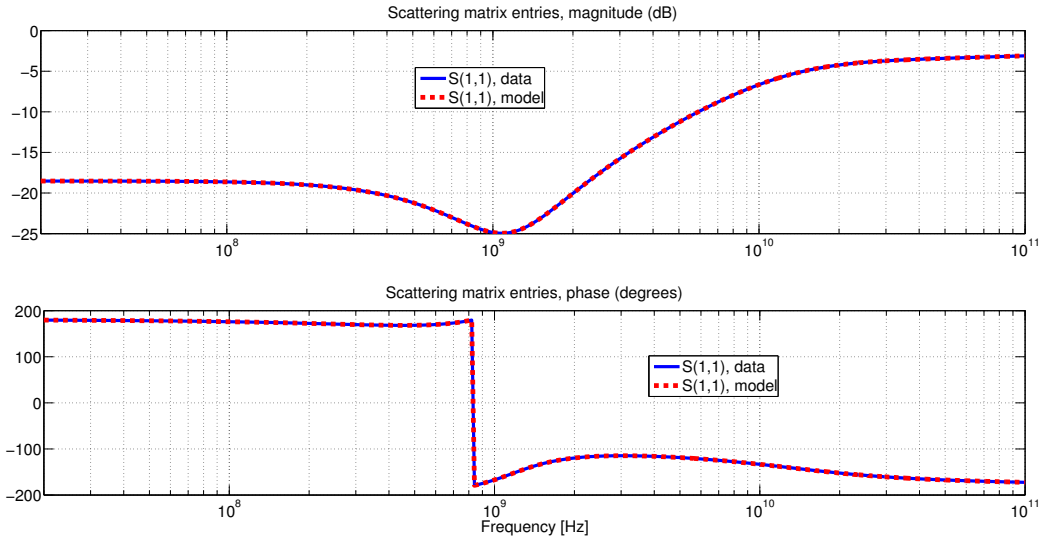

Figure 3.6: $S_{1,1}$ from the small-signal model of the LNA. The red dashed line is the model response while the blue lines are the data used for the identification.

for the raw LTFM, and

$$\epsilon_{\bar{y}} = \left| \frac{\bar{\boldsymbol{y}}_{DC} - \boldsymbol{y}_{DC}}{\boldsymbol{y}_{DC}} \right| \tag{3.17}$$

for the DC-corrected LTFM, where $\bar{\boldsymbol{y}}_{DC}$ represents the DC output obtained from the LTFM after the application of the DC correction sources defined by (3.15).

The results obtained by a circuit simulation of the original CB and synthesized LTFM are reported in Table 3.1, where all DC results for all port

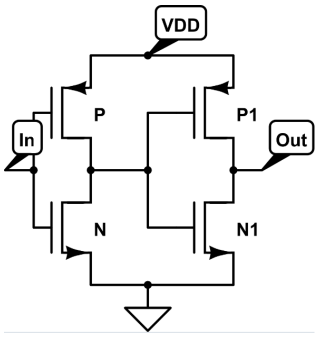

Figure 3.7: A two-stage buffer.

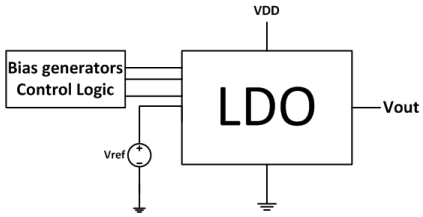

Figure 3.8: High-level schematic of a Low-Drop Out (LDO) regulator CB extracted from a real transceiver block. The Control Logic can be used to select the desired voltage output V_{out}, while V_{ref} and V_{VDD} are reference and supply voltages.

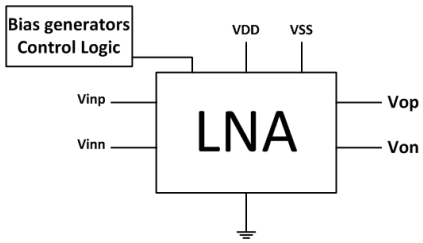

Figure 3.9: High-level schematic of an integrated LNA, which is part of a real receiver chain (Figure 3.2); terminals V_{inp} and V_{inn} define the differential input, while V_{DD} is the supply voltage and V_{op} V_{on} define the differential output pair.

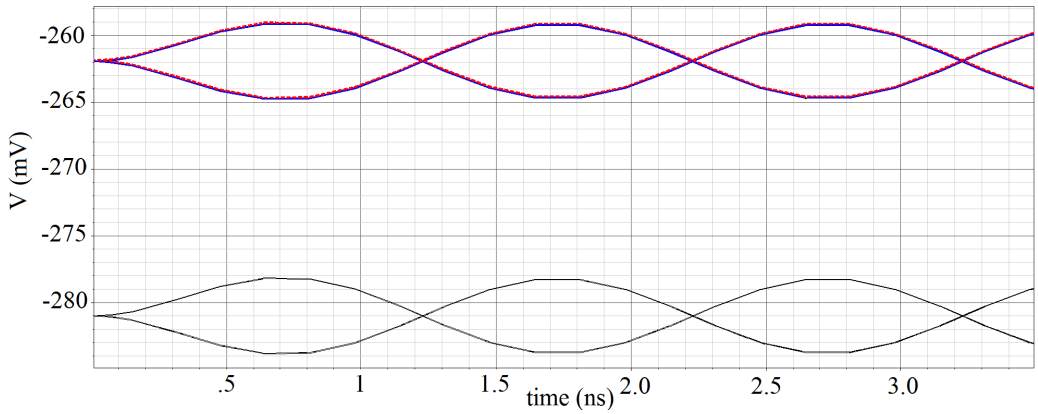

Figure 3.10: Output transient results for the LNA example obtained with the original CB (solid blue line) the raw LTFM (solid black line), and the DC-corrected LTFM (dashed red line). The input signal for the LNA is a simple sine wave having $1mV$ peak to peak amplitude. This simple example clearly demonstrates the effectiveness of the proposed strategy. The transient response obtained using the LTFM (black solid line), is very accurate except for the vertical shift due to its incorrect DC level. The DC-corrected LTFM is completely overlapped to the transient response obtained from the nonlinear CB.

variables are reported, together with the corresponding LTFM relative errors. As seen from this table the DC-corrected LTFM results are exact, as expected, whereas the raw LTFM provides an incorrect DC solution.

In order to further illustrate the advantages of the proposed reduced-order modelling strategy, a transient simulation is performed for the LNA structure using both the original nonlinear CB and the small-signal raw and DC-corrected macromodels. The results are depicted in Figure 3.10. It is clear that the DC-corrected macromodel provides practically coincident results with the reference, whereas the raw LTFM results in a DC shift of its response. Note that the reference simulation took 10 minutes to perform a transient analysis of 500ns, whereas the DC-corrected LTFM simulation only required 5 seconds, with a significant speedup.

3.2 Parameterized small-signal models

In this section the DC correction strategy from Section 3.1.1 is extended to the parameterized case and combined with the state of the art techniques for the identification of parameterized models from a set of raw data.

Table 3.1: Voltage and currents for the test cases in Figure 3.7-3.9. Where $\boldsymbol{y}_{DC}$ are the DC data from the CB under analysis, $\check{\boldsymbol{y}}_{DC}$ are the DC data obtained from the LTFM before the application of the correction strategy and $\bar{\boldsymbol{y}}_{DC}$ are the same data after the application of the DC correction strategy. Error norms $\epsilon_{\check{y}}$ and $\epsilon_{\bar{y}}$ are defined according to (3.16) and (3.17).

Test		$\boldsymbol{y}_{DC}$	$\check{\boldsymbol{y}}_{DC}$	$(\epsilon_{\check{y}})$	$\bar{\boldsymbol{y}}_{DC}$	$(\epsilon_{\bar{y}})$
Buffer	I_{in}	-1.58e-11	0	(1)	-1.58e-11	(0)
	I_{out}	1.55e-3	3.87e-3	(1.5)	1.55e-3	(0)
	I_{DD}	-1.55e-3	-3.87e-3	(1.5)	-1.55e-3	(0)
	V_{out}	1.55e-6	3.87e-6	(1.5)	1.55e-6	(0)
LDO	I_{DD}	-3.39e-4	-1.32e-3	(28)	-3.39e-4	(0)
	I_{ref}	-2.5e-3	-2.6e-3	(0.04)	-2.5e-3	(0)
	I_{out}	3.39e-4	1.32e-3	(28)	3.39e-4	(0)
	V_{out}	1.294	1.295	(0.04)	1.294	(0)
LNA	I_{DD}	-1.81e-3	8.3e-5	(1)	-1.81e-3	(0)
	I_{SS}	-1.85e-3	0.024	(10)	-1.85e-3	(0)
	I_{op}	-5.24e-3	-5.62e-3	(7e-2)	-5.24e-3	(0)
	I_{on}	-5.24e-3	-5.62e-3	(7.2e-2)	-5.24e-3	(0)
	V_{op}	-0.262	-0.28	(6.8e-2)	-0.262	(0)
	V_{on}	-0.262	-0.28	(6.8e-2)	-0.262	(0)

Consider a generic nonlinear and dynamic Circuit Block (CB) represented by the following state-space equations [135]

$$\dot{\boldsymbol{x}}(t;\boldsymbol{\eta}) = \boldsymbol{f}(\boldsymbol{x}(t;\boldsymbol{\eta}),\boldsymbol{u}(t);\boldsymbol{\eta}) \tag{3.18}$$
$$\boldsymbol{y}(t;\boldsymbol{\eta}) = \boldsymbol{g}(\boldsymbol{x}(t;\boldsymbol{\eta}),\boldsymbol{u}(t);\boldsymbol{\eta}) \tag{3.19}$$

where $\boldsymbol{u},\boldsymbol{y} \in \mathbb{R}^P$ denote system inputs and outputs, $\boldsymbol{x} \in \mathbb{R}^N$ is an internal state vector, and $\dot{\boldsymbol{x}}$ indicates the time derivative of the state vector. In (3.18), the vector $\boldsymbol{\eta} \in \mathbb{R}^\nu$ collects the ν physical or design parameters which the circuit block response depends on, that are the main subject of this investigation. Note that both state and output equations may depend on $\boldsymbol{\eta}$, inducing a parameter dependence on their solution. Therefore, both state $\boldsymbol{x}(t;\boldsymbol{\eta})$ and output $\boldsymbol{y}(t;\boldsymbol{\eta})$ vectors are multivariate functions of time t and parameters $\boldsymbol{\eta}$. Assume that inputs are invariant for each geometrical or physical configuration of the system, so that $\boldsymbol{u}(t)$ does not depend on $\boldsymbol{\eta}$.

3.2.1 Linear Transfer Function Models

For AMS and RF applications several circuit blocks such as Low Noise Amplifiers (LNA's) or programmable active filters are designed to operate almost linearly when suitably biased and excited by small-signal inputs within the maximum allowed range of input power. Under these conditions, input, output and state vectors can be represented as a superposition of constant DC terms $\boldsymbol{u}_{DC},\boldsymbol{x}_{DC}(\boldsymbol{\eta}),\boldsymbol{y}_{DC}(\boldsymbol{\eta})$ and small-signal time dependent terms $\tilde{\boldsymbol{u}}(t),\tilde{\boldsymbol{x}}(t;\boldsymbol{\eta}),\tilde{\boldsymbol{y}}(t;\boldsymbol{\eta})$ as

$$\boldsymbol{u}(t) = \boldsymbol{u}_{DC} + \tilde{\boldsymbol{u}}(t), \tag{3.20}$$
$$\boldsymbol{x}(t;\boldsymbol{\eta}) = \boldsymbol{x}_{DC}(\boldsymbol{\eta}) + \tilde{\boldsymbol{x}}(t;\boldsymbol{\eta}), \tag{3.21}$$
$$\boldsymbol{y}(t;\boldsymbol{\eta}) = \boldsymbol{y}_{DC}(\boldsymbol{\eta}) + \tilde{\boldsymbol{y}}(t;\boldsymbol{\eta}). \tag{3.22}$$

If the small-signal input $\tilde{\boldsymbol{u}}(t)$ is switched off and only the constant DC bias is applied, it follows

$$\boldsymbol{u}(t) = \boldsymbol{u}_{DC} \quad \text{and} \quad \dot{\boldsymbol{x}}(t;\boldsymbol{\eta}) = \boldsymbol{0} \tag{3.23}$$

uniformly for each instance of the parameter vector $\boldsymbol{\eta}$. Application to (3.18)-(3.19) leads to the definition of the parameter-dependent DC operating point as the solution of

$$\boldsymbol{f}(\boldsymbol{x}_{DC}(\boldsymbol{\eta}),\boldsymbol{u}_{DC}) = \boldsymbol{0}, \tag{3.24}$$
$$\boldsymbol{y}_{DC}(\boldsymbol{\eta}) = \boldsymbol{g}(\boldsymbol{x}_{DC}(\boldsymbol{\eta}),\boldsymbol{u}_{DC}). \tag{3.25}$$

The triplet $\boldsymbol{u}_{DC}, \boldsymbol{x}_{DC}(\boldsymbol{\eta}), \boldsymbol{y}_{DC}(\boldsymbol{\eta})$ is available from a direct DC sweep of the transistor-level circuit block covering the desired range of variation of the parameter vector.

Conversely, when the small-signal input is switched on, insertion of (3.20)-(3.22) into (3.18)-(3.19) leads to

$$\dot{\tilde{\boldsymbol{x}}}(t;\boldsymbol{\eta}) = \boldsymbol{f}(\boldsymbol{x}_{DC}(\boldsymbol{\eta}) + \tilde{\boldsymbol{x}}(t;\boldsymbol{\eta}), \boldsymbol{u}_{DC} + \tilde{\boldsymbol{u}}(t)), \tag{3.26}$$

$$\tilde{\boldsymbol{y}}(t;\boldsymbol{\eta}) + \boldsymbol{y}_{DC}(\boldsymbol{\eta}) = \boldsymbol{g}(\boldsymbol{x}_{DC}(\boldsymbol{\eta}) + \tilde{\boldsymbol{x}}(t;\boldsymbol{\eta}), \boldsymbol{u}_{DC} + \tilde{\boldsymbol{u}}(t)), \tag{3.27}$$

which can be approximated by a first-order Taylor expansion of both state and output equations as

$$\dot{\tilde{\boldsymbol{x}}}(t;\boldsymbol{\eta}) \approx \mathbf{A}(\boldsymbol{\eta})\tilde{\boldsymbol{x}}(t;\boldsymbol{\eta}) + \mathbf{B}(\boldsymbol{\eta})\tilde{\boldsymbol{u}}(t), \tag{3.28}$$

$$\tilde{\boldsymbol{y}}(t;\boldsymbol{\eta}) \approx \mathbf{C}(\boldsymbol{\eta})\tilde{\boldsymbol{x}}(t;\boldsymbol{\eta}) + \mathbf{D}(\boldsymbol{\eta})\tilde{\boldsymbol{u}}(t), \tag{3.29}$$

where $\mathbf{A}(\boldsymbol{\eta}) \in \mathbb{R}^{N\times N}$, $\mathbf{B}(\boldsymbol{\eta}) \in \mathbb{R}^{N\times P}$, $\mathbf{C}(\boldsymbol{\eta}) \in \mathbb{R}^{P\times N}$ and $\mathbf{D}(\boldsymbol{\eta}) \in \mathbb{R}^{P\times P}$ denote parameter-dependent state-space matrices defining the small-signal Linear Transfer Function Model (LTFM) of the CB around the specified bias conditions, with frequency- and parameter-dependent input-output response

$$\mathbf{H}(s;\boldsymbol{\eta}) = \mathbf{C}(\boldsymbol{\eta})(s\mathbf{I} - \mathbf{A}(\boldsymbol{\eta}))^{-1}\mathbf{B}(\boldsymbol{\eta}) + \mathbf{D}(\boldsymbol{\eta}). \tag{3.30}$$

The elements of these state matrices are formally defined as partial derivatives of the various components of (3.18)-(3.19) evaluated at the current DC point.

3.2.2 Frequency and Time-domain macromodeling

The standard approach for the characterization of the small-signal input-output behaviour of the CB is to extract a set of frequency- and parameter-dependent small-signal Scattering, Admittance or Impedance parameters, through a set of small-signal AC (Alternate Current) analyses. Standard circuit solvers of the SPICE class are able to perform this operation only for discrete values of frequency

$$\omega \in \{\omega_l,\ l = 1, \ldots, L\} \tag{3.31}$$

and parameters

$$\boldsymbol{\eta} \in \{\boldsymbol{\eta}_k,\ k = 1, \ldots, K\}, \tag{3.32}$$

resulting in a set of $P \times P$ complex matrices

$$\mathbf{H}_{l,k} = \mathbf{H}(\jmath\omega_l; \boldsymbol{\eta}_k)\,. \tag{3.33}$$

The computed DC operating points for the state and the output vectors over the parameter grid are denoted as

$$\boldsymbol{x}_{DC,k} = \boldsymbol{x}_{DC}(\boldsymbol{\eta}_k) \tag{3.34}$$

$$\boldsymbol{y}_{DC,k} = \boldsymbol{y}_{DC}(\boldsymbol{\eta}_k)\,. \tag{3.35}$$

The discrete samples (3.33) of the linearised system response provide an excellent approximation of the system behaviour for design and verification purposes, as long as this verification is conducted in the frequency domain and for the available parameter values $\boldsymbol{\eta}_k$. However, if the CB response is required for an arbitrary parameter configuration $\boldsymbol{\eta}_*$ that is not part of the discrete set $\{\boldsymbol{\eta}_k\}$, a new extraction is required by solving the original CB system (3.18)-(3.19). It is clear that for complex CB's and for repeated parameter instances this approach may be overly time-consuming.

If the verification has to be performed in the time-domain, a frequency-to-time conversion is further required. Several macromodeling approaches are available [62, 138, 139] for performing this conversion and obtaining an approximate state-space representation in form of (3.28)-(3.29) or (3.30). This process usually leads to a reduced-order compact system with a number of states $n \ll N$.

Macromodeling approaches are standard for non-parameterized systems. In the proposed setting, for any fixed parameter instance $\boldsymbol{\eta} = \boldsymbol{\eta}_k$, the frequency dependence of the data samples $\mathbf{H}_{l,k}$ is approximated by a rational model, or equivalently a state-space system in form

$$\mathbf{H}_k(s) = \mathbf{C}_k(s\mathbf{I} - \mathbf{A}_k)^{-1}\mathbf{B}_k + \mathbf{D}_k \tag{3.36}$$

by minimizing the macromodel error $\|\mathbf{H}_k(\jmath\omega_l) - \mathbf{H}_{l,k}\|$ in the desired norm. The Vector Fitting (VF) scheme [62] with all its possible variants provides therefore an excellent numerical tool.

The standard VF approach however does not solve the problem of making a compact model available for any desired values of the parameters $\boldsymbol{\eta}$. Fortunately, an explicit treatment of the parameter dependence for the derivation of a parameterized macromodel is also possible, using one of the available parameterized macromodeling strategies [112, 140, 128, 82, 80, 129, 131, 130, 132, 134, 133]. These methods are able to process collectively the samples (3.33) to obtain a multivariate representation of the system as a parameterized reduced-order macromodel in a form identical to (3.30), by minimizing the error $\|\mathbf{H}(\jmath\omega_l;\boldsymbol{\eta}_k) - \mathbf{H}_{l,k}\|$ over the entire set of frequency and parameter samples. A more detailed description of this approach is postponed to Section 3.2.5.

3.2.3 Parameterized macromodeling

The Sanathanan-Koerner algorithm discussed in Section 2.2 is now extended to the parameterized identification problem. Consider the following representation for the parameterized small-signal macromodel

$$\mathbf{H}(s,\boldsymbol{\eta}) = \frac{\mathbf{N}(s,\boldsymbol{\eta})}{d(s,\boldsymbol{\eta})} = \frac{\sum_{m=0}^{M} \mathbf{R}_m(\boldsymbol{\eta})\phi_m(s)}{\sum_{m=0}^{M} r_m(\boldsymbol{\eta})\phi_m(s)} \tag{3.37}$$

where the frequency-dependent basis functions are partial fractions associated to a set of distinct prescribed poles q_m

$$\phi_0(s) = 1, \quad \phi_m(s) = \frac{1}{s - q_m} \tag{3.38}$$

and where the parameter-dependent coefficients are expressed as a superposition of multivariate basis functions $\xi_j(\boldsymbol{\eta})$ as

$$\mathbf{R}_m(\boldsymbol{\eta}) = \sum_{j=1}^{J} \mathbf{R}_{m,j}\xi_j(\boldsymbol{\eta}), \quad r_m(\boldsymbol{\eta}) = \sum_{j=1}^{J} r_{m,j}\xi_j(\boldsymbol{\eta}) \tag{3.39}$$

with constant and unknown coefficients $\mathbf{R}_{m,j}$ and $r_{m,j}$. The representation (3.37) is quite general, since it provides an implicit parameterization of M-th order rational matrices with both parameter-dependent poles and residues [141, 112, 128].

Several choices are possible for the basis functions $\xi_j(\boldsymbol{\eta})$, such as monomials, orthogonal polynomials, or finite elements defined over structured or unstructured grids in the parameter space [112, 140, 128, 82, 80, 129, 130, 131, 132, 134, 133]. In this work standard monomials are used by setting

$$\xi_j(\boldsymbol{\eta}) = \prod_i \eta_i^{\kappa_{j,i}} \tag{3.40}$$

with i spanning the number of free parameters (components of $\boldsymbol{\eta}$), with j interpreted as a global index spanning the set of all multivariate monomials with overall degree $\sum_i \kappa_{j,i} \leq \bar{\kappa}$. The choice of polynomials is justified here by the expected smooth parameter dependence for the structures of interest. This will be confirmed by all examples of Section 3.3. This choice is however not restrictive, since the same procedure can be applied without any modification to different parameterization schemes based on general basis functions $\xi_j(\boldsymbol{\eta})$.

Given the set $\mathbf{H}_{l,k}$ of small-signal transfer matrices available at the frequency points ω_l and parameter grid values $\boldsymbol{\eta}_k$, the coefficients $\mathbf{R}_{m,j}$ and $r_{m,j}$

are computed through a generalized parametric Sanathanan-Koerner (SK) iteration [60, 81], an extension of the algorithm presented in Section 2.2, which minimizes the following cost function

$$\mathcal{E}_{\mu}^{2}=\sum_{l=1}^{L}\sum_{k=1}^{K}\left\|w_{l,k}^{(\mu)}\left[\mathbf{N}^{(\mu)}(\jmath\omega_{l};\boldsymbol{\eta}_{k})-d^{(\mu)}(\jmath\omega_{l};\boldsymbol{\eta}_{k})\mathbf{H}_{l,k}\right]\right\|^{2} \tag{3.41}$$

at each iteration $\mu=1,2,\ldots$, where the iteration-dependent weight $w_{l,k}^{(\mu)}$ is defined as the inverse of the denominator estimate available at the previous iteration

$$w_{l,k}^{(\mu)}=\left[d^{(\mu-1)}(\jmath\omega_{l};\boldsymbol{\eta}_{k})\right]^{-1} \tag{3.42}$$

with the initialization $w_{l,k}^{(0)}=1$. The above SK formulation is a standard approach in linear and parameterized macromodeling. As discussed in Section 2.2 it allows to cast a global nonconvex optimization problem as a sequence of linearized problems (3.41), since the residual whose norm is being minimized at each iteration is an affine combination of the free variables $\mathbf{R}_{m,j}$ and $r_{m,j}$. Therefore, the numerical solution of (3.41) does not involve particular difficulties, requiring a simple linear least squares solver. There is however an additional difficulty, due to the fact that (3.41) will minimize the least squares error, without any control over the accuracy of the fitted model at prescribed frequency points, including DC. For the applications of interest, which requires an exact representation of the DC response of the small-signal macromodel, a better control is needed.

The DC response of the parameterized macromodel is readily computed from (3.37) as

$$\mathbf{H}(0,\boldsymbol{\eta})=\frac{\mathbf{N}(0,\boldsymbol{\eta})}{d(0,\boldsymbol{\eta})}=\frac{\sum_{m=0}^{M}\mathbf{R}_{m}(\boldsymbol{\eta})\phi_{m}(0)}{\sum_{m=0}^{M}r_{m}(\boldsymbol{\eta})\phi_{m}(0)}\,. \tag{3.43}$$

Denoting with

$$\mathbf{H}_{0,k}=\mathbf{H}(0;\boldsymbol{\eta}_{k}) \tag{3.44}$$

the DC value of the linearized response of the original system, which is easily extracted or extrapolated from a circuit solution of the original schematic, the parameterized macromodel can be enforced to match exactly this DC response by adding the following set of equality constraints

$$\sum_{m=0}^{M}\mathbf{R}_{m}(\boldsymbol{\eta}_{k})\phi_{m}(0)-\mathbf{H}_{0,k}\sum_{m=0}^{M}r_{m}(\boldsymbol{\eta}_{k})\phi_{m}(0)=0 \tag{3.45}$$

for $k=1,\ldots,K$ to the linear least squares problem (3.41). The constraints (3.45) are also expressed as affine combinations of the decision variables. Therefore, the minimization of (3.41) subject to (3.45) is easily achieved

through any standard solver for linearly-constrained linear least squares problems.

Once the macromodel coefficients are available, the computation of the DC bias correction sources using (3.48) is performed, for each of the available parameter grid values $\boldsymbol{\eta}_k$. Then, a parameterized set of DC correction sources is defined as a superposition of the basis functions $\xi_j(\boldsymbol{\eta})$ as

$$\bar{\Delta}_{DC}(\boldsymbol{\eta}) = \sum_{j=1}^{J} \bar{\Delta}_j \xi_j(\boldsymbol{\eta}). \tag{3.46}$$

The coefficients $\bar{\Delta}_j$ are computed by enforcing the fitting/interpolation condition (3.49) for each k, which requires the solution of a further linear least squares system.

3.2.4 The need for DC correction

Another issue may affect the above described macromodeling flow, possibly making the resulting small-signal parametric macromodels completely useless when employed in time-domain transient simulations. In fact, a direct replacement of the nonlinear CB with the Linear Transfer Function Model (LFTM) in a transient simulation setup leads to possibly incorrect biasing, since the small-signal macromodel does not include any information of the underlying DC operating point. When excited by constant inputs $\boldsymbol{u}(t) = \boldsymbol{u}_{DC}$, the LTFM (3.30) provides its closed form DC output solution

$$\begin{aligned} \check{\boldsymbol{y}}_{DC}(\boldsymbol{\eta}) &= \mathbf{H}(0;\boldsymbol{\eta})\boldsymbol{u}_{DC} \\ &= (\mathbf{D}(\boldsymbol{\eta}) - \mathbf{C}(\boldsymbol{\eta})\mathbf{A}^{-1}(\boldsymbol{\eta})\mathbf{B}(\boldsymbol{\eta}))\boldsymbol{u}_{DC}, \end{aligned} \tag{3.47}$$

which has no relationship with the true DC operating point of the original nonlinear CB. This information is not embedded in the LTFM, which only represents the dynamics of small signal variations around the bias point. This problem becomes severe when several CB's are connected together to form a complete RF transceiver path. If one of the CB models provides the incorrect DC bias as its output, which is in turn fed to the input of another block, the latter will not function properly due to inconsistent biasing, and the entire verification results will be wrong.

For the non-parametric case, or equivalently for any fixed instance $\boldsymbol{\eta} = \boldsymbol{\eta}_k$, it was shown in [90] and Section 3.1 how the correct DC bias can be recovered by adding suitable constant DC correction sources at the macromodel ports. Assume that for any discrete parameter value $\boldsymbol{\eta}_k$, the correct bias conditions provided by the input $\boldsymbol{u}_{DC}$ and output $\widehat{\boldsymbol{y}}_{DC,k} = \boldsymbol{y}_{DC}(\boldsymbol{\eta}_k)$ are known from

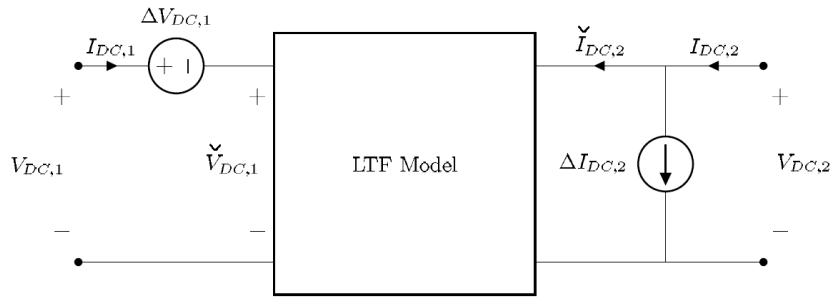

Figure 3.11: DC point correction for a two port LTFM, assuming a hybrid configuration with one current-controlled (left) and one voltage-controlled (right) port. The correct DC bias is set via constant current sources $\Delta V_{DC,1}$ and $\Delta I_{DC,2}$ applied at the input ports of the LTFM. The source values are provided by the elements of the correction vector (3.48).

a solution of (3.25) for the original non-linear system. The DC solution $\boldsymbol{y}_{DC}(\boldsymbol{\eta}_k)$ of the LTFM driven by the same nominal biasing inputs $\boldsymbol{u}_{DC}$ is computed as in (3.47), and the difference

$$\boldsymbol{\Delta}_{DC}(\boldsymbol{\eta}_k) = \boldsymbol{y}_{DC,k} - \check{\boldsymbol{y}}_{DC}(\boldsymbol{\eta}_k) \tag{3.48}$$

is evaluated, which represents the correction that must be applied to the DC solution of the LTFM in order to obtain the nominal CB bias level. The correction terms $\boldsymbol{\Delta}_{DC}(\boldsymbol{\eta}_k)$ are applied by defining an enlarged DC-corrected small-signal macromodel which embeds the original LTFM and adds at its interface ports suitable constant sources (see Figure 3.11), whose values are the components of $\boldsymbol{\Delta}_{DC}(\boldsymbol{\eta}_k)$. It should be noted that using constant correction sources will affect and fix the DC point only, without any effect on the accuracy of the LTFM around the operating point under small-signal excitation.

The above approach is valid only for a fixed parameter value $\boldsymbol{\eta} = \boldsymbol{\eta}_k$. Therefore, a new LTF macromodel extraction and a new computation of the DC correction sources for any new instance of the parameters is required. The main purpose of this work is to present a general strategy that is able to process the full set of samples (3.33), providing a DC-compliant reduced-order parameterized LTF macromodel that can be directly used to replace a transistor-level CB for any system-level time-domain verification and for any arbitrary parameter value $\boldsymbol{\eta}$ within an admissible range.

3.2.5 DC-compliant parameterized macromodeling

The proposed strategy for the extraction of a DC-compliant and parameterized small-signal macromodel can be summarized in the following steps:

1. create a suitable CB characterization test bench in the adopted circuit simulation environment and apply the desired biasing circuitry to each CB pin;

2. extract DC bias information $\boldsymbol{y}_{DC,k}$ and small-signal frequency response $\mathbf{H}_{l,k}$ of the CB from a set of circuit simulations of the non-linear system, here represented by (3.18)-(3.19), for a set of discrete parameter values $\boldsymbol{\eta} \in \{\boldsymbol{\eta}_k,\ k = 1, \ldots, K\}$ and at a discrete set of frequencies $\omega \in \{\omega_l,\ l = 1, \ldots, L\}$;

3. perform a parameterized rational curve fitting of the data $\mathbf{H}_{l,k}$ using a multivariate parametric macromodeling scheme, and obtain a state-space realization (3.30) of the LTFM;

4. compute $\check{\boldsymbol{y}}_{DC}(\boldsymbol{\eta}_k)$ from (3.47) and $\boldsymbol{\Delta}_{DC}(\boldsymbol{\eta}_k)$ from (3.48) over the discrete parameter grid $\boldsymbol{\eta}_k$;

5. interpolate the data $\boldsymbol{\Delta}_{DC}(\boldsymbol{\eta}_k)$ with a closed-form parametric expression $\bar{\boldsymbol{\Delta}}_{DC}(\boldsymbol{\eta})$ so that
$$\bar{\boldsymbol{\Delta}}_{DC}(\boldsymbol{\eta}_k) = \boldsymbol{\Delta}_{DC}(\boldsymbol{\eta}_k) \tag{3.49}$$

6. synthesize a circuit netlist with a standard parameterized macromodel realization, complemented by DC correction sources $\bar{\boldsymbol{\Delta}}_{DC}(\boldsymbol{\eta})$ connected at its external ports.

Figure 3.11 depicts the result of this process in terms of high-level schematic blocks. Next sections provide more details on the proposed modelling strategy for steps 3), 5), and 6).

3.2.6 Macromodel representation

The above described procedure results in a DC compliant parameterized small-signal macromodel $\mathbf{H}(s, \boldsymbol{\eta})$ defined in (3.37), plus a set of parameter-dependent DC correction sources $\bar{\Delta}_{DC}(\boldsymbol{\eta})$ defined in (3.46). These two block elements are connected as in Figure 3.11. The final step consists of casting these expressions in a form that can be used in a circuit solver of the SPICE class.

For the small-signal macromodel part, the detailed derivation in [81], see also [126, 142, 143] shows that $\mathbf{H}(s,\boldsymbol{\eta})$ can be easily converted into a parameterized descriptor form

$$\begin{aligned} \mathbf{E}\dot{\tilde{\boldsymbol{x}}}(t;\boldsymbol{\eta}) &= \mathbf{A}(\boldsymbol{\eta})\tilde{\boldsymbol{x}}(t;\boldsymbol{\eta}) + \mathbf{B}(\boldsymbol{\eta})\tilde{\boldsymbol{u}}(t), \\ \tilde{\boldsymbol{y}}(t;\boldsymbol{\eta}) &= \mathbf{C}(\boldsymbol{\eta})\tilde{\boldsymbol{x}}(t;\boldsymbol{\eta}) \end{aligned} \tag{3.50}$$

where

$$\mathbf{E} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \qquad \mathbf{A}(\boldsymbol{\eta}) = \begin{bmatrix} \mathbf{A}_1 & \mathbf{B}_1 \\ \mathbf{C}_2(\boldsymbol{\eta}) & \mathbf{D}_2(\boldsymbol{\eta}) \end{bmatrix} \tag{3.51}$$

$$\mathbf{B} = \begin{bmatrix} \mathbf{0} & -\mathbf{I}_P \end{bmatrix}^T \qquad \mathbf{C}(\boldsymbol{\eta}) = \begin{bmatrix} \mathbf{C}_1(\boldsymbol{\eta}) & \mathbf{D}_1(\boldsymbol{\eta}) \end{bmatrix} \tag{3.52}$$

and where

$$\mathbf{A}_1 = \text{blkdiag}\{q_m \mathbf{I}_P\}_{m=1}^{M} \tag{3.53}$$

$$\mathbf{B}_1 = [\mathbf{I}_P, \ldots, \mathbf{I}_P]^T \tag{3.54}$$

$$\mathbf{C}_1(\boldsymbol{\eta}) = [\mathbf{R}_1(\boldsymbol{\eta}), \ldots, \mathbf{R}_M(\boldsymbol{\eta})] \tag{3.55}$$

$$\mathbf{C}_2(\boldsymbol{\eta}) = [r_1(\boldsymbol{\eta})\mathbf{I}_P, \ldots, r_M(\boldsymbol{\eta})\mathbf{I}_P] \tag{3.56}$$

$$\mathbf{D}_1(\boldsymbol{\eta}) = \mathbf{R}_0(\boldsymbol{\eta}) \tag{3.57}$$

$$\mathbf{D}_2(\boldsymbol{\eta}) = r_0(\boldsymbol{\eta})\mathbf{I}_P \tag{3.58}$$

with q_m basis functions poles from (3.38). The main advantage of representation (3.50) is that those state-space matrix elements that are parameterized coincide with the coefficients $\mathbf{R}_m(\boldsymbol{\eta})$ and $r_m(\boldsymbol{\eta})$. Since polynomial basis functions $\xi_j(\boldsymbol{\eta})$ are used in the expansion, a SPICE synthesis of these equations is straightforward using elementary dependent sources with polynomial gain. The same consideration and synthesis applies for the DC correction sources $\bar{\boldsymbol{\Delta}}_{DC}(\boldsymbol{\eta})$.

3.2.7 Stability and passivity

The proposed macromodeling flow is applied here to describe the linearized behavior of active nonlinear CB's. Therefore, passivity verification and enforcement is not required at all since the original CB is not a passive device. Should the application at hand require a guaranteed passive parameterized macromodel, an internally passive parameterization should be used instead of (3.37). See [129, 130, 131, 132, 133, 134] for more details.

Conversely, uniform stability is important for any subsequent transient analysis. All macromodel poles (which depend on the parameters $\boldsymbol{\eta}$) should be confined into the left half complex plane for any value of the parameters

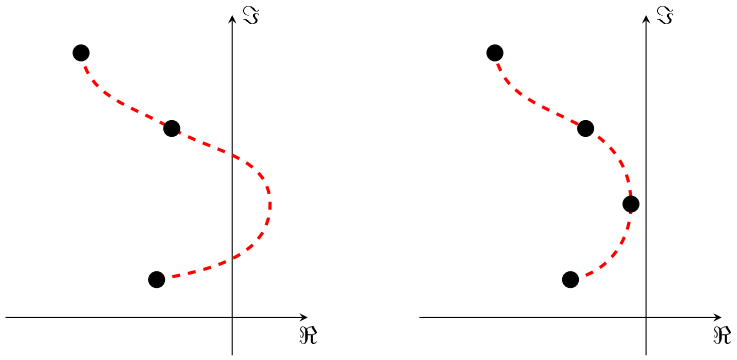

Figure 3.12: Left panel: using a coarse grid (dots) for model identification may lead to parameterized pole trajectories (dashed line) leaking into the right hand complex plane. Right panel: grid refinement constraints the parameterized poles into the stable region.

within the admissible range. A simplistic approach to enforce uniform stability is to not parameterize the poles at all, at the price of a reduced accuracy and generality of the small-signal macromodel. This is easily achieved by removing in (3.37) the dependence on the parameters $\boldsymbol{\eta}$ of the denominator coefficients $r_m(\boldsymbol{\eta})$, see [136]. In general, necessary and sufficient criteria that are able to guarantee uniform stability without compromising the macromodel accuracy, e.g. by imposing additional structure in the model equations, are still not available.

Guaranteed stable non-parameterized macromodels (3.36) for any fixed parameter value are easy to obtain, see [62]. When introducing the external parameters $\boldsymbol{\eta}$, the essential condition for preserving uniform stability is to start with a sufficiently dense parameter grid $\boldsymbol{\eta}_k$, so that all system poles are tracked with sufficient resolution between grid values. Figure 3.12 provides an intuitive illustration that instability may occur for coarse grids due to insufficient knowledge of the original system dynamics between grid values. A proper dense grid facilitates the enforcement of uniform stability.

In the proposed implementation, after computing an initial parameterized macromodel (3.37), the macromodel poles are computed, i.e. the generalized eigenvalues of pencil $(\mathbf{E}, \mathbf{A}(\boldsymbol{\eta}))$, over a dense grid in the parameter space. Note that this verification involves a minimal cost due to the compact size of the macromodel. Should unstable poles be detected for some parameter value $\boldsymbol{\eta}_*$, the identification grid $\boldsymbol{\eta}_k$ is enlarged by adding $\boldsymbol{\eta}_*$ and the macromodel is recomputed. This last repeated fitting stage was never required for all application examples that were tested.

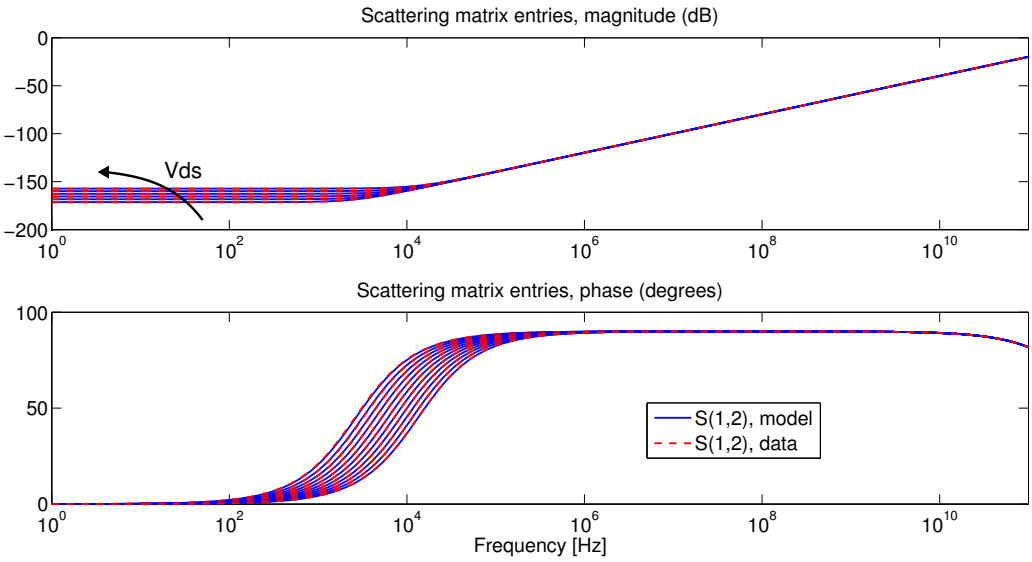

Figure 3.13: Magnitude (top) and phase (bottom) of S_{12} for the parameterized small-signal NMOS model (blue solid lines) compared to the corresponding original responses (dashed red lines), plotted for different values of the parameter V_{ds} ranging from 0.8 V to 1.2 V. The S_{12} is the response with the smaller values at DC for a sweep of the V_{ds}. This result demonstrates the effectiveness of the proposed DC enforcement strategy.

3.3 Examples

The effectiveness of the proposed methodology is demonstrated on three examples. The first two cases are very simple: a single NMOS transistor and a two-stage buffer. These examples are mainly used as a proof of concept. The third example is instead a fully implemented circuit block, namely a Low Dropout Voltage regulator used in a commercial 3G transceiver design.

3.3.1 A NMOS transistor

The first example illustrates the proposed methodology on a single NMOS transistor, for which a small-signal linearized model is derived using the source-drain bias voltage V_{ds} as a free parameter. A 3-port configuration is considered, where port one is the drain, port two the gate, and port three the bulk, all referenced to the source, as depicted in Figure 3.14. This is the typical test pattern used to characterize field effect transistors. Because of the technology used $V_{th} \approx 0.6V$. The NMOS is biased with $V_{gs} = 1.2V$ and $V_{bs} = 0V$. As a consequence a sweep of V_{ds} from 0.8 V to 1.2 V explores the

linear region of the NMOS characteristic. A comparison of the small-signal S_{12} response of the original device with the corresponding parameterized model is reported in Figure 3.13 for a V_{ds} sweep ranging from 0.8 V to 1.2 V. This figure demonstrates that, even if the dynamic variation of the responses is very large, the proposed DC constraint is able to guarantee a very accurate macromodel, even at low frequencies where the magnitude response is very small (lower than -150dB), thanks to the DC enforcement strategy proposed.

Figure 3.15 shows the computed parametric correction source to be applied to the input port (Gate) for DC compliance. Only the points marked with red squares were used for the macromodel identification, whereas the blue crosses indicate additional validation points used to verify the interpolation. As expected, the dependence of this correction source on V_{ds} is very smooth and therefore well captured by a low-order interpolation. The parameterized model has dynamical order 2, while both numerator and denominator polynomial bases (3.37) have degree 2. Finally, Figure 3.16 reports the parameterized macromodel (real) poles, that for this simple device show a weak and smooth dependence on the free parameter V_{ds} as a consequence of the small variation of charges in the MOS channel while working in the linear region.

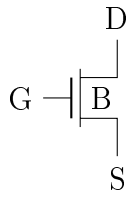

Figure 3.14: Typical characterization test pattern for NMOS.

3.3.2 A two-stage buffer

The second example is the two-stage buffer depicted in Figure 3.17. For this test case, two parameters are used: the supply voltage $V_{dd} \in [0.7, 1.2]$ V and the ambient temperature, in the range $T \in [-25°, 125°]$ C. The accuracy of the parameterized macromodel is demonstrated by comparing the small-signal S-parameter S_{22} of the original buffer to the macromodel response for two sweeps of V_{dd} and T in the two panels of Fig. 3.18. For this example, a dynamical order 4 was used, with both numerator and denominator polynomial bases (3.37) having degree 2.

Figure 3.19 depicts the parameterized DC correction sources at the supply and output ports of the buffer, comparing the raw data with the interpolated model. Considering that temperature effects in transistors models are described by low degree polynomials (two or three at most), these two dimensional correction functions can be expected to be smooth as well thanks to the proposed explicit DC constraint in the macromodel fitting. Therefore, a low-order interpolation scheme is appropriate. For this example it was used

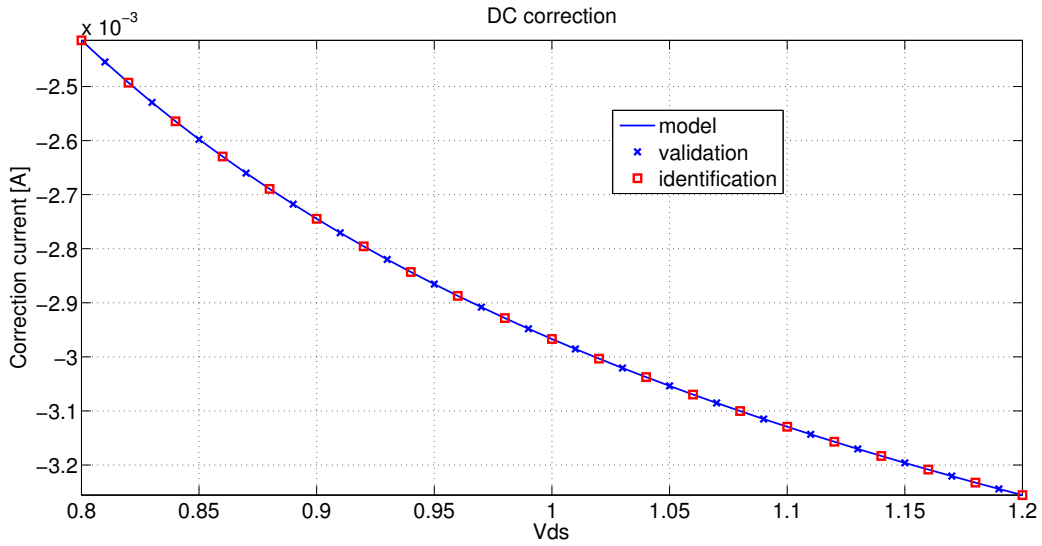

Figure 3.15: Parametric DC current correction source (Gate) for the small-signal NMOS model, plotted as a function of V_{ds}.

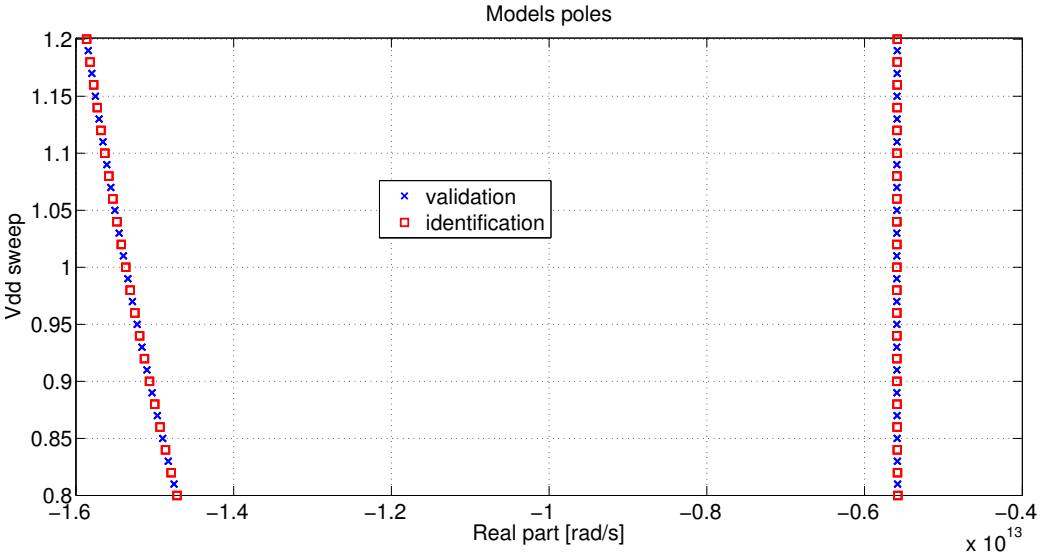

Figure 3.16: Parameter-dependent poles of the small-signal NMOS model.

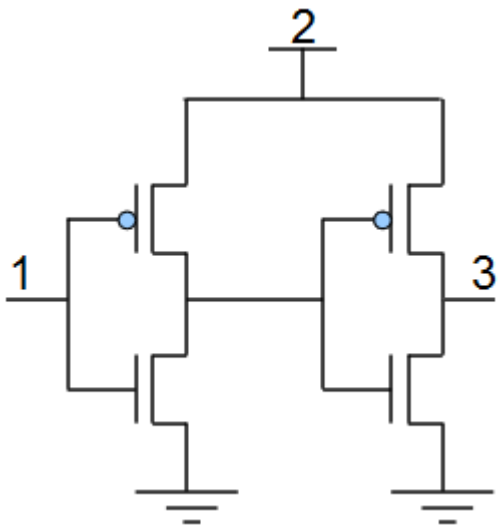

Figure 3.17: A two-stage buffer with ports numbering used in this work.

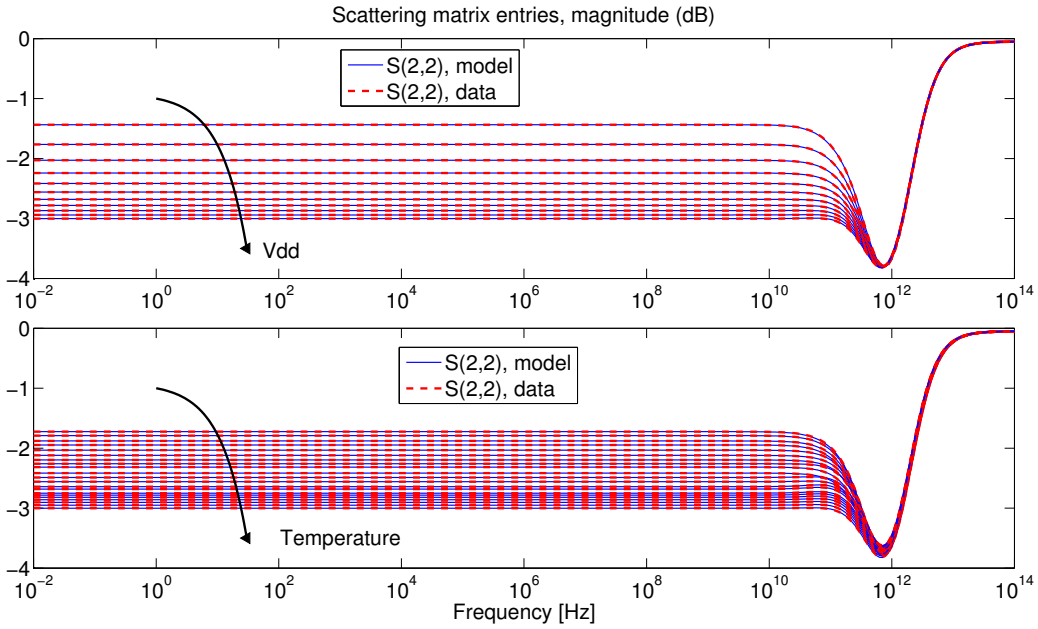

Figure 3.18: S_{22} for the parameterized small-signal buffer model (blue solid lines) compared to the corresponding original response (dashed red lines). In the bottom panel, V_{dd} is fixed to 0.75 V and the temperature sweeps in the range $-25 \div 120$°C, while in the top panel T is fixed to 20°C and V_{dd} sweeps from 0.7 V to 1.2 V. S_{22} is presented being the S-parameter with the wider variation with both parameters V_{dd} and T. The phase is not depicted because the variation with parameters V_{dd} and T is small.

a multivariate polynomial of order 2, leading to a root mean square error of the polynomial interpolation less than 10^{-5}.

3.3.3 A Low Dropout (LDO) voltage regulator

Here a Low Dropout (LDO) voltage regulator is considered, whose transistor level schematic is taken from a commercial 3G transceiver design. This device is intended to provide a stabilized output voltage, under control by external biases provided from the logic unit. The parameter that is considered is again the supply voltage $V_{dd} \in [1.2, 1.7]$ V. For such a sweep of V_{dd} and using a reference voltage of 0.6V, the LDO works in the linear region of the characteristic. The original schematic includes hundreds of transistors, therefore a reduced-order macromodel is desirable to reduce complexity and runtime in system-level simulations.

A representative scattering response of the computed parameterized macromodel is compared to the corresponding small-signal scattering response of the transistor-level netlist in Figure 3.20. Also for this case, it can be seen that an excellent accuracy is achieved for all values of the parameter V_{dd} within the range of interest. The parameterized model has dynamical order 16, while numerator and denominator polynomial bases (3.37) have respectively degree 3 and 2. Figure 3.21 shows the computed parametric correction source to be applied to the power supply port (V_{dd}) for DC compliance. Only the points marked with red squares were used for the macromodel identification, whereas the blue dots indicate additional validation points used to verify the interpolation.

The transient analysis result of the synthesized parameterized macromodel is compared to the response obtained using the transistor-level netlist in Figure 3.22. The simulation time for a short transient analysis like the one depicted in Figure 3.22 require $3s$ using the synthesized parameterized model and $30s$ using the transistor level model. The real benefit of the proposed methodology should be addressed on complex system level simulation scenarios: replacing several CB with parameterized-LTFM can lead to a tremendous complexity reduction while preserving the accuracy of the simulation.

3.3.4 A system-level application

The proposed macromodeling procedure is illustrated on a circuit block composed by a single OPerational Amplifier whose voltage source is provided by a Low-DropOut (LDO) voltage regulator; test bench is depicted in Figure 3.23.

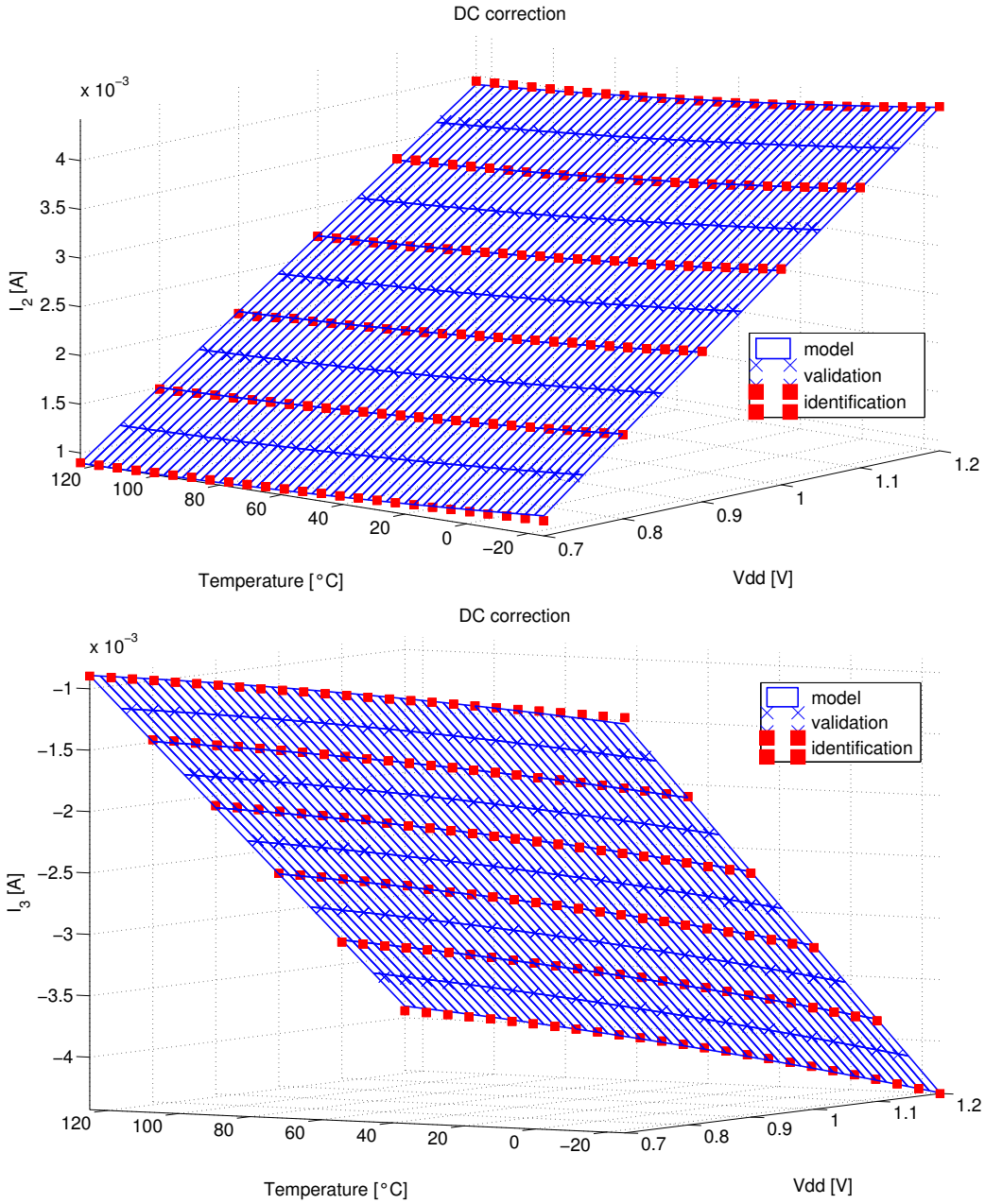

Figure 3.19: Parameterized DC correction sources for the supply (top) and output port (bottom) of the two-stage buffer. The current correction sources, are interpolated using a multivariate polynomial of order 2. The root means square error of the polynomial interpolation is lower than $1e^{-5}$.

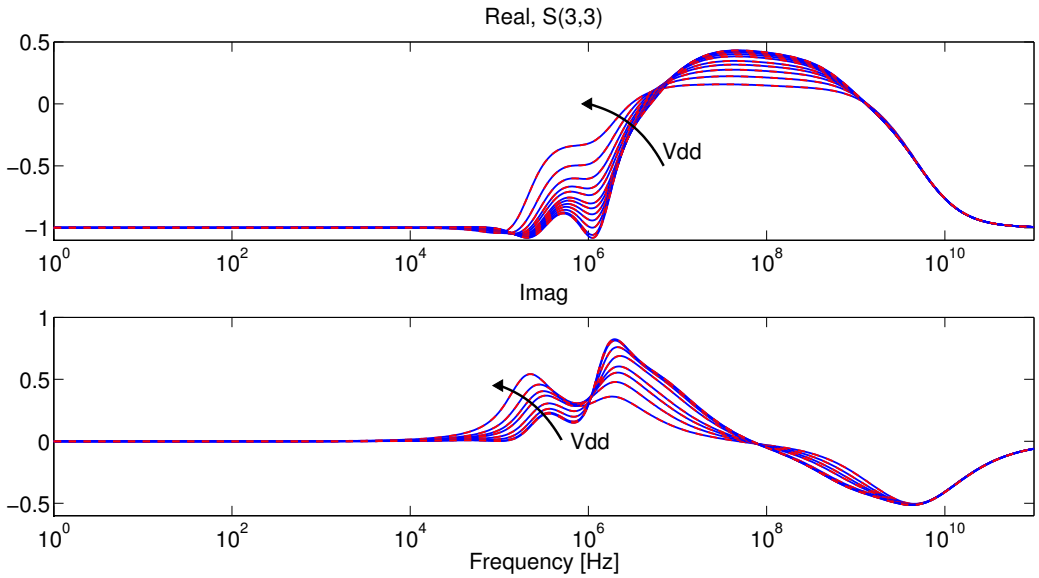

Figure 3.20: Real (top) and imaginary (bottom) part of S_{33} for the parameterized small-signal LDO model (blue solid lines) compared to the corresponding original responses (dashed red lines), plotted for different values of the parameter V_{dd} ranging from 1.2 V to 1.7 V. The S_{33} element is depicted being the one with the wider variation with V_{dd}.

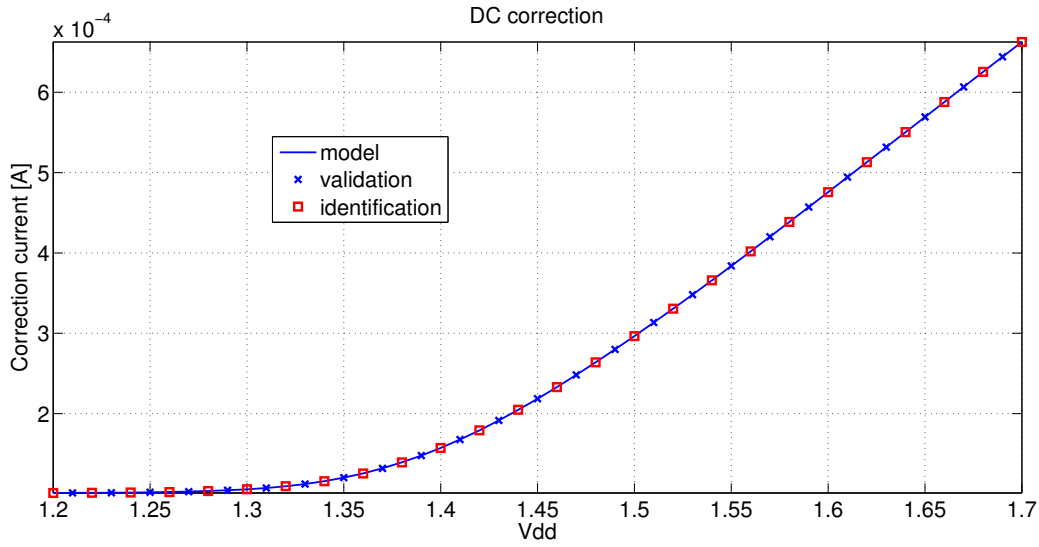

Figure 3.21: Parametric DC current [A] correction source (Power supply port V_{dd}) for the small-signal LDO model, plotted as a function of V_{dd}. Similar results are obtained for the current correction sources on the other ports.

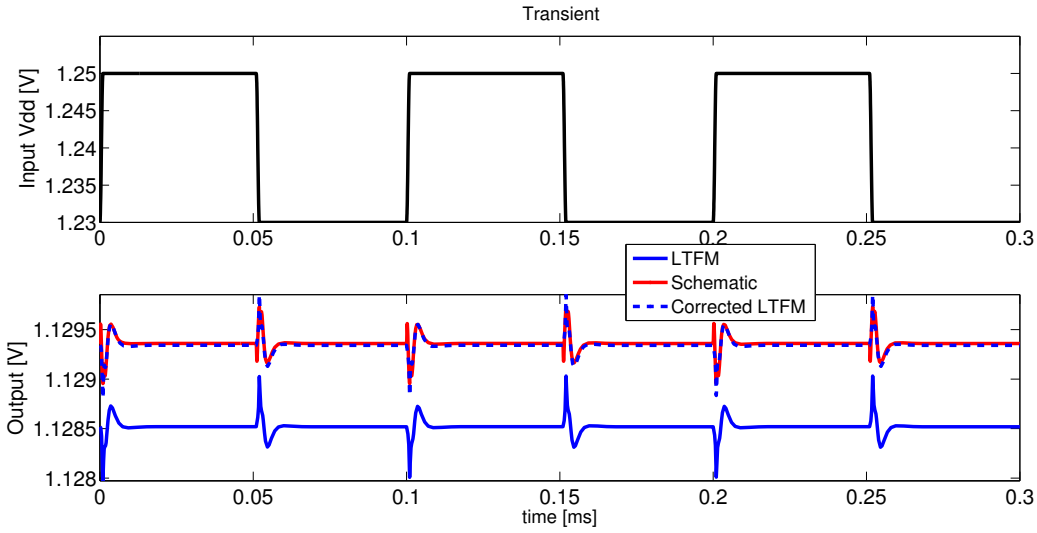

Figure 3.22: Transient simulation for a typical test case scenario of an LDO. The effect of a small variation on the V_{dd} is considered. In the top plot the square wave applied to the V_{dd} is depicted. In the bottom plot the transient response obtained from the synthesized parameterized-LTFM before the DC correction (blue continuous line) and after (blue dashed line) is compared with the response from the transistor level schematic.

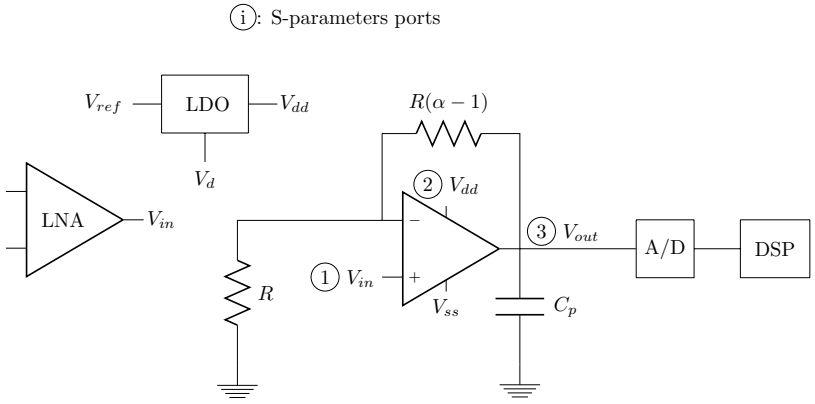

Figure 3.23: Subset of high-level circuit blocks inside an hypothetical base-band receiver chain.

The operational amplifier is a fundamental building block of analogue circuits, used in a vast variety of applications like A/D-D/A converters, high-speed wired/wireless transmitter and receivers and sensors. In conjunction with a negative-feedback network like in Figure 3.23, it can implement signal amplification, complex active filters, generation of voltage and current references and voltage buffers. In the context of RF and transceiver design OPAs are the building blocks for voltage regulators, low-noise amplifiers and active filters. Under proper biasing the OPA behaves almost linearly, i.e. it does not produce spurious harmonics thus preserving the quality of the signal in the transceiver chain introducing a negligible noise contribution.

For the results of this work a real OPA component implemented in a commercial 3G transceiver was used. The OPA circuit block, depicted in Figure 3.23, is parameterized by a supply voltage $V_{\rm dd} \in [1.1, 1.3]$ V with 20 mV steps and a gain $\alpha \in [1, 2]$ with steps 0.05, which are ranges of practical interest. Linearity and closed-loop stability were verified in practice by means of Spice simulations. The LDO model is parameterized by a $V_{\rm d} \in [1.2, 1.7]$ V using a nominal voltage reference $V_{\rm ref} = 0.6$ V. The effectiveness of the proposed methodology for voltage regulars was discussed in Section 3.3.3.

Figures 3.24-3.25 compare the computed macromodel responses to the original small-signal scattering responses for various combination of the parameters. The accuracy is excellent. These figures show that the variability induced by supply voltage variations is very small, whereas the sensitivity to a gain variation is larger. This is further demonstrated in Figure 3.26.

The same macromodeling process was also applied to the LDO in Section 3.3.3. Then, the parameterized macromodels of OPA and LDO were synthesized in SPICE, and a transient analysis was performed to validate the macromodel vs the full transistor level circuits. For illustration, a common signal-integrity scenario is addressed: the output from a differential LNA in a base-band receiver chain is amplified and filtered using an OPA. Signal quality and noise rejection are of paramount importance since the analog output from the OPA is then processed by and A/D converter and provided to a Digital Processing Block. Disturbances on the voltage reference $V_{\rm d}$, due to cross-talk or external noise sources must be handled by the LDO resulting in a stable $V_{\rm dd}$ for the OPA. Therefore a multi-tone (1 GHz-567 MHz-40MHz) multi-amplitude distortion is added to a 10 kHz square wave used as disturbance on the $V_{\rm d}$ of the LDO, while the input for the OPA is a 4 MHz square wave.

A small part of the input signal and the corresponding outputs are depicted in Fig. 3.28 for the OPA and in Fig. 3.27 for the LDO. A 200 μs transient simulation is required in order to properly asses the effect of the disturbances on the LDO voltage reference $V_{\rm d}$. The transistor level simula-

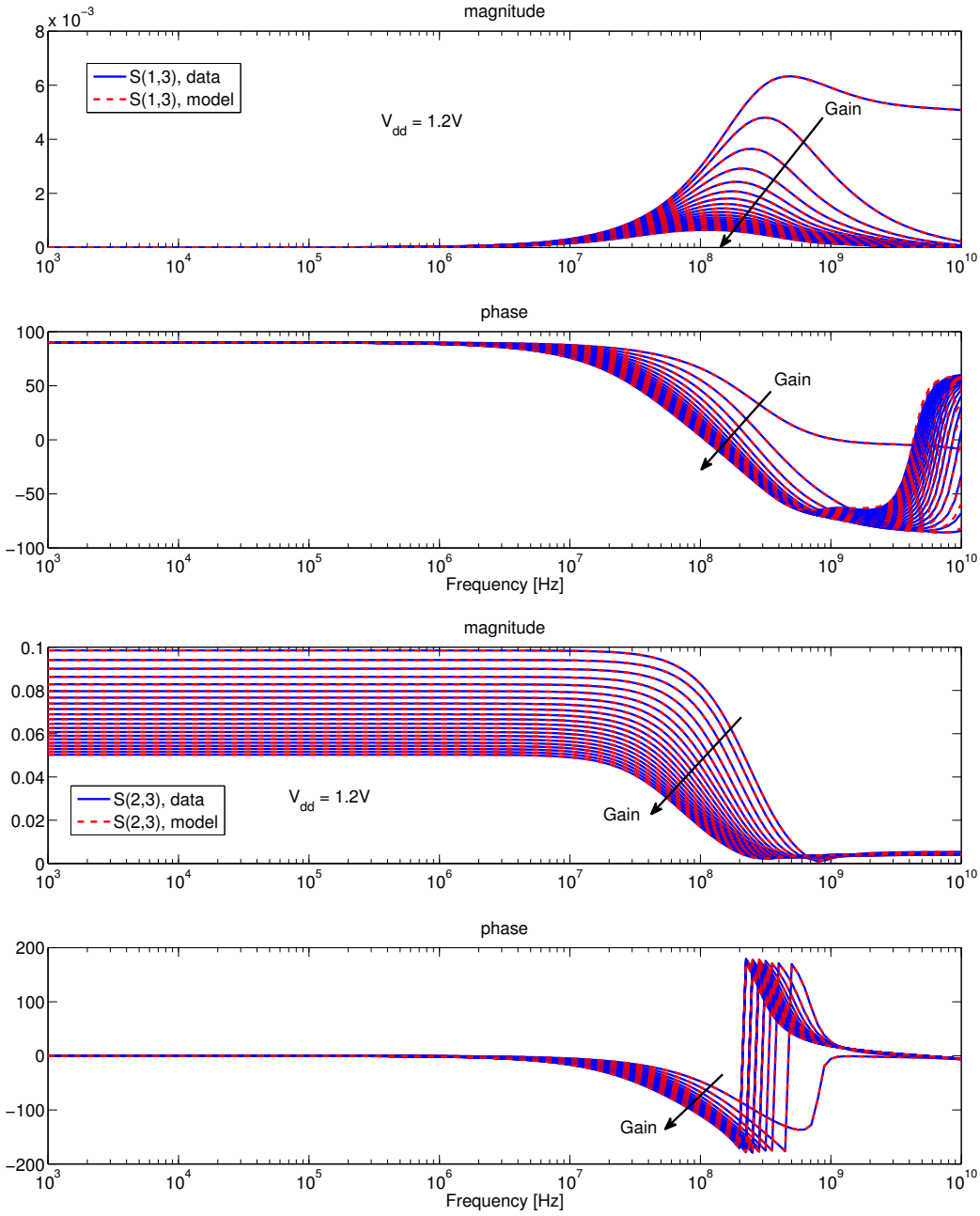

Figure 3.24: Comparison between parameterized small-signal macromodel (red dashed lines) and small-signal S-parameters $S_{1,3}$ and $S_{2,3}$ responses of the OPA circuit block for fixed supply voltage $V_{\mathrm{dd}} = 1.2$ V and variable gain.

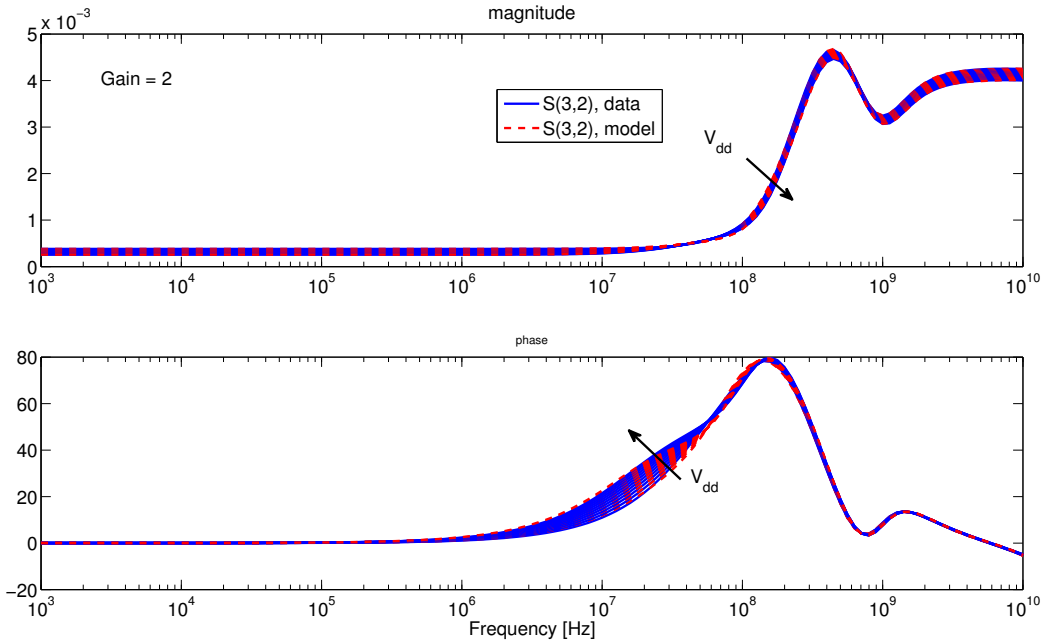

Figure 3.25: As in Fig. 3.24, but for $S_{3,2}$ with fixed gain $\alpha = 2$ and variable supply voltage $V_{\rm dd}$.

tion required 10 h. Such large simulation time is quite common and basically due to: the complexity of the transistor level models, involving 600 transistors, 100 diodes and 600 dynamical elements, and the complexity of the multi-tone disturbance on the LDO. The linear macromodel completed the simulation in only 8 minutes leveraging on the synthesized low order model: order 11 for the OPA and 16 for the LDO. As can be seen in Figures 3.27 and 3.28, such a tremendous speedup can be achieved with no compromise on accuracy. The figures further demonstrate the necessity of including DC correction sources, since the results without such sources present a clear DC offset. Dealing with two parameters, i.e. $V_{\rm dd}$ and α, the DC correction current sources were modelled using two-dimensional polynomials; results are depicted in Figure 3.29.

3.4 Conclusions

This Chapter presented a systematic methodology for the extraction of compact parameterized small-signal macromodels of complex nonlinear circuit blocks for Mixed-Signal and RF applications. Thanks to an explicit constraint on the DC response of the macromodel and to the inclusion of pa-

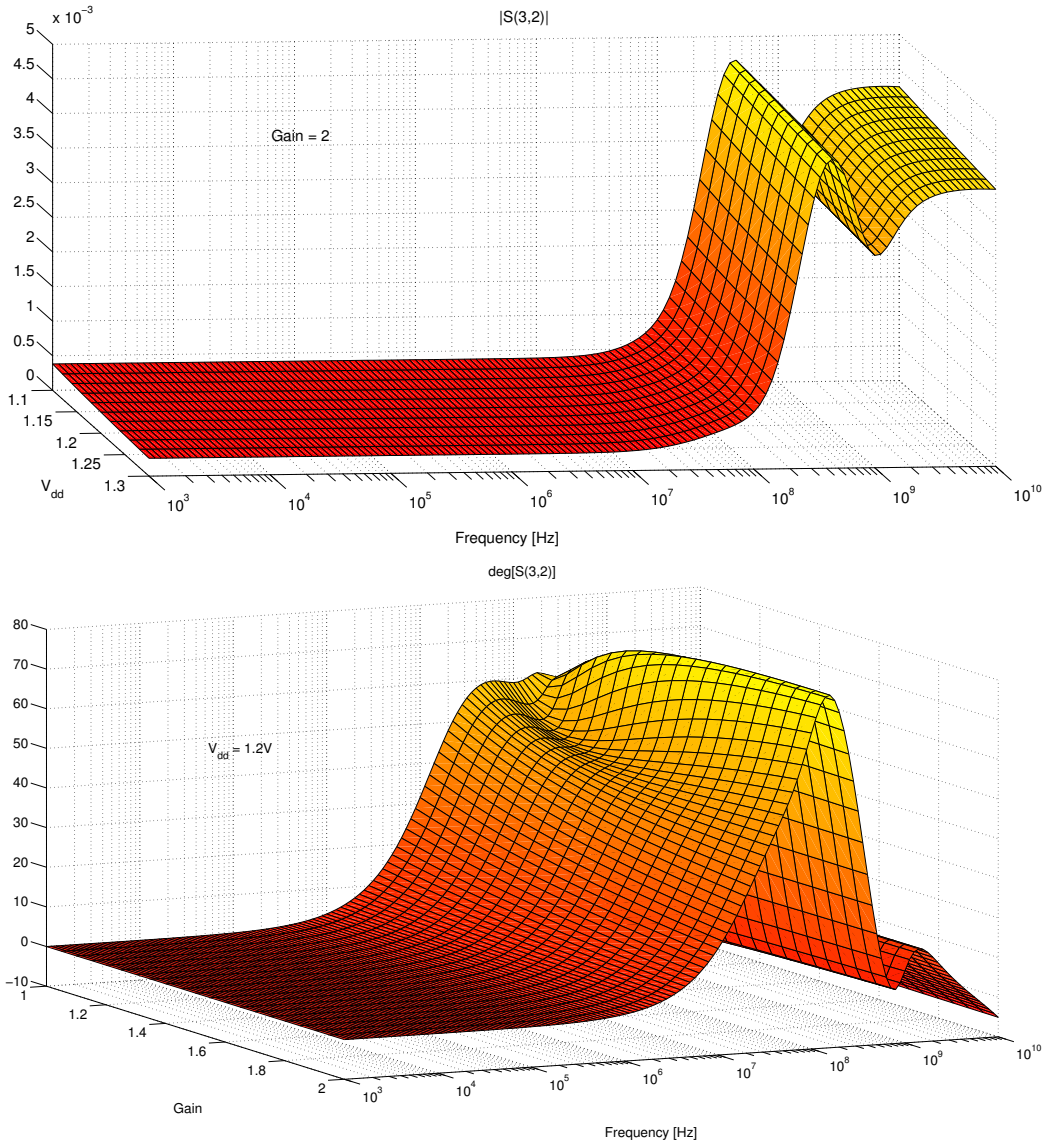

Figure 3.26: Illustration of the variability in the phase of a sample scattering response $S_{3,2}$ of the parameterized OPA macromodel with respect to supply voltage (top) and gain (bottom).

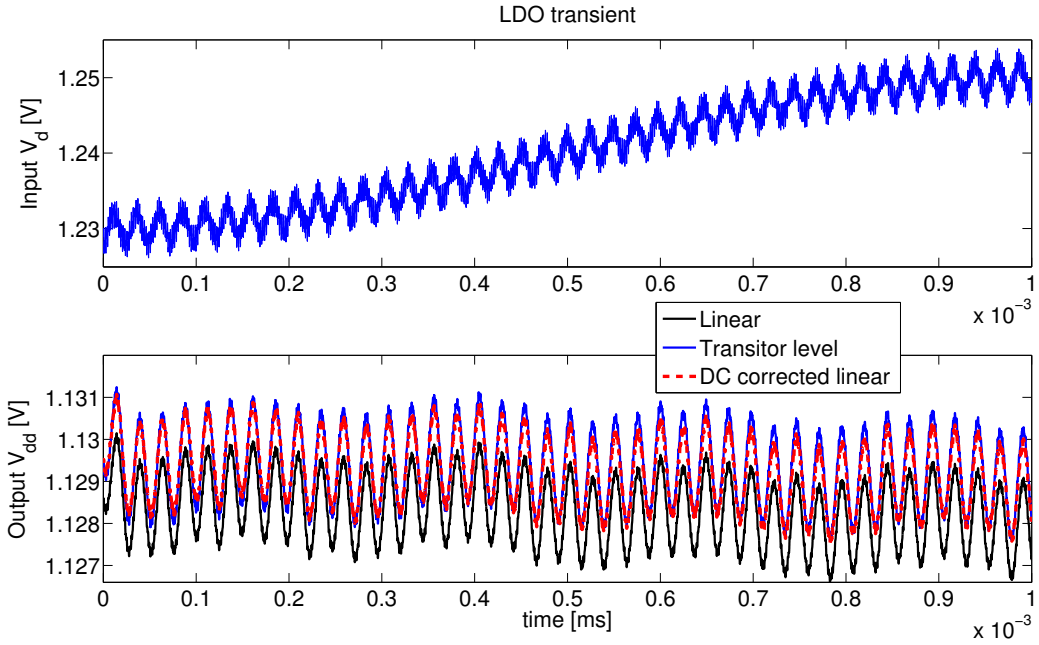

Figure 3.27: Bottom panel: detail of the transient analysis of the LDO transistor-level circuit block (blue continuous line) and parameterized macromodel, with (red dashed line) and without (black line) DC correction sources. The supply voltage affected by noise is depicted in the top panel. As expected, the output from the LDO is always close to the nominal value of 1.13 V.

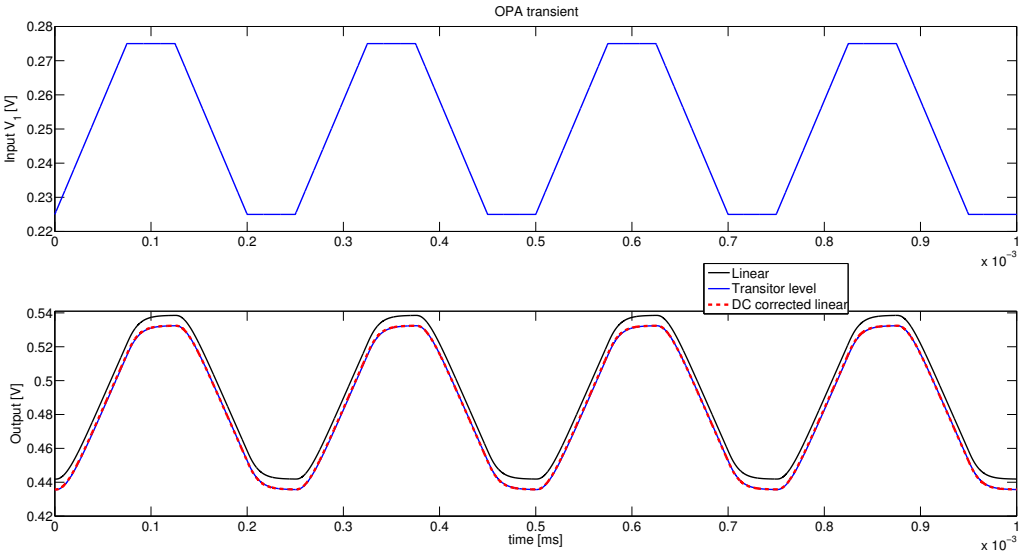

Figure 3.28: Bottom panel: detail of the transient analysis of the OPA transistor-level circuit block (blue continuous line) and parameterized macromodel, with (red dashed line) and without (black line) DC correction sources.

rameterized DC correction sources, the proposed macromodel can seamlessly replace the corresponding transistor level schematic in system-level Signal Integrity verifications, leading to a significant speedup in the computing time required by transient simulations.

The feasibility of the proposed approach was demonstrated on two simple academic examples (a single transistor and a two-stage buffer) as well as on two complex circuit models: a Low Dropout voltage regulator and an Operational amplifier, both taken from a real 3G transceiver design. The availability of accurate and efficient macromodels is considered as a key enabling factor for comprehensive system verification, allowing a fast systematic analysis of the large number of configurations and operation modes required by modern digitally-programmable systems.

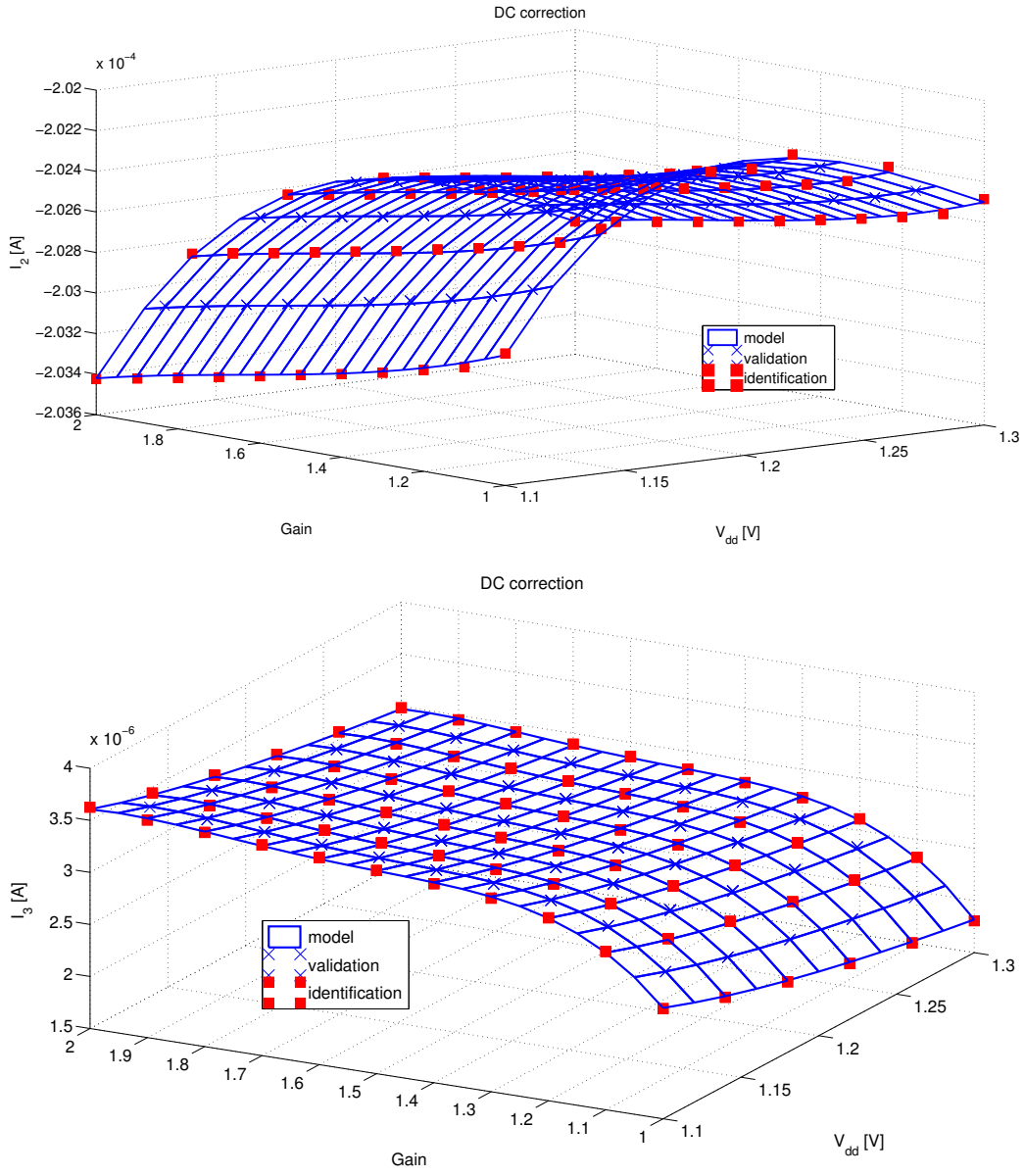

Figure 3.29: OPA current correction sources are depicted above as a function of V_{dd} and gain α. The 2D polynomial models (blue continuous line grid) attains a RMS error of $1e^{-5}$. Polynomial degree is 4 in both variables. The identification dataset is represented by red squares while the validation points are the blue crosses in the plots above. Correction sources I_1 for the input port of the OPA are not depicted here since negligible in magnitude.

Chapter 4

Noise-compliant macromodel synthesis

Lumped passive network synthesis, according to Cauer's definition [144, 145], consists in the design of a primitive network, i.e. a circuit composed only of primitive elements (inductances, mutual inductances, resistances and capacitances), which exhibits a prescribed frequency response. Such a definition is strictly related with the design of linear filters, hot topic at the beginning of last century. In spite of the specific application, Cauer's approach was very general since he was the first to consider the synthesis of a linear network as an inverse problem of network analysis. In this work, network synthesis is mainly used to convert linear macromodels, like the ones described in Chapter 2 and 3, into Spice-based compliant netlists for system-level simulations.

The synthesis of Linear Time Invariant (LTI) networks dates back to the beginning of the last century with the pioneering work of Foster [146] and Darlington [147]. The fathers of network theory: Cauer [148], Brune [149], Tellegen [150] and later Bott and Duffin [151] were the first to study and establish synthesis procedures for 1-port networks described as positive real functions. The natural extension to the N-port case required the study of positive real matrices [107], leading to the more general results of Belevitch [109]. Two are the main classes of linear network synthesis methods: non-reciprocal methods, based on the usage of gyrators [152] (passive and non-reciprocal network elements) and gyratorless (reciprocal) methods. The required conditions for both methods were deeply studied in [109].

After the introduction of State-Space techniques for the analysis of linear networks made by Bashkow [48] in 1957, several authors adopted the State-Space approach also in the synthesis context. As it was noted by Anderson [153] and Youla [154] the state-space methodology is theoretically sound and easier to be interpreted from a physical perspective. Moreover,

this mathematical approach was well consolidated in Physics and Control Theory, leading to a convenient transfer of knowledge to the Circuit Theory field. The seminal idea for an application of Control Theory results to the problem of network synthesis can be found in the work of Kalman [155]. Indeed the Reactance Extraction Method [154, 119], cornerstone for all the successive synthesis techniques, was inspired by [155]. From a state space perspective the conditions for a passive and reciprocal (gyratorless) synthesis can be easily related with the physical properties of the network [119], avoiding the involved and abstract mathematical details required by the previous methods [109]. The set of primitive network elements required for the reciprocal synthesis of a passive network can be restricted at most to the RLCT subset, i.e.: Resistors, Inductors, Capacitors and ideal multi-port Transformers [156]. Several years of research activity were devoted to further reduce this RLCT subset. The question whether ideal transformers could be avoided in the synthesis of passive networks had no answer for a long time, see [151] and [157] as an example. In [156] McMillan found the solution demonstrating that ideal transformers can not be avoided in the synthesis of passive networks. Motivated by the last results about the need to use ideal transformers some research effort was spent to reduce the complexity of the Reactance extraction algorithm [154], being the canonical[1] synthesis of choice.

Although this topic was widely studied and is well consolidated, no effort has been devoted so far in order to characterize the noise compliance of the synthesized network, i.e. under which conditions the obtained network shows the same input-output noise response in comparison with the original physical system/circuit. Therefore, in this Chapter several of the most popular network synthesis algorithms are analysed focusing on noise compliance properties. As discussed in Section 1.3.1, the transition from hard to firm IP blocks requires the availability of flexible netlists, i.e. the same circuit description should fit to multiple simulation contexts: transient, S-parameters and noise analysis. As a consequence, the adopted synthesis algorithm must preserve not only input-output response of the original system but also the noise behaviour.

This Chapter deals with the noise-compliant synthesis of linear, lumped, finite, time invariant, and passive networks. Section 4.1 introduces the problem of noise-compliance in the synthesis process. Section 4.2 discusses static network synthesis, i.e. time/frequency independent circuits. This case is of

[1]The Reactance Extraction Synthesis is canonical, i.e. requires the minimum number of passive elements only in the case of non-reciprocal synthesis (using gyrators). In the reciprocal (gyratorless) case only the minimum number of dynamic elements (C and L) or the minimum number of resistors, but not both minima together, can be guaranteed [119].

interest by itself since static networks are sufficient to perform connectivity and static IR drop analysis for SoC devices like discussed in Section 1.2. The extension to dynamic networks is treated in Section 4.3. For each section results and example test cases are provided.

4.1 Problem statement

In the context of linear passive networks thermal noise (also known as Gaussian or white noise) is the most relevant intrinsic noise contribution. It results in a small fluctuation of voltage and current at the ports of a source-free passive device. This phenomenon was predicted by A. Einstein in 1905 [158] as a consequence of the Brownian motion of free electrons inside a piece of metal in thermal equilibrium. Then, it was first observed by Johnson in 1928 [159] and explained by Nyquist the same year in [160].

Physically, white noise is due to the random thermal motion of free electrons inside a piece of conductive material which leads to temporary agglomeration of carriers. At macroscopic level it implies a floating (in magnitude and polarity) potential difference between two conductor ends. In a physical resistor this is perceived as a fluctuation in the electrical current (if the resistor is in a closed loop) or in the electrical voltage across its terminals (if the resistor is open-circuited). In both cases the Direct Current (DC) component of the fluctuation is zero.

In [160] Nyquist demonstrated that for linear resistances in thermal equilibrium at temperature T, the current or voltage fluctuations are quite independent of the conduction mechanism, type of material and shape and geometry of the resistor. The generated thermal noise depends exclusively upon the value of the resistance and its temperature. This result is known as Nyquist's theorem and can be written as

$$\bar{V}_n^2(\omega) = 4K_bTR\,, \tag{4.1}$$

with $\omega = 2\pi f$, $K_b = 1.38065\,10^{-23} J/K$ (Boltzmann constant), resistor value R and T temperature expressed in Kelvin. $\bar{V}_n$ is the noise voltage spectral density and is measured in $V/\sqrt{Hz}$. A dual result holds for the current, i.e.

$$\bar{I}_n^2(\omega) = 4K_bTG\,, \tag{4.2}$$

where $G = 1/R$ is the conductance.

Two equivalent circuit models for a noisy resistor follow from (4.1) and (4.2):

- Thevenin model: composed by a noiseless resistor in series with a noise voltage source based on (4.1);

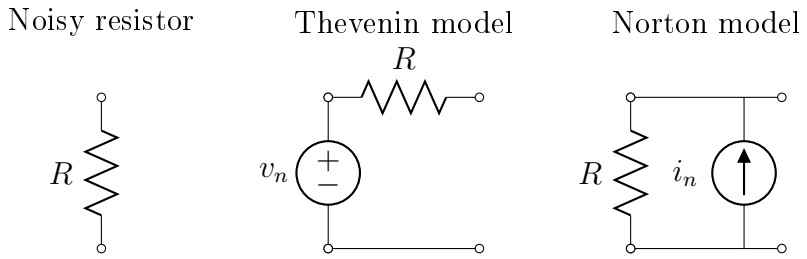

Figure 4.1: Thevenin and Norton equivalent circuits for a noisy resistor.

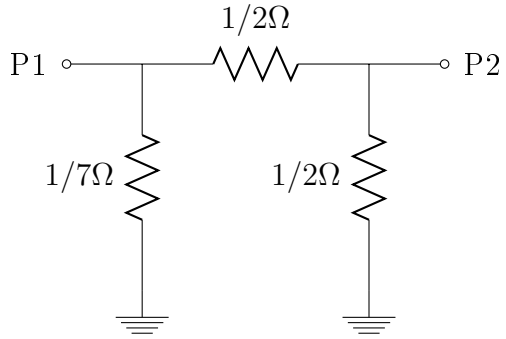

Figure 4.2: Simple 2-port resistive circuit.

- Norton model: composed by a noiseless conductance in parallel with a noise current source based on (4.2).

Figure 4.1 depicts Thevenin and Norton equivalent circuits for a noisy resistor.

The noise analysis on a passive network can be easily performed relying on the equivalent circuits depicted in Figure 4.1. As an example consider the 2-port resistive circuit in Figure 4.2. Noise analysis is conducted on the equivalent circuit in Figure 4.3, where each noisy resistor was substituted with the equivalent Thevenin model. To further simplify calculations it is assumed that resistor noise sources are uncorrelated (statistically independent, a condition that is verified in practice). The output noise voltage spectral density v_o on port 2 is evaluated given the input voltage v_{in} on port 1. Shorting the first resistor and its noise source, using linear superposition it follows

$$v_o^2 = \left(\frac{1/2}{1/2+1/2}\right)\underbrace{2K_bT}_{v_{n_2}^2} + \left(\frac{1/2}{1/2+1/2}\right)\underbrace{2K_bT}_{v_{n_3}^2} = K_bT\,, \tag{4.3}$$

for $T = 300K$ the voltage noise spectral density is

$$v_o = 6.43e^{-11}V/\sqrt{Hz}\,. \tag{4.4}$$

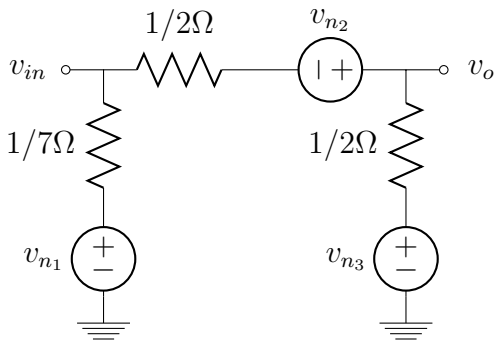

Figure 4.3: Equivalent circuit for the noise analysis of the simple 2-port resistive circuit in Figure 4.2.

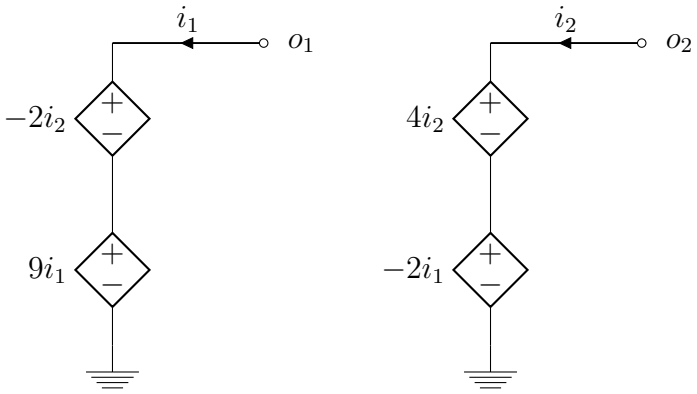

Figure 4.4: Equivalent controlled sources network for the resistive network in Figure 4.2.

This result is verified with a noise analysis in Spice, since Spice uses internally the equivalent circuits of Figure 4.1

It is well known that a resistive network is electrically equivalent to a circuit composed of controlled sources. Given the conductance matrix for the resistive network in Figure 4.2

$$\mathbf{G}_{e1} = \begin{bmatrix} 9 & -2 \\ -2 & 4 \end{bmatrix} \tag{4.5}$$

an equivalent circuit solely composed of controlled sources is depicted in Figure 4.4. Since for controlled sources there does not exist an equivalent noise model, the network in Figure 4.4 is noiseless. Indeed performing the noise analysis in Spice leads to zero voltage and current noise spectral densities. Therefore, different circuit realizations that are equivalent in the input-output responses are not equivalent for what concern the noise analysis. Such

an issue motivates the investigation of a noise-compliant network synthesis. As it will be further explained in Section 4.2 and 4.3, in order to obtain a noise-compliant synthesis, controlled sources can only be used to realize noiseless (lossless) components.

4.2 Static network synthesis

Static networks are used to perform connectivity analysis and static IR drop verification for Power Distribution Networks (PDNs). As discussed in Section 1.2, the design of PDNs is becoming more and more cumbersome due to low power constraints. The availability of a noise compliant static network is therefore of paramount importance in order to asses Power Integrity.

In the following, some basic notions related with the description of static networks are provided in Section 4.2.1, then the synthesis with a predefined network topology is considered in Section 4.2.2, and in Section 4.2.3 the canonical RT (Resistors and ideal Transformers) synthesis is summarized.

4.2.1 Basic assumptions

Some fundamental results from network theory [109] are briefly summarized in the following since they are needed to tackle the static synthesis problem. In particular: matrix description of static networks, existence of each representation and passivity and reciprocity conditions are considered. For an exhaustive presentation of those topics refer to [161].

Several equivalent (when they exist) representations can be used to describe a static network. Applying Kirchhoff current and voltage laws network variables can be ordered (and weighted, in the case of scattering parameters) obtaining: impedance $\mathbf{R}$, admittance $\mathbf{G}$, hybrid $\mathbf{H}$ and scattering $\mathbf{S}$ matrices. It was demonstrated in [109] that for passive networks it is always possible to combine voltages and currents in order to obtain a hybrid matrix (a similar result hold for the Scattering matrix case dealing with power waves). Casting networks description in matrix form greatly simplifies analysis and synthesis methods. Indeed passivity and reciprocity conditions, which are the constraint of interest here, can be summarized in Table 4.1. To further simplify the notation for the characterization of reciprocity in the Hybrid case the external signature matrix $\mathbf{S}_{ext} \in \mathbb{R}^{P \times P}$ is introduced. In general, signature matrices [162] are diagonal matrices with 1 or -1 on the main diagonal. In the particular case of hybrid matrices the external signature matrix is defined

Table 4.1: Summary of passivity and reciprocity conditions for static network matrix representations. In the reciprocity condition for the Hybrid matrix the external signature matrix $\mathbf{S}_{ext}$ defined by (4.6) is used. In the passivity conditions, zero equality means that the network is lossless, i.e. does not absorb active power.

Type	$\mathbf{R}$	$\mathbf{G}$	$\mathbf{H}$	$\mathbf{S}$
Passivity	$\mathbf{R}+\mathbf{R}^T \succeq 0$	$\mathbf{G}+\mathbf{G}^T \succeq 0$	$\mathbf{H}+\mathbf{H}^T \succeq 0$	$\mathbf{S}+\mathbf{S}^T \preceq 2\mathbf{I}$
Reciprocity	$\mathbf{R}=\mathbf{R}^T$	$\mathbf{G}=\mathbf{G}^T$	$\mathbf{S}_{ext}\mathbf{H}=\mathbf{H}^T\mathbf{S}_{ext}$	$\mathbf{S}=\mathbf{S}^T$

as

$$S_{ext_{i,i}} = \begin{cases} 1, & \text{port } i \text{ is voltage (current) controlled} \\ -1, & \text{port } i \text{ is current (voltage) controlled} \end{cases}, \tag{4.6}$$

From a mathematical perspective, the circuit synthesis of a generic (not necessarily associated to a physical circuit) matrix $\mathbf{N} \in \mathbb{R}^{M\times N}$, given N input variables $\boldsymbol{i}$ and M outputs $\boldsymbol{o}$ can always be performed using controlled sources. Despite its simplicity, such an approach is the core idea behind all the static (and also dynamic) synthesis methods presented in the following sections. Due to its relevance, a small example is provided to further clarify the previous statement: consider a generic matrix $\mathbf{N} \in \mathbb{R}^{M\times N}$ and the resulting system of equations

$$\begin{cases} o_1 &= n_{11}i_1 + \ldots + n_{1n}i_n \\ \vdots & \\ o_m &= n_{m1}i_1 + \ldots + n_{mn}i_n \end{cases} \tag{4.7}$$

assuming that the inputs $\boldsymbol{i}$ are currents and the outputs $\boldsymbol{o}$ are voltages, a straightforward synthesis is depicted in Figure 4.5. Although simple, such approach produces a canonical synthesis for the system (4.7) which exactly reproduces the system in (4.7). Similar realizations are straightforward if (i, o) are (voltages, currents). At this point the question is whether a generic matrix $\mathbf{N}$ describes a stable, passive and noise compliant network (passive and stable) and how to preserve those properties during a synthesis process.

The list of properties that should be considered when dealing with the synthesis of a linear circuit is provided here:

- canonicity: a synthesis is canonical when it involves the minimum possible number of primitive network elements to synthesize all the systems

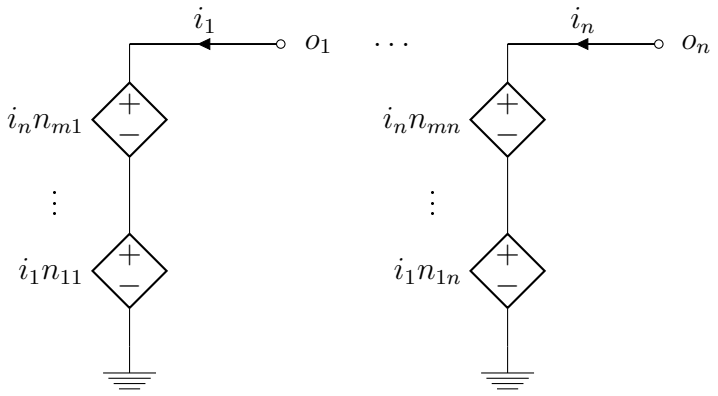

Figure 4.5: Direct synthesis for the system of equations (4.7).

describing a particular set of networks[2]. The canonical property is of paramount importance to keep under control the complexity of the network resulting from the synthesis process.

- passivity: is defined as the incapability of the network to provide a power gain. Passivity conditions for static network descriptions are summarized in Table 4.1. A linear network that does not absorb active power is lossless.

- reciprocity: a P-port circuit is reciprocal if for any pair of voltages and currents at circuit ports, i.e. $\{\boldsymbol{v}_a, \boldsymbol{i}_a\} \in \mathbb{R}^P$ and $\{\boldsymbol{v}_b, \boldsymbol{i}_b\} \in \mathbb{R}^P$ that satisfy the circuit characteristics, it holds

$$\boldsymbol{i}_b^T \boldsymbol{v}_a = \boldsymbol{i_a}^T \boldsymbol{v}_b \,.$$

  All linear multi-port circuits composed only by RLCT elements are reciprocal [163, 164]. Also for this property, a generic synthesis based on controlled sources must meet the conditions listed in Table 4.1 in order to preserve reciprocity. A circuit only composed by RLCT elements will be called purely-reciprocal in the following.

- topology-based: the primitive network elements involved by the synthesis procedure can be connected according to a specific topology. The most common configurations are the Π and T, see Figure 4.6. As discussed in Section 4.2.2 it is not possible in general to obtain a passive

[2]Please note that in the classic textbooks of network theory [109] canonical syntheses are characterized considering the number of independent variables involved by the synthesis. Instead in this work the number of primitive network elements is considered as the metric of canonical definition begin more intuitive and immediate from a practical point of view.

and reciprocal synthesis with the added constraint of topology. Conditions under which a fixed topology passive and reciprocal synthesis is possible are discussed in Section 4.2.2 as well.

- noise compliance: this is a more sensitive property since it is strictly related to the components used to perform the synthesis. Due to the fact that controlled sources are by definition noiseless components, the only way to achieve a noise compliant synthesis consist in the use of resistors. One possible approach consists in the selection of a particular topology leading to a pure resistive network synthesis. Unfortunately the latter method can not be used in general, since as explained in more details in Section 4.2.2, topology constraints may lead to non-physical networks (due to the need of negative resistors). In the general case a synthesis based on positive Resistors and ideal Transformers (RT) is the noise compliant synthesis of choice for static networks, see Section 4.2.3 for details, since it extracts a purely resistive sub-network (having only positive resistors) that connected to a proper lossless one realizes the desired circuit. In this way, the noise associated to the resistive subnetwork is not altered by the lossless one (which can also be non-reciprocal in general) thus producing a noise-compliant synthesis by construction.

4.2.2 Fixed topology

Several years of research and considerable effort were spent in order to perform the synthesis of a static network with a fixed topology [165, 166, 167] to avoid the need of ideal transformers like in the synthesis of Belevitch [109]. The final and negative answer to the general feasibility of a passive and reciprocal network synthesis that does not involve ideal transformers was provided only in more recent times by McMillan [156].

Sufficient and necessary conditions for a pure resistive synthesis based on an admittance (similar results hold for the impedance) matrix $\mathbf{G}$ where already considered in [109]

- dominant matrix: the admittance $\mathbf{G} = \mathbf{G}^T \in \mathbb{R}^{P\times P}$ is dominant iff

$$g_{i,i} - \sum_{\substack{j=1 \\ j\neq i}}^{P} |g_{i,j}| \geq 0, \tag{4.8}$$

- superdominant matrix: a dominant matrix satisfying $g_{i,j} < 0 \quad \forall \quad i \neq j$ is called superdominant [109].

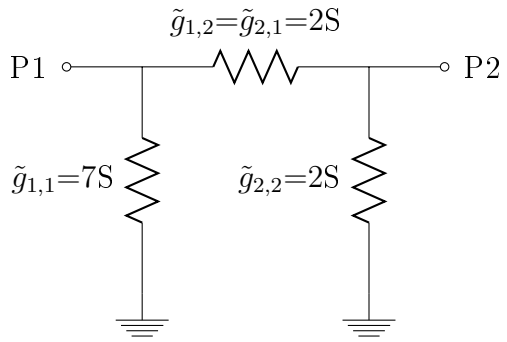

Figure 4.6: Π topology synthesis for the impedance matrix $\mathbf{G}_{e1}$ (4.11) based on (4.9) and (4.10).

A real symmetric superdominant admittance matrix $\mathbf{G}$ admits a pure resistive synthesis with Π topology. The admittances $\tilde{g}$ between nodes i and j come directly from the admittance matrix $\mathbf{G}$ according to

$$\tilde{g}_{i,j} = -g_{i,j} \tag{4.9}$$

$$\tilde{g}_{i,i} = \sum_{j=1}^{P} g_{i,j}. \tag{4.10}$$

As an example consider the simple admittance matrix $\mathbf{G}_{e1}$ for a 2-port network

$$\mathbf{G}_{e1} = \begin{bmatrix} 9 & -2 \\ -2 & 4 \end{bmatrix}. \tag{4.11}$$

$\mathbf{G}_{e1}$ is superdominant (positive definite). As a consequence the direct application of (4.9) and (4.10) leads the Π topology synthesis depicted in Figure 4.6. Although superdominant matrices are positive definite, i.e. passive by construction, the opposite is not true, thus reducing the applicability of this synthesis methodology and requiring the use of ideal transformers discussed in the following Section 4.3. A small example demonstrates that it is not difficult to construct a symmetric positive definite matrix that is not dominant. Consider the simple 2×2 symmetric positive definite matrix

$$\mathbf{G}_{e2} = \begin{bmatrix} 9 & -5 \\ -5 & 4 \end{bmatrix} \tag{4.12}$$

applying (4.9) if follows $\tilde{g}_{2,2} = -1$, thus a Π topology synthesis based only on positive resistors is not feasible for such a simple case, and ideal transformers are needed like discussed in the following section.

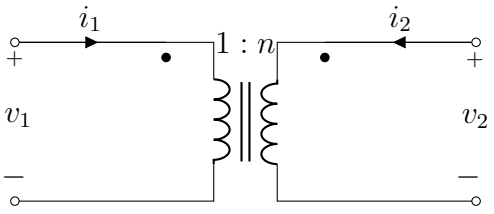

Figure 4.7: Ideal transformer circuit symbol. The first port on the left side is called the primary port while the second port on the right is called secondary port. The coupling factor n is defined as the ratio between the number of turns on the primary N_1 and on the secondary N_2 port.

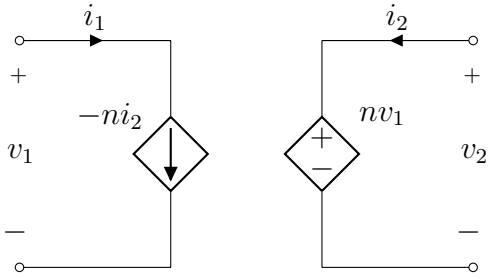

Figure 4.8: Equivalent circuit for the 2-port ideal transformer based on equations (4.13)-(4.14).

4.2.3 Synthesis with Resistors and ideal Transformers

To overcome the intrinsic limitations of the purely resistive synthesis of Section 4.2.2, Belevitch introduced in [109] the use of multiport ideal transformers. It was demonstrated in [156] that ideal transformers, together with resistors, inductors and capacitors, form the smallest set of network elements needed for the synthesis of passive reciprocal linear systems. The restriction to the static case further limits this set to ideal transformers and positive resistors only. The constitutive equations for a 2-port ideal transformer are

$$i_1 + ni_2 = 0 \tag{4.13}$$

$$v_2 = nv_1, \tag{4.14}$$

and the component symbol is depicted in Figure 4.7. One possible circuit equivalent for the ideal transformer based on (4.13)-(4.14) is depicted in Figure 4.8. It is important to note that the total instantaneous power absorbed by an ideal transformer is zero [109], since applying the constitutive equations (4.13)-(4.14) for the two port case leads to

$$i_1 v_1 + i_2 v_2 = 0. \tag{4.15}$$

The extension of this result to the multi-port case is straightforward [109]. The Hybrid matrix associated to a multi-port transformer, having R secondary ports and P primary ports, reads

$$\begin{bmatrix} \boldsymbol{i}_r \\ \boldsymbol{v}_p \end{bmatrix} = \underbrace{\begin{bmatrix} \mathbf{0} & -\mathbf{N} \\ \mathbf{N}^T & \mathbf{0} \end{bmatrix}}_{\mathbf{H}_t} \begin{bmatrix} \boldsymbol{v}_r \\ \boldsymbol{i}_p \end{bmatrix} \tag{4.16}$$

where $\mathbf{N} \in \mathbb{R}^{R \times P}$ collects the turns ratio of the multi-port ideal transformer. Using the results in Table 4.1 trivially confirms that $\mathbf{H}_t + \mathbf{H}_t^T = \mathbf{0}$, i.e. the multi-port ideal transformer is lossless.

Belevitch demonstrated in [109] that the synthesis of a passive reciprocal impedance (admittance) matrix is equivalent to the synthesis of an ideal multi-port transformer whose secondary ports are closed on positive conductances (resistors). Define $\mathbf{G}_d = \mathrm{diag}(G_{d_i}) > 0$ as a matrix collecting some positive conductances G_{d_i} on its main diagonal, with $i = 1, \ldots, R$; closing the R secondary ports of the ideal multi-port transformer described by equation (4.16) on $\mathbf{G}_d$ imposes the relation

$$\boldsymbol{i}_r = -\mathbf{G}_d \boldsymbol{v}_r \,, \tag{4.17}$$

whose substitution in (4.16) reads

$$\begin{bmatrix} \mathbf{0} \\ \boldsymbol{v}_p \end{bmatrix} = \begin{bmatrix} \mathbf{G}_d & -\mathbf{N} \\ \mathbf{N}^T & \mathbf{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{v}_r \\ \boldsymbol{i}_p \end{bmatrix} . \tag{4.18}$$

Since $\mathbf{G}_d$ has full rank, elimination of $\boldsymbol{v}_r$ from (4.18) leads to $\boldsymbol{v}_p = \mathbf{Z}\boldsymbol{i}_p$, where

$$\mathbf{Z} = \mathbf{N}^T \mathbf{G}_d^{-1} \mathbf{N} \,, \tag{4.19}$$

and the dimension R of $\mathbf{G}_d$ is the rank of $\mathbf{Z}$. An equivalent result to (4.19) holds for the admittance matrix case terminating the P ports on positive resistors $\mathbf{R}_d$

$$\mathbf{Y} = \mathbf{N} \mathbf{G}_d \mathbf{N}^T \,. \tag{4.20}$$

From a mathematical perspective, equation (4.19) states that the synthesis of a symmetric positive definite impedance matrix $\mathbf{Z} \in \mathbb{R}^{P \times P}$ is equivalent to the synthesis of a multi-port ideal transformer with turns ratio matrix $\mathbf{N} \in \mathbb{R}^{R \times P}$ and secondary ports closed on the R positive resistors associated to the diagonal elements of $\mathbf{G}_d^{-1}$. Therefore the synthesis problem is now equivalent to a matrix decomposition. The most convenient decomposition for full-rank symmetric positive definite matrices is the LDL decomposition [111] (basically an extension of the widespread Cholesky factorization), while for the low-rank case a modification of LDL decomposition is preferred [168].

Once the decomposition (4.19) is performed on $\mathbf{Z}$, the synthesis is straightforward: the turns ratio matrix $\mathbf{N}$ associated to the multi-port ideal transformer is synthesized using controlled sources as explained in Section 4.2.1 (like in Figure 4.5), while the positive elements on the main diagonal of $\mathbf{G}_d^{-1}$ are the resistors closing the R secondary ports of the ideal multi-port transformer. Note that in the case of low-rank impedance or admittance matrices, R is less than P.

The RT (Resistors and ideal Transformers) synthesis resulting from the application of (4.19) is noise compliant by construction. Indeed the multi-port ideal transformer, synthesized using controlled sources, results into a lossless network that does not affect the noise produced by the positive resistors placed at its secondary ports. Since the only noisy elements involved in the RT synthesis are positive resistors, the synthesis is noise compliant. This result was verified in practice relying on the automated noise testing procedure discussed in Section 4.2.4.

The RT synthesis is now applied to the admittance matrix $\mathbf{G}_{e2}$ from (4.12). For this example the pure resistive synthesis with fixed topology failed in Section 4.2.2 due to the need of negative resistors. According to the matrix decomposition (4.20), the LDL factorization is performed on $\mathbf{G}_{e2}$ leading to

$$\mathbf{G}_{e2} = \underbrace{\begin{bmatrix} 1 & 0 \\ -\frac{5}{9} & 1 \end{bmatrix}}_{\mathbf{N}} \underbrace{\begin{bmatrix} 9 & 0 \\ 0 & \frac{11}{9} \end{bmatrix}}_{\mathbf{G}_d} \underbrace{\begin{bmatrix} 1 & -\frac{5}{9} \\ 0 & 1 \end{bmatrix}}_{\mathbf{N}^T} \tag{4.21}$$

where the contributions from (4.20) are highlighted. The resulting synthesis for this example is depicted in Figure 4.9. Noise compliance for this simple case can be verified analytically relying on the results of Section 4.1 or using a Spice simulation, as in Section 4.2.4. An automated procedure for the validation of RT synthesis is discussed in Section 4.2.4.

4.2.4 Static synthesis results

The automated test procedure described in this section is focused on

- Synthesis accuracy: DC analysis results are compared using the admittance matrix (similar results hold for the impedance) and the associated RT synthesis, both in Spice. Of particular relevance are the cases in which the admittance matrix is rank deficient, because for such cases a low-rank matrix decomposition is used.

- Noise compliance: like for the DC analysis, the admittance matrix and the resulting RT synthesis are used in Spice to perform the static noise analysis.

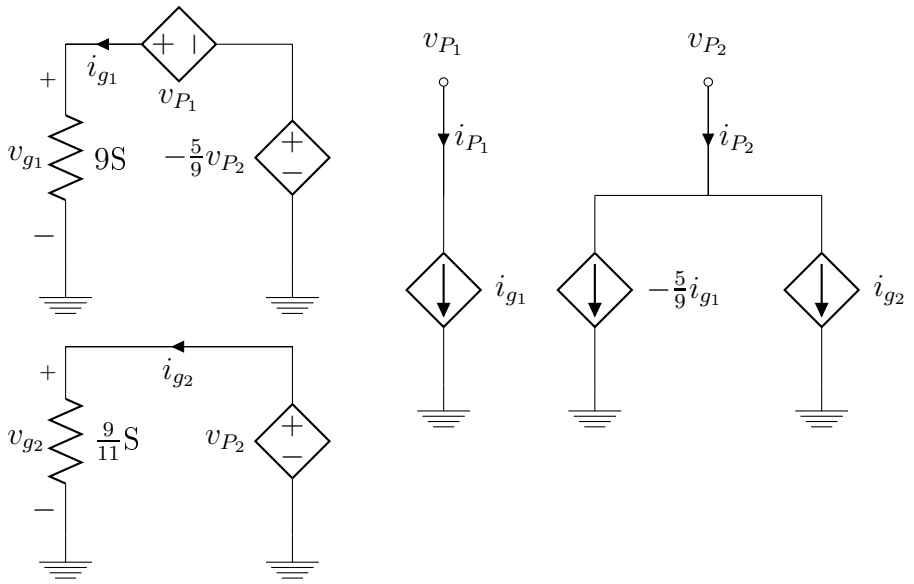

Figure 4.9: Resistors ideal Transformers (RT) synthesis for the admittance matrix $\mathbf{G}e_2$ based on the LDL factorization (4.21).

For the sake of completeness several test cases are needed. Since the availability of many real test cases is not feasible in a short amount of time, an automated procedure for the creation of semi-positive definite impedance or admittance matrices was created based on the following methodologies:

- random matrix: starting from a pseudo random matrix, reciprocity and positive definitiveness are imposed (acting directly on the eigenvalues of the symmetrized random matrix);
- greatest common divisor matrices: obtained from Matlab®'s function gallery (option: *gcdmat*). Those matrices are symmetric positive semidefinite by construction;
- symmetric, ill-conditioned Toeplitz matrices: obtained from Matlab®'s function gallery (options: *prolate* and $w \in [0, 0.5]$). Those matrices are low-rank and symmetric positive definite.

The number of ports for each case is selected randomly within a specified range, i.e. $P \in [1, 50]$.

Two are the main steps of the proposed automated procedure

1. DC analysis: using unitary inputs, in accordance to the matrix representation used, i.e. impedance or admittance, the DC analysis is performed in Spice using the impedance/admittance matrix and the

Table 4.2: Selection of results obtained using the proposed automatic procedure. ϵ_{DC} is the error on the DC analysis defined by (4.22), while max ϵ_n is the maximum on all the errors ϵ_n, defined by (4.23), obtained from the static noise analysis related to a specific test case.

Test	P	ϵ_{DC}	max ϵ_n
1	4	1e-12	1e-13
2	8	2e-12	1e-13
3	20	8e-13	2e-13

associated Belevitch's (RT) synthesis. The results of the DC analysis based on the direct usage of the admittance/impedance matrix are denoted by $\boldsymbol{y}_d$, while the results obtained from the synthesized netlist are $\boldsymbol{y}_b$. The error metric used for the DC analysis is

$$\epsilon_{DC} = |\boldsymbol{y}_d - \boldsymbol{y}_b| \ ; \tag{4.22}$$

2. Static noise analysis: considering all the possible combinations of input-output ports, voltage and current spectral densities are evaluated in Spice, using the impedance/admittance matrix and the associated Belevitch's (RT) synthesis. The results of the noise analysis obtained from Spice using the admittance/impedance matrix are denoted by ν_d, while the results obtained from the synthesized netlist are ν_b. The error metric used for the DC analysis is

$$\epsilon_n = \frac{|\nu_d - \nu_b|}{|\nu_d|} \, , \tag{4.23}$$

where the fraction is always well posed since noiseless cases are not considered, i.e. $\nu_d \neq 0$ by construction.

A small subset of the results obtained with the automatic verification strategy are proposed here listed in Table 4.2. Similar results were obtained on a large set of more than one thousand test cases.

4.3 Dynamic network synthesis

Several techniques are available for the synthesis of dynamic networks associated to state-space models [119]. The three most efficient and widespread methods are: direct state-space procedure, described in Section 4.3.2, that is based on the immediate conversion of state-space equations into a circuit [169]; Darlington's resistance extraction, presented in 4.3.4 which is an

extension of Resistors ideal Transformers (RT) static synthesis from Section 4.2; Youla's reactance extraction [154], discussed in Section 4.3.3, which is a clever reformulation of Darlington's resistance extraction. Results and complexity tables are summarized in Section 4.3.5 for all synthesis methods, described below.

The main contributions of this chapter are: presentation of well known synthesis methods focusing on complexity and noise compliance in order to clarify a topic not adequately covered in classic books of network theory; and thus introduction of noise preserving sparsification techniques for each synthesis method aimed at reducing the complexity of the synthesis while preserving the desired physical properties.

Before describing in more details the synthesis of dynamic networks, some preliminary results concerning state-space models of passive devices are summarized in Section 4.3.1.

4.3.1 Preliminaries on state-space models

Some basic results concerning state-space models are summarized here since they are needed for the presentation of dynamic synthesis methods in Sections 4.3.2 4.3.3 and 4.3.4. In this section minimal state-space realizations and their canonical forms are discussed, together with reciprocity and passivity conditions.

As already mentioned in Section 1.3, it is common practice to model reciprocal passive circuit blocks via frequency dependent network parameters (scattering or hybrid). Using standard techniques, like those presented in Chapter 2, a mathematical model is extracted and converted to state-space form (2.46), repeated here for reader convenience

$$\dot{\boldsymbol{x}}(t) = \mathbf{A}\boldsymbol{x}(t) + \mathbf{B}\boldsymbol{u}(t), \tag{4.24}$$

$$\boldsymbol{y}(t) = \mathbf{C}\boldsymbol{x}(t) + \mathbf{D}\boldsymbol{u}(t), \tag{4.25}$$

with $\mathbf{A} \in \mathbb{R}^{N\times N}$, $\mathbf{B} \in \mathbb{R}^{N\times P}$, $\mathbf{C} \in \mathbb{R}^{P\times N}$ and $\mathbf{D} \in \mathbb{R}^{P\times P}$. This is the starting point for the synthesis algorithms considered in this Chapter.

In the following, models are assumed without loss of generality to be in the impedance input-output representation, i.e.

$$\mathbf{Z}(s) = \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B} + \mathbf{D} \leftrightarrow \left(\begin{array}{c|c} \mathbf{A} & \mathbf{B} \\ \hline \mathbf{C} & \mathbf{D} \end{array}\right), \tag{4.26}$$

where s is the complex frequency (Laplace) variable.

It is worth noting that the complexity of dynamic network synthesis is directly related with the dynamic order of the model. In particular, a necessary but not sufficient condition to have a canonical[3] synthesis is that the state-space model must have McMillan degree [119] of $\mathbf{Z}(s)$ equal to N (the size of $\mathbf{A}$). This is equivalent to state that the state-space realization (4.26) is minimal, i.e. the system is both controllable and observable.

Besides minimality, also the structure of state-space matrices has a direct impact on the number of elements required by the synthesis. State-space realizations are not unique. Two minimal state-space realizations of the same system

$$\left(\begin{array}{c|c} \mathbf{A} & \mathbf{B} \\ \hline \mathbf{C} & \mathbf{D} \end{array}\right) \leftrightarrow \left(\begin{array}{c|c} \tilde{\mathbf{A}} & \tilde{\mathbf{B}} \\ \hline \tilde{\mathbf{C}} & \tilde{\mathbf{D}} \end{array}\right) \tag{4.27}$$

are equivalent to each other through a change of basis in the state space [119], applied though a similarity transformation as

$$\tilde{\mathbf{A}} = \mathbf{T}^{-1}\mathbf{A}\mathbf{T}, \qquad \tilde{\mathbf{B}} = \mathbf{T}^{-1}\mathbf{B}, \tag{4.28}$$

$$\tilde{\mathbf{C}} = \mathbf{C}\mathbf{T}, \qquad \tilde{\mathbf{D}} = \mathbf{D}, \tag{4.29}$$

with $\mathbf{T} \in \mathbb{R}^{N\times N}$ invertible. In particular, Gilbert in [170] proposed a minimal state-space realization that is relevant for the direct synthesis discussed in Section 4.3. Detail on how to construct Gilbert's realization can be found in [126]. It is worth noting here that such realization presents a sparse state-space matrix $\mathbf{A}$ with the following block-diagonal structure

$$\mathbf{A}_{i,j} = \begin{cases} p_c, & i = j = c \\ \begin{bmatrix} \mathbb{R}\mathrm{e}\{p_l\} & \mathbb{I}\mathrm{m}\{p_l\} \\ -\mathbb{I}\mathrm{m}\{p_l\} & \mathbb{R}\mathrm{e}\{p_l\} \end{bmatrix} & i = l, l+1 \quad j = l, l+1 \end{cases} \tag{4.30}$$

where $\{p_l, p_c\} \in \lambda(\mathbf{A})$ are the eigenvalues (real poles p_c, and complex poles p_l) associated to the minimal state-space model. Note that Gilbert's realization can be derived iff all the eigenvalues $\lambda(\mathbf{A})$ have algebraic multiplicity one [126]. This condition is imposed by construction in the identification methods described in Chapter 2.

Besides the complexity constraint, also physical constraints must be taken into consideration when dealing with the state-space model associated to a real circuit. It is a well known result of network theory [161] that all RLCT linear networks are reciprocal. As a consequence reciprocity is a physical

[3]Note that Anderson in [119] demonstrated that it is not possible to synthesize a reciprocal dynamic network attaining both the minimum number of dynamic and resistive elements.

property of interest that should be preserved by a well posed synthesis process. Such a property has a direct impact on the structure of the state-space model. A square system (4.24)-(4.25) is reciprocal with respect to the signature matrix (4.6) if and only if [171] its hybrid transfer function $\mathbf{H}(s)$ is sign symmetric with respect to $\mathbf{S}_{ext}$ (4.6), i.e.

$$\mathbf{S}_{ext}\mathbf{H}(s) = \mathbf{H}(s)^T\mathbf{S}_{ext}\,. \tag{4.31}$$

For the impedance (admittance) case condition (4.31) simplifies to

$$\mathbf{Z}(s) = \mathbf{Z}(s)^T\,. \tag{4.32}$$

It is clear that this is a straightforward extension of the definition of reciprocity that was given for the static case in Section 4.2.1. In addition to conditions (4.31)-(4.32), it can be demonstrated [171] that for reciprocal state-space models there exists a symmetric matrix $\mathbf{\Pi} = \mathbf{\Pi}^T \in \mathbb{R}^{N\times N}$ such that

$$\left\{\begin{array}{rcl} \mathbf{A\Pi} & = & \mathbf{\Pi A}^T, \\ \mathbf{B} & = & \mathbf{\Pi C}^T\mathbf{S}_{ext}. \end{array}\right. \tag{4.33}$$

If $\mathbf{\Pi} = \mathbf{S}_{int}$ with $\mathbf{S}_{int} = \text{blkdiag}(\mathbf{I}_{N_l}, -\mathbf{I}_{N_c}) \in \mathbb{R}^{N\times N}$ internal signature matrix, then the state-space model (4.25) is called internally reciprocal. Two strategies are available to evaluate matrix $\mathbf{\Pi}$ (4.33) for a reciprocal state-space model: solving directly the system (4.33) via optimization packages like SeDuMi [124], or by defining the similarity transformation matrix $\mathbf{T}$ (4.28) relating the state-space model and its transposed [119].

Another physical constraint of paramount importance for linear models (4.25) is passivity. The general frequency-domain passivity conditions discussed in Section 2.5 can be transformed into purely algebraic ones [103] called Positive Real Lemma (hybrid case) and Bounded Real Lemma (scattering case), also know as Kalman-Yakubovich-Popov (KYP) lemma. For the sake of clarity and reader convenience passivity and reciprocity conditions are summarized in Table 4.3. A real square state-space model (4.25)-(4.24), based on hybrid or scattering representation, is passive if and only if the corresponding Linear Matrix Inequality (LMI) in the column *Passive* admits a symmetric positive definite solution matrix $\mathbf{P}$. A similar result holds for reciprocity considering the column *Reciprocal* and the solution matrix $\mathbf{\Pi}$. As it will be more clear in (4.3.3) and (4.3.4), different solution matrices $\mathbf{\Pi}$ and $\mathbf{P}$ obtained for the condition in Table 4.3 will lead to different Youla's and Darlington's syntheses.

In the following sections the synthesis algorithms will be introduced together with simple explanatory examples. Results and a comparative study of complexity for all the synthesis methods can be found in Section (4.3.5).

Table 4.3: Summary of passivity and reciprocity conditions for dynamic networks representations. In the passivity conditions equality to zero means that the network is lossless, i.e. does not absorb instantaneous/active power.

System	Reciprocal	Passive
$\underset{hybrid}{\mathbf{H}(s)}$	$\begin{cases} \mathbf{A\Pi} = \mathbf{\Pi A}^T \\ \mathbf{B} = \mathbf{\Pi C}^T\mathbf{S}_{ext} \end{cases}$	$\begin{bmatrix} \mathbf{A}^T\mathbf{P}+\mathbf{PA} & \mathbf{PB}-\mathbf{C}^T \\ \mathbf{B}^T\mathbf{P}-\mathbf{C} & -\mathbf{D}-\mathbf{D}^T \end{bmatrix} \preceq 0$
$\underset{scattering}{\mathbf{S}(s)}$	$\begin{cases} \mathbf{A\Pi} = \mathbf{\Pi A}^T \\ \mathbf{B} = \mathbf{\Pi C}^T \end{cases}$	$\begin{bmatrix} \mathbf{A}^T\mathbf{P}+\mathbf{PA}+\mathbf{C}^T\mathbf{C} & \mathbf{PB}+\mathbf{C}^T\mathbf{D} \\ \mathbf{B}^T\mathbf{P}+\mathbf{D}^T\mathbf{C} & \mathbf{D}^T\mathbf{D}-\mathbf{I}_P \end{bmatrix} \preceq 0$
with	$\mathbf{\Pi} = \mathbf{\Pi}^T \in \mathbb{R}^{N\times N}$	$\mathbf{P} = \mathbf{P}^T \succ 0, \mathbf{P} \in \mathbb{R}^{N\times N}$
internal if	$\mathbf{\Pi} = \mathbf{S}_{int} = \begin{bmatrix} \mathbf{I}_{N_l} & \mathbf{0} \\ \mathbf{0} & -\mathbf{I}_{N_c} \end{bmatrix}$	$\mathbf{P} = \mathbf{I}_N$

4.3.2 Direct state-space synthesis

The dynamic synthesis method presented in this section can be considered as the extension of the direct synthesis method introduced in Section 4.2.1 for the static case. The basic idea relies on a mere "translation" of the state-space equations into an equivalent electric network. It was proposed by [169] in the context of Model Order Reduction techniques in order to obtain a low-complexity synthesis, in terms of number of primitive network elements, for reduced order state-space models. As such this synthesis algorithm found a great diffusion and it is commonly used in practice.

Beside the low complexity feature, another major benefit is the simplicity of the algorithm itself. Indeed, considering the state-space model associated to an impedance matrix

$$\dot{\boldsymbol{x}}(t) = \mathbf{A}\boldsymbol{x}(t) + \mathbf{B}\boldsymbol{i}(t)\,, \tag{4.34}$$

$$\boldsymbol{v}(t) = \mathbf{C}\boldsymbol{x}(t) + \mathbf{D}\boldsymbol{i}(t)\,, \tag{4.35}$$

with input vector $\boldsymbol{i} \in \mathbb{R}^P$ and output vector $\boldsymbol{v} \in \mathbb{R}^P$, a direct conversion into an equivalent network is straightforward and can be divided in two main steps

1. state synthesis: consider a single row l of (4.34)

$$\dot{x}_l(t) = \sum_{j=1}^{N} a_{l,j}x_j(t) + \sum_{k=1}^{P} b_{l,k}i_k(t)\,. \tag{4.36}$$

Collecting on the left side of (4.36) all the elements related with the state $x_l(t)$ it follows

$$\dot{x}_l(t) - a_{l,l}x_l(t) = \sum_{\substack{j=1 \\ j\neq l}}^{N} a_{l,j}x_j(t) + \sum_{k=1}^{P} b_{l,k}i_k(t)\,. \tag{4.37}$$

The left-hand side of (4.37) can be seen as the parallel of a unitary capacitor and a resistor of value $-1/a_{l,l}$ to whom a voltage $x_l(t)$ is applied. The resulting current must equal the right-hand side of (4.37), which can then be interpreted as the parallel interconnection of controlled current sources, where the control variables are the states $x_j(t)$ and the inputs $i_k(t)$. The sub-network associated to (4.37) is depicted in Figure (4.10). Please note that resistors $-1/a_{l,l}\,\forall\, l$ will never be negative[4], since the poles of (4.34) are in $\mathbb{R}^-$ by construction as a consequence of model stability [62];

2. output equation synthesis: in a similar fashion to the previous synthesis step, consider a single row m of (4.35), i.e.

$$v_m(t) = \sum_{j=1}^{N} c_{m,j}x_j + \sum_{k=1}^{P} d_{m,k}i_k(t) \tag{4.38}$$

with $m, k = 1, \ldots, P$ and $j = 1, \ldots, N$. Since the output of (4.35) has voltage units, the right side of (4.38) is equivalent to a series interconnection of controlled voltage sources, where the control variables are the states $x_j(t)$ and the inputs $i_k(t)$. The sub-network associated to (4.38) is depicted in Figure (4.11).

Performing the two steps above for each row of (4.34) and (4.35) concludes the synthesis process. Since the network elements involved in the synthesis are only Resistors, Capacitors and Controlled Sources, in the following this synthesis will be also defined as RCCS.

For what concerns the complexity of the resulting network, from equations (4.37) (4.38) it is clear how the number of elements used by the RCCS synthesis is directly related to the structure of the state-space matrices in (4.34)-(4.35). Consequently a low complexity synthesis can be obtained converting a generic state-space realization into the Gilbert canonical form that was introduced in 4.3.1. This way the number of primitive network elements will scale as $\mathcal{O}(NP^2)$.

[4]It is always possible to transform the negative definite state-space matrix $\mathbf{A}$ into an equivalent one having negative elements on the main diagonal.

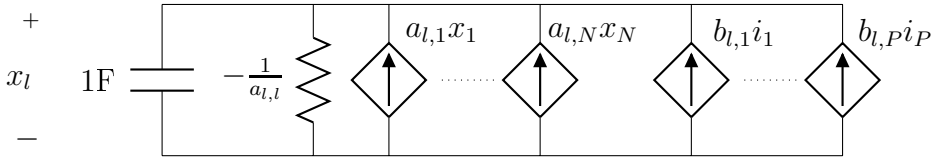

Figure 4.10: Direct states synthesis based on equation (4.37).

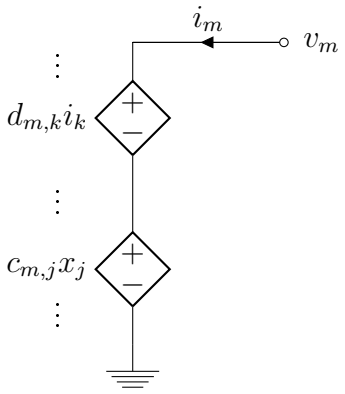

Figure 4.11: Direct ports synthesis based on equation (4.38).

The main drawback of this method is the lack of noise-compliance. As it was discussed in Section 4.2.1 for the static synthesis case, the use of controlled sources in the synthesis process demands particular care, since those components are not equipped with a noise model in standard Spice-based solvers. A simple example in the following section will further clarify this issue.

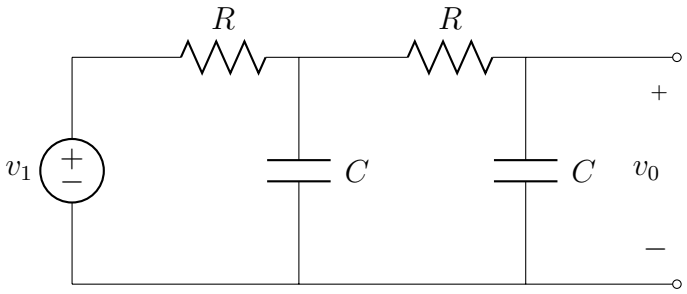

Figure 4.12: RC ladder circuit.

Noise compliance issue

In order to further clarify the lack of noise compliance in the direct synthesis method presented in Section 4.3.2, the simple RC ladder network depicted in Figure 4.12 will be analysed here. The state-space model associated to the circuit in Figure 4.12 will be synthesized using the RCCS algorithm, then the analytic expression of the output noise spectral density will be evaluated for both circuits by means of standard techniques [172]. Moreover, noise analysis results will be compared with Spice simulations for the sake of completeness.

First, the analytic expression of the output voltage spectral density is considered for the RC ladder network in Figure 4.12. As explained in Section 4.1, noise analysis is based on the substitution of noisy resistors with the equivalent Norton/Thevenin noise model; the resulting network is depicted in Figure 4.13. Considering now the KCL at two nodes results

$$\begin{cases} G(V_1 - V_{n_1}) + sCV_1 + G(V_1 + V_{n_2} - V_o) = 0 \\ sCV_o + G[V_o - (V_1 + V_{n_2})] = 0 \end{cases} \tag{4.39}$$

From the second row it follows

$$(G + sC)V_o - GV_{n_2} = GV_1 \tag{4.40}$$

leading to

$$(1 + sRC)V_o - V_{n_2} = V_1. \tag{4.41}$$

Substituting now (4.41) in the second row of (4.39) after simple algebraic calculations reads

$$(1 + sRC + 2sRC + s^2R^2C^2)V_o - V_{n_1} - (1 + sRC)V_{n_2} = 0. \tag{4.42}$$

Highlighting now the contributions from each noise source it follows

$$V_o = \frac{V_{n_1}}{1 + 3RCs + s^2R^2C^2} + \frac{V_{n_2}(1 + sRC)}{1 + 3RCs + s^2R^2C^2}. \tag{4.43}$$

with V_{n_1} and V_{n_2} statistically independent. The analytical expression for the voltage noise spectral density is obtained after straightforward calculations substituting (4.1) in (4.43) and considering the Root Mean Square (RMS) value

$$\bar{V}_o^2(\omega) = \frac{2 + (\omega RC)^2}{1 + 7(\omega RC)^2 + (\omega RC)^4}. \tag{4.44}$$

This result can be verified analytically considering that, by Nyquist theorem (4.1)

$$\bar{V}_o^2(\omega) = 4K_bT\mathbb{R}\mathrm{e}\{Z_{out}(\omega)\}. \tag{4.45}$$

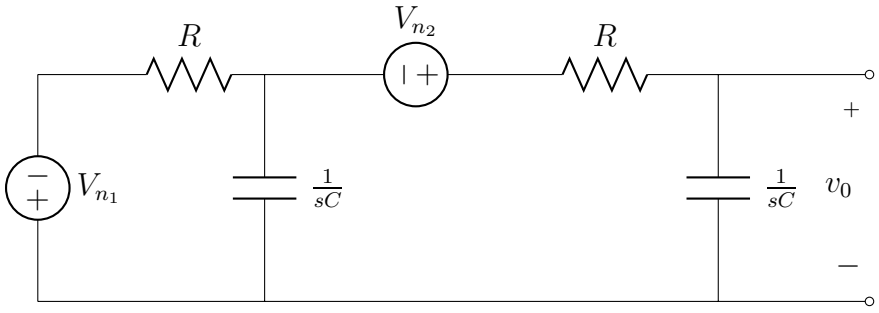

Figure 4.13: RC ladder circuit for the noise analysis in the Laplace domain: V_n are the noise equivalent voltage sources (the sign is arbitrary) and R are noiseless resistors.

In fact, from the circuit in Figure 4.12, the expression of Z_{out} can be easily obtained as

$$Z_{out}(s) = \frac{\frac{1}{sC}\frac{R(2+sCR)}{1+sCR}}{\frac{1}{sC} + \frac{R(2+sCR)}{1+sCR}} = \frac{R(2+sCR)}{1+3RCs+(RCs)^2}\,, \tag{4.46}$$

which matches (4.44)-(4.45) as expected. The real part of Z_{out} can be found considering that $\mathbb{R}\mathrm{e}\{Z_{out}(s)\} = 0.5(Z_{out}(s) + Z_{out}(-s))$, i.e.

$$\mathbb{R}\mathrm{e}\{Z_{out}(s)\} = R\frac{2-(RCs)^2}{[1+(RCs)^2]^2-(3RCs)^2}. \tag{4.47}$$

Equation (4.44) can be verified performing the noise analysis in Spice for the network in Figure 4.12. The results depicted in Figure 4.14 match to machine precision.

Now, a state-space model is derived for the RC ladder network and then synthesized back to a circuit whose spectral noise density is evaluated analytically. From standard calculations[5] the state-space model, normalized by C, in Gilbert canonical form for the RC ladder network in Figure 4.12 results in

$$\mathbf{A} = \frac{1}{R}\begin{pmatrix} p_1 & 0 \\ 0 & p_2 \end{pmatrix}\,, \quad \boldsymbol{b} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}\,, \quad \boldsymbol{c} = \begin{pmatrix} \rho_1 \\ \rho_2 \end{pmatrix}^T\,, \tag{4.48}$$

corresponding to

$$Z(s) = \frac{R\rho_1}{sCR-p_1} + \frac{R\rho_2}{sCR-p_2}\,, \tag{4.49}$$

[5]For a detailed derivation of the Gilbert canonical state-space form associated to the RC ladder network example see Appendix C.

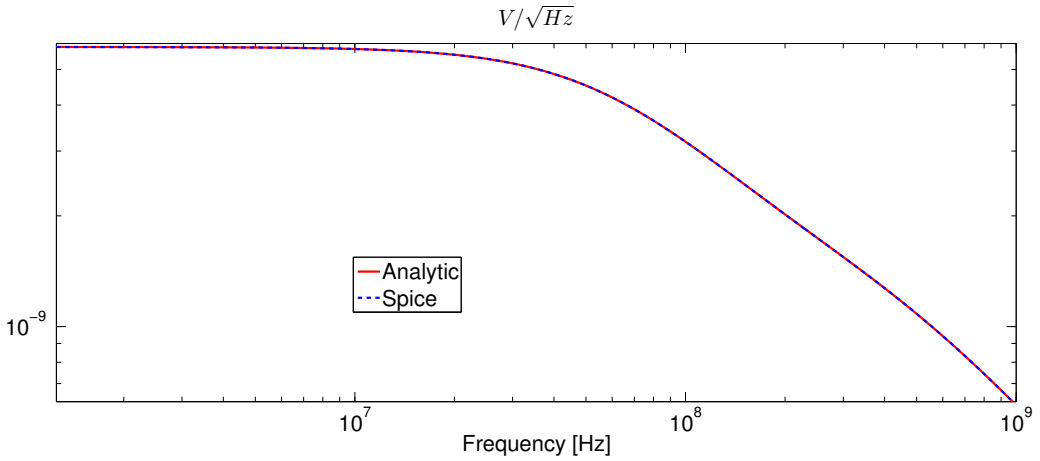

Figure 4.14: The result of the noise analysis from Spice performed using the circuit in Figure 4.12 (blue dashed line) is compared with the direct application of (4.44) (red continuous line).

where $p_{1,2}$ are

$$p_{1,2} = \frac{-3 \pm \sqrt{9-4}}{4} = \frac{-3 \pm \sqrt{5}}{2} \tag{4.50}$$

and

$$\rho_{1,2} = \frac{5 \pm \sqrt{5}}{10}\,. \tag{4.51}$$

Applying now the direct synthesis method to the state-space model (4.48) leads to the network depicted in Figure 4.15. Like for the RC ladder circuit, also for the network resulting from the direct synthesis of the Gilbert canonical model the analytic expression describing the output spectral noise density can be derived using the noise analysis method described in Section 4.1. Considering that the mean square voltage noise associated to each resistor in Figure 4.16 is

$$\bar{v}_{n_i}^2 = 4K_bT\left(-\frac{R}{p_i}\right), \quad p_i \in \mathbb{R}^-, \tag{4.52}$$

it follows that the mean square voltage noise $\bar{v}_{r_i}^2$ from each sub-circuit in Figure 4.16 reads

$$\bar{v}_{r_i}^2 = \bar{v}_{n_i}^2\left(\frac{p_i}{p_i + RCs}\right)^2. \tag{4.53}$$

Voltage controlled current sources do not affect the noise, therefore the noise contributions $\bar{v}_{r_i}^2$ lead to the mean square voltage output by means of the weighted sum

$$\bar{v}_o^2 = \rho_1^2\bar{v}_{r_1}^2 + \rho_2^2\bar{v}_{r_2}^2. \tag{4.54}$$

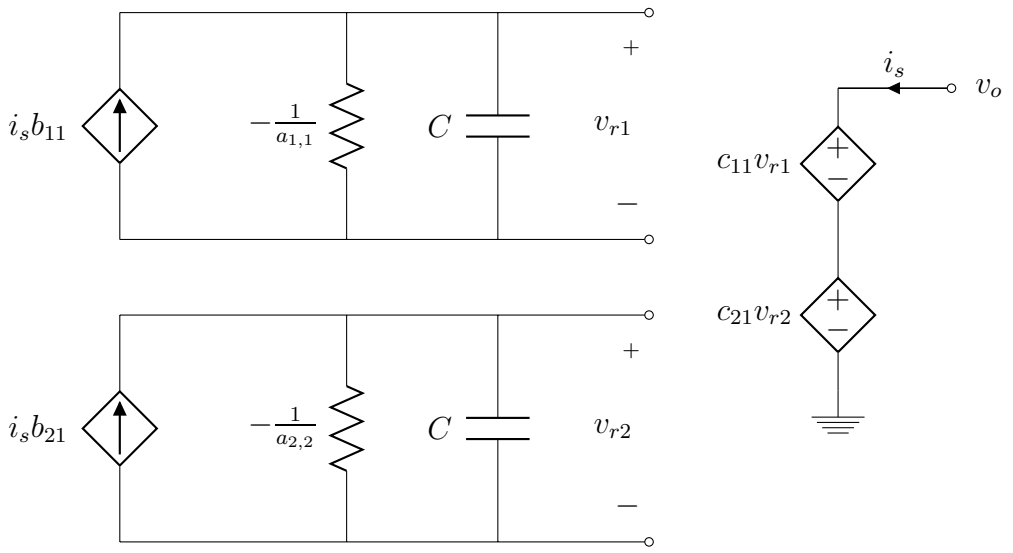

Figure 4.15: Direct synthesis from Gilbert canonical form (4.48). Note that the values of the resistors are positive since the poles p_i must be negative to describe a stable system.

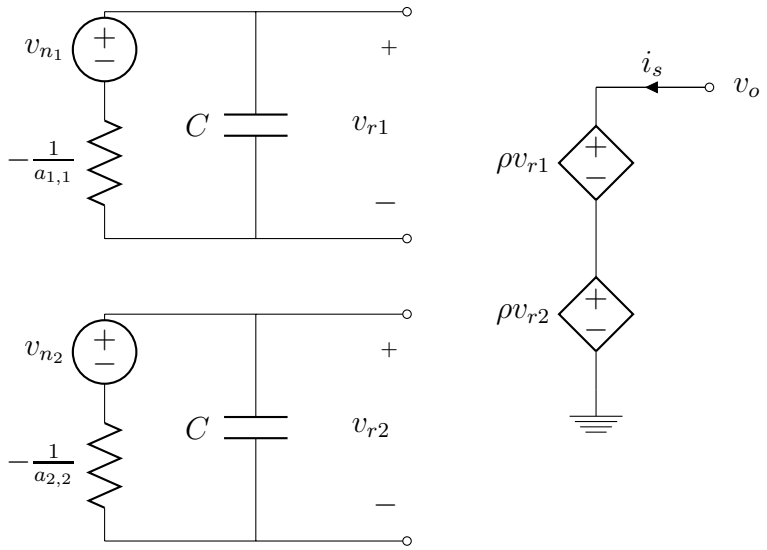

Figure 4.16: Noise analysis network from Gilbert canonical form (4.48) synthesis.

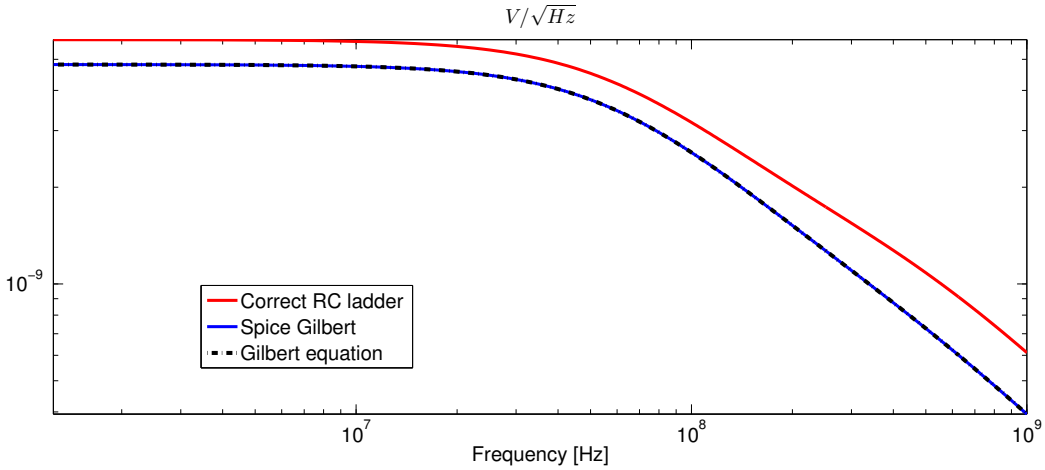

Figure 4.17: Noise analysis results comparison: the red continuous line is the result from equation (4.44) (previously verified via Spice), while the black dashed line is the result from equation (4.55), verified via Spice using the network in Figure 4.15 (blue continuous line).

Substituting (4.53) and (4.52) into (4.54) leads to

$$\bar{V}_o^2(\omega) = 4K_bTR\left(-\frac{\rho_1^2 p_1}{\omega^2R^2C^2+p_1^2} - \frac{\rho_2^2 p_2}{\omega^2R^2C^2+p_2^2}\right). \qquad (4.55)$$

The two analytical expressions for the output spectral voltage noise derived so far are compared graphically in Figure 4.17. As can be seen, equation (4.55), which describes the spectral noise associated to the synthesised network, matches the result from Spice noise analysis. This confirms that equation (4.55) describes properly the noise response of the circuit in Figure 4.15, but the two curves do not match the result from (4.44) (previously verified via Spice), which gives the output voltage noise spectral density of the original RC ladder network (Figure 4.12). Since the state-space model is correct, see Appendix C, this result confirms that the direct synthesis method is not able to preserve the noise response of the original circuit, i.e. the RC ladder network.

The noise compliance issue described in this section is the consequence of the improper usage of controlled sources in the synthesis process. In the following sections two noise-compliant synthesis methods will be presented. In both cases the root idea that allows to preserve the noise behaviour is quite simple, and inherited from the static synthesis case: controlled sources can only be used to construct lossless sub-networks; in this way controlled sources will have no impact on the noise produced by resistors (the only

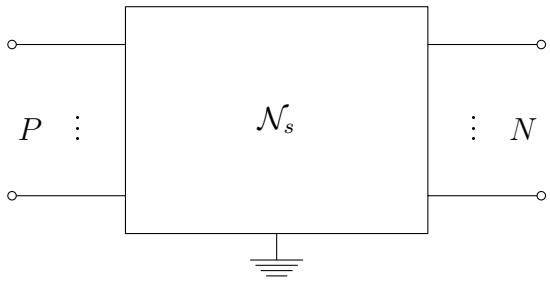

Figure 4.18: Generic $(P+N)$-port network $\mathcal{N}_s$.

elements responsible for Gaussian noise in a linear circuit, as explained in Section 4.1.).

4.3.3 Youla's reactance extraction

To overcome the lack of noise compliance in the RCCS synthesis, the classic reactance extraction algorithm, originally proposed by Youla and Tissi in [154], can be used. As for the RT synthesis in the static case, also Youla's synthesis is noise compliant by construction, since controlled sources are only used to construct lossless sub-networks. Moreover, reciprocity constraints are also imposed, in order to avoid the usage of non-reciprocal elements (gyrators), preserving all the physical properties of linear time-invariant lumped network. As a preliminary step, the inductance extraction procedures is briefly outlined in order to simplify the presentation of Youla's synthesis.

Inductance extraction

The inductance extraction idea can be easily introduced considering a static $(P+N)$-port network $\mathcal{N}_s$ depicted in Figure 4.18. Supposing that the impedance matrix representation exists for this network, voltages and currents can be related by

$$\boldsymbol{v}_p = \mathbf{Z}_{1,1}\boldsymbol{i}_p + \mathbf{Z}_{1,2}\boldsymbol{i}_n\,, \tag{4.56}$$

$$\boldsymbol{v}_n = \mathbf{Z}_{2,1}\boldsymbol{i}_p + \mathbf{Z}_{2,2}\boldsymbol{i}_n\,, \tag{4.57}$$

with $\mathbf{Z}_{1,1} \in \mathbb{R}^{P\times P}$, $\mathbf{Z}_{1,2} \in \mathbb{R}^{P\times N}$, $\mathbf{Z}_{2,1} \in \mathbb{R}^{N\times P}$ and $\mathbf{Z}_{2,2} \in \mathbb{R}^{N\times N}$. It is well known that closing N ports of network $\mathcal{N}_s$ on unitary inductors imposes the relation

$$\boldsymbol{v}_n = -\mathbf{Z}_L\boldsymbol{i}_n\,, \tag{4.58}$$

where $\mathbf{Z}_L = s\mathbf{I}_{N_l}$. Substituting (4.58) into (4.57) and solving for $\boldsymbol{v}_p$ leads to the input-output relation

$$\boldsymbol{v}_p = (\mathbf{Z}_{1,1} - \mathbf{Z}_{1,2}(\mathbf{Z}_L + \mathbf{Z}_{2,2})^{-1}\mathbf{Z}_{2,1})\boldsymbol{i}_p\,. \tag{4.59}$$

Note that (4.59) has a remarkable similarity with (4.26), repeated here for convenience

$$\boldsymbol{v}_p = (\mathbf{D} + \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B})\boldsymbol{i}_p\,. \tag{4.60}$$

From a direct comparison, it follows that the state space realization $\{\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}\}$ of a P-port system can be synthesised as a static $(P+N)$-port impedance network (4.56)-(4.57) with

- $\mathbf{Z}_{1,1} = \mathbf{D} \in \mathbb{R}^{P\times P}$;
- $\mathbf{Z}_{1,2} = -\mathbf{C} \in \mathbb{R}^{P\times N}$;
- $\mathbf{Z}_{2,1} = \mathbf{B} \in \mathbb{R}^{N\times P}$;
- $\mathbf{Z}_{2,2} = -\mathbf{A} \in \mathbb{R}^{N\times N}$,

and with its last N ports closed on unitary inductors $\mathbf{I}_{N_l}$.

This simple example suggests that the state-space matrices of the transfer function (4.60) could constitute the impedance matrix of a static network. The extension to the capacitance extraction case is straightforward and well documented [119]. The main question now is how and under which conditions it is possible to construct from state-space matrices (4.60) a passive and symmetric impedance (or reciprocal hybrid matrix, in the general case) matrix (4.56) in a similar fashion to the previous example. Since the state-space matrices describing a dynamic network are not unique (like discussed in Section 4.3.1) the main concern in the following will be to find the similarity transformation (4.28) leading to a state-space model that allows to construct the hybrid matrix associated to a passive, reciprocal static network. This is the main task of Youla's reactance extraction method and the enabling factor for a reciprocal and noise-compliant synthesis. Note that this task is tightly related with passivity and reciprocity conditions outlined in Table 4.3 for a state-space model.

As will be explained in the following, Youla's reactance extraction synthesis requires to transform the state-space matrices of the transfer function (4.60) into a positive real balanced and internally reciprocal realization. Details on how to define and perform such transformations are the main topic in the rest of this section.

Positive real balanced state-space realizations

Starting from the passive state-space realization $\{\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}\}$ associated to the impedance transfer function

$$\mathbf{Z}(s) = \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B} + \mathbf{D} \leftrightarrow \left(\begin{array}{c|c} \mathbf{A} & \mathbf{B} \\ \hline \mathbf{C} & \mathbf{D} \end{array}\right), \tag{4.61}$$

the following steps are required in order to obtain a positive real balanced realization

- step 1: explicitly solve the Positive Real Lemma (PRL) for $\mathbf{P}$, i.e. the LMI in Table 4.3 for the hybrid case. Also form the dual system $\{\mathbf{A}^T, \mathbf{C}^T, \mathbf{B}^T, \mathbf{D}^T\}$ and solve its associated PRL for the corresponding matrix $\mathbf{Q}$. Restricting now the analysis to the case $\mathbf{R} = \mathbf{D} + \mathbf{D}^T \succ 0$ (corresponding to asymptotic strict dissipativity), it follows that the matrices $\mathbf{P}$ and $\mathbf{Q}$ can be found by solving the Continuous Algebraic Riccati Equations (CARE) [173, 174]

$$\mathbf{A}^T\mathbf{P} + \mathbf{P}\mathbf{A} + \left(\mathbf{P}\mathbf{B} - \mathbf{C}^T\right)\mathbf{R}^{-1}\left(\mathbf{B}^T\mathbf{P} - \mathbf{C}\right) = 0, \tag{4.62}$$

$$\mathbf{A}\mathbf{Q} + \mathbf{Q}\mathbf{A}^T + \left(\mathbf{Q}\mathbf{C}^T - \mathbf{B}\right)\mathbf{R}^{-1}\left(\mathbf{C}\mathbf{Q} - \mathbf{B}^T\right) = 0, \tag{4.63}$$

  with $\mathbf{P} = \mathbf{P}^T \succ 0$ and $\mathbf{Q} = \mathbf{Q}^T \succ 0$. This calculation can be performed through the Laub's method [175], based on the evaluation of the invariant subspaces of the Hamiltonian matrices associated to (4.62) and (4.63);

- step 2: compute the Cholesky factorization [110] of $\mathbf{P}$ and $\mathbf{Q}$

$$\mathbf{P} = \mathbf{F}^T\mathbf{F}, \tag{4.64}$$

$$\mathbf{Q} = \mathbf{G}^T\mathbf{G}, \tag{4.65}$$

  with $\mathbf{F}, \mathbf{G} \in \mathbb{R}^{N\times N}$ triangular matrices;

- step 3: apply the Singular Value Decomposition [110] on the matrix product $\mathbf{F}\mathbf{G}^T$, i.e.

$$\mathbf{F}\mathbf{G}^T = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T, \tag{4.66}$$

  with $\mathbf{U}, \mathbf{V} \in \mathbb{R}^{N\times N}$ orthogonal, where the diagonal matrix $\mathbf{\Sigma} \in \mathbb{R}^{N\times N}$ stores the singular values in decreasing order on its main diagonal;

- step 4: construct the invertible similarity transformation matrix $\mathbf{T}$ by

$$\mathbf{T} = \mathbf{G}^T\mathbf{V}\mathbf{\Sigma}^{-1/2}. \tag{4.67}$$

  and apply it to the state-space system (4.61).

The result of this process has the following property:

$$\mathbf{P} = \mathbf{Q} = \mathbf{\Sigma}\,, \tag{4.68}$$

i.e. the solutions of CAREs (4.62)-(4.63) are equal and diagonal. The resulting state-space realization $\{\tilde{\mathbf{A}}, \tilde{\mathbf{B}}, \tilde{\mathbf{C}}, \tilde{\mathbf{D}}\}$ is called "positive real balanced". Therefore the starting passive state-space realization $\{\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}\}$ can be converted into a positive real balanced one using the similarity transformation (4.67) according to (4.27). A complete proof that the resulting state-space realization verifies the passivity conditions in Table 4.3 and is positive real balanced can be found in [171].

It can be demonstrated [119] that the static hybrid matrix $\tilde{\mathbf{M}}$, resulting from the reactance extraction method applied on the positive real balanced realization $\{\tilde{\mathbf{A}}, \tilde{\mathbf{B}}, \tilde{\mathbf{C}}, \tilde{\mathbf{D}}\}$, i.e.

$$\tilde{\mathbf{M}} = \begin{bmatrix} \tilde{\mathbf{D}} & -\tilde{\mathbf{C}} \\ \tilde{\mathbf{B}} & -\tilde{\mathbf{A}} \end{bmatrix} \tag{4.69}$$

satisfies the static passivity condition in Table 4.1: $\tilde{\mathbf{M}} + \tilde{\mathbf{M}}^T \succeq 0$.

The next step is to obtain an internally reciprocal state-space model starting from $\{\tilde{\mathbf{A}}, \tilde{\mathbf{B}}, \tilde{\mathbf{C}}, \tilde{\mathbf{D}}\}$. Details on how to guarantee the reciprocity in the reactance extraction method are discussed in the following.

Reciprocal state-space realization

As it was demonstrated in [119], the reactance extraction procedure applied to a internally reciprocal state-space model leads to a reciprocal static hybrid matrix $\tilde{\mathbf{M}}$ (4.69). In other words: if the state-space matrices $\{\tilde{\mathbf{A}}, \tilde{\mathbf{B}}, \tilde{\mathbf{C}}, \tilde{\mathbf{D}}\}$ verify the internal reciprocity conditions in Table 4.3, repeated here as

$$\tilde{\mathbf{A}}\mathbf{\Pi} = \mathbf{\Pi}\tilde{\mathbf{A}}^T\,, \tag{4.70}$$

$$\tilde{\mathbf{B}} = \mathbf{\Pi}\tilde{\mathbf{C}}^T\,, \tag{4.71}$$

(where $\mathbf{S}_{ext} = \mathbf{I}_P$ because an impedance transfer function is considered) then the static hybrid matrix resulting from the reactance extraction procedure will satisfy static reciprocity conditions in Table 4.1, i.e.

$$\mathbf{S}_{ext}\tilde{\mathbf{M}} = \tilde{\mathbf{M}}^T\mathbf{S}_{ext}\,. \tag{4.72}$$

In order to attain a reciprocal and passive synthesis, two alternative strategies are possible, depending whether reciprocity constraints (4.70)-(4.71) are imposed before or after the solution of the Algebraic Riccati Equation (CARE) for the construction of the positive real balanced realization, i.e.

- pre-care reciprocity: in this case a state-space with reciprocity constraints is obtained before solving the CAREs (4.62)-(4.62). Which means that the matrix $\mathbf{\Pi}$ relating the state-space matrices in a reciprocal model is found by means of standard techniques [124]. To preserve reciprocity in the positive real balanced state-space model resulting from the similarity transformation obtained from the solution of CAREs (4.62)-(4.62), a particular solution to the dual-CARE (4.62) can be found. Indeed once the solution matrix $\mathbf{P}$ of (4.62) is available, since all the solutions of the PRL in Table 4.3 are related by similarity transformations [119], the solution of the dual-CARE (4.62) can be found as

$$\mathbf{Q} = \mathbf{\Pi P \Pi}\,, \tag{4.73}$$

with $\mathbf{\Pi} = \mathbf{\Pi}^T = \mathbf{\Pi}^{-1}$, i.e. symmetric and orthogonal, from the solution of the reciprocity constraint in Table 4.3. Relation (4.73) simply results from the imposition of state-space reciprocity constraints in the CARE (4.62). Applying now the similarity transformation matrix $\mathbf{T}$ (4.67) leads to a balanced reciprocal state-space model, with reciprocity matrix

$$\tilde{\mathbf{\Pi}} = \mathbf{T}^T \mathbf{\Pi T}\,. \tag{4.74}$$

In order to obtain an internally reciprocal model, according to Table 4.3, the reciprocity matrix $\tilde{\mathbf{\Pi}}$ has to be a signature matrix. Therefore $\tilde{\mathbf{\Pi}}$ is factored according to [171] (Algorithm 2) obtaining

$$\tilde{\mathbf{\Pi}} = \mathbf{V \Lambda V}^T \tag{4.75}$$

with $\mathbf{\Lambda}$ signature matrix, i.e. diagonal matrix with 1 and -1 on the main diagonal. Please note that factorization (4.75) can be performed as a consequence of (4.74), see [171] for details. Using now $\mathbf{V}$ as a new similarity transformation, the resulting state-space model will be internally reciprocal and positive real balanced;

- post-care reciprocity: in this case the starting point is the positive real balanced state-space model $\{\tilde{\mathbf{A}}, \tilde{\mathbf{B}}, \tilde{\mathbf{C}}, \tilde{\mathbf{D}}\}$. Reciprocity conditions (4.70)-(4.71) are explicitly imposed using an orthogonal symmetric matrix $\hat{\mathbf{\Pi}}$ obtained from the direct solution of the system

$$\tilde{\mathbf{A}}^T \hat{\mathbf{\Pi}} = \hat{\mathbf{\Pi}} \tilde{\mathbf{A}}\,, \tag{4.76}$$

$$\tilde{\mathbf{C}} = \tilde{\mathbf{B}}^T \hat{\mathbf{\Pi}}\,, \tag{4.77}$$

via standard tools like SeDuMi [124]. Please note that in addition to symmetry also orthogonality is required to preserve the positive real

property [171]. Internal reciprocity is obtained decomposing $\hat{\mathbf{\Pi}}$ as

$$\hat{\mathbf{\Pi}} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^T\,, \tag{4.78}$$

with $\mathbf{\Lambda}$ signature matrix.

In both cases the outcome will be an internally reciprocal balanced state-space model $\{\hat{\mathbf{A}}, \hat{\mathbf{B}}, \hat{\mathbf{C}}, \hat{\mathbf{D}}\}$ with internal signature matrix

$$\mathbf{\Lambda} = \begin{bmatrix} \mathbf{I}_{N_l} & \mathbf{0} \\ \mathbf{0} & -\mathbf{I}_{N_c} \end{bmatrix}\,, \tag{4.79}$$

with $N_l + N_c = N$.

The balanced realization obtained via (4.67) guarantees the passivity of the hybrid matrix

$$\hat{\mathbf{M}} = \begin{bmatrix} \hat{\mathbf{D}} & -\hat{\mathbf{C}} \\ \hat{\mathbf{B}} & -\hat{\mathbf{A}} \end{bmatrix} \tag{4.80}$$

resulting from the reactance extraction procedure, i.e.

$$\hat{\mathbf{M}} + \hat{\mathbf{M}}^T \succeq 0\,. \tag{4.81}$$

The internal reciprocity guarantees the sign symmetry of the hybrid matrix (4.80), i.e.

$$\mathbf{S}_M\hat{\mathbf{M}} = \hat{\mathbf{M}}^T\mathbf{S}_M\,, \tag{4.82}$$

with signature matrix $\mathbf{S}_M$ defined by

$$\mathbf{S}_M = \begin{bmatrix} \mathbf{I}_P & \mathbf{0} \\ \mathbf{0} & \mathbf{\Lambda} \end{bmatrix} \tag{4.83}$$

and $\mathbf{\Lambda}$ internal signature matrix from (4.79).

Once the internally reciprocal balanced state-space model $\{\hat{\mathbf{A}}, \hat{\mathbf{B}}, \hat{\mathbf{C}}, \hat{\mathbf{D}}\}$ is available, the dynamic synthesis problem is mainly reduced to the synthesis of the associated hybrid static matrix (4.80). Next subsection outlines how to perform the synthesis of (4.80) thus completing the description of Youla's reactance extraction algorithm.

Synthesis algorithm

The starting point is the internally reciprocal and balanced state-space realization $\{\hat{\mathbf{A}}, \hat{\mathbf{B}}, \hat{\mathbf{C}}, \hat{\mathbf{D}}\}$ from which the hybrid, passive and sign symmetric,

matrix $\hat{\mathbf{M}}$ (4.80) is obtained. In order to convert (4.80) into an equivalent static $P+N$ circuit it is important to note that the signature matrix $\mathbf{S}_M$ (4.83) suggests the following partition of matrix $\hat{\mathbf{M}}$ (4.80),

$$\mathbf{S}_M = \begin{bmatrix} \mathbf{I}_P & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_{N_l} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & -\mathbf{I}_{N_c} \end{bmatrix} \rightarrow \hat{\mathbf{M}} = \begin{bmatrix} \hat{\mathbf{D}} & -\hat{\mathbf{C}}_l & -\hat{\mathbf{C}}_c \\ \hat{\mathbf{B}}_l & -\hat{\mathbf{A}}_{l,l} & -\hat{\mathbf{A}}_{l,c} \\ \hat{\mathbf{B}}_c & -\hat{\mathbf{A}}_{c,l} & -\hat{\mathbf{A}}_{c,c} \end{bmatrix} . \tag{4.84}$$

According to the partitioning of the hybrid matrix $\hat{\mathbf{M}}$ in (4.84), currents and voltages of the resulting static hybrid network can be arranged as

$$\begin{bmatrix} \boldsymbol{v}_p \\ \boldsymbol{v}_l \\ \boldsymbol{i}_c \end{bmatrix} = \begin{bmatrix} \hat{\mathbf{D}} & -\hat{\mathbf{C}}_l & -\hat{\mathbf{C}}_c \\ \hat{\mathbf{B}}_l & -\hat{\mathbf{A}}_{l,l} & -\hat{\mathbf{A}}_{l,c} \\ \hat{\mathbf{B}}_c & -\hat{\mathbf{A}}_{c,l} & -\hat{\mathbf{A}}_{c,c} \end{bmatrix} \begin{bmatrix} \boldsymbol{i}_p \\ \boldsymbol{i}_l \\ \boldsymbol{v}_c \end{bmatrix} \tag{4.85}$$

where $\boldsymbol{i}_p \in \mathbb{R}^P$ are the input ports of the state-space model, $\boldsymbol{i}_l \in \mathbb{R}^{N_l}$ are the ports of the hybrid matrix to be closed on inductors and $\boldsymbol{v}_c \in \mathbb{R}^{N_c}$ are the ports of the hybrid matrix to be closed on capacitors.

Three are the main steps involved by the synthesis of the hybrid matrix $\hat{\mathbf{M}}$ partitioned as in (4.85) in order to extract noise-compliant positive resistors, i.e.

1. impedance sub-network synthesis: from (4.85) an impedance sub-network is identified as

$$\mathbf{Z}_M = \begin{bmatrix} \hat{\mathbf{D}} & -\hat{\mathbf{C}}_l \\ \hat{\mathbf{B}}_l & -\hat{\mathbf{A}}_{l,l} \end{bmatrix} . \tag{4.86}$$

   Due to passivity and reciprocity of $\hat{\mathbf{M}}$, it can be demonstrated [119] that $\mathbf{Z}_M$ is a symmetric positive definite matrix. The synthesis of a static impedance matrix was discussed in Section 4.2.3. Applying the eigenvalue decomposition on $\mathbf{Z}_M$ leads to

$$\mathbf{Z}_M = \mathbf{N}_z \mathbf{D}_z \mathbf{N}_z^T , \tag{4.87}$$

   with $\mathbf{D}_z \in \mathbb{R}^{N_r \times N_r}$ positive diagonal matrix and $\mathbf{N}_z \in \mathbb{R}^{(P+N_l)\times N_r}$ turn ratio matrix for a lossless ideal multi-port transformer having $P+N_l$ primary ports and N_r secondary ports. The rank of $\mathbf{Z}_M$ determines the value of N_r;

2. admittance sub-network synthesis: from (4.85) an admittance sub-network is identified as

$$\mathbf{Y}_M = -\hat{\mathbf{A}}_{c,c} . \tag{4.88}$$

Due to the stability of the state space model, it can be demonstrated [119] that $\mathbf{Y}_M$ is a symmetric positive definite matrix. The synthesis of a static admittance matrix was discussed in Section 4.2.3. Applying the eigenvalue decomposition on $\mathbf{Y}_M$ reads

$$\mathbf{Y}_M = \mathbf{N}_y \mathbf{D}_y \mathbf{N}_y^T , \tag{4.89}$$

with $\mathbf{D}_y \in \mathbb{R}^{N_g \times N_g}$ positive diagonal matrix and $\mathbf{N}_z \in \mathbb{R}^{(P+N_c)\times N_g}$ turn ratio matrix for a lossless ideal multi-port transformer having $P + N_c$ primary ports and N_g secondary ports. The rank of $\mathbf{Y}_M$ determines the value of N_g;

3. transformer sub-network: from (4.85) an ideal multi-port transformer connecting impedance (4.87) and admittance (4.89) sub-networks previously identified can be defined as

$$\mathbf{N}_t = \begin{bmatrix} \hat{\mathbf{B}}_c & -\hat{\mathbf{A}}_{c,l} \end{bmatrix} = \begin{bmatrix} \hat{\mathbf{C}}_c & \hat{\mathbf{A}}_{l,c} \end{bmatrix}^T . \tag{4.90}$$

As a consequence of the sign symmetry of $\hat{\mathbf{M}}$ (4.82) it is easy to prove that $\mathbf{N}_t \in \mathbb{R}^{N_c \times (P+N_l)}$ is the turns ratio matrix of an ideal multi-port transformer having $P + N_l$ primary ports and N_c secondary ports.

The connection of the three sub-networks $\mathbf{Z}_M$ (4.87) $\mathbf{Y}_M$ (4.89) and $\mathbf{N}_t$ (4.90), closed on N_l unitary inductances and N_c unitary capacitances concludes the synthesis of the state-space model associated to the impedance transfer function (4.61). Figure (4.19) clearly demonstrates how to interconnect the sub-circuits that constitute the desired Youla's reactance extraction synthesis.

RC ladder network example

The RC ladder network example from Section (4.3.3) is revisited here to show the noise-compliance of Youla's reactance extraction synthesis.

The starting point is the state-space model for the RC ladder network in Gilbert canonical form, i.e.

$$\mathbf{A} = \frac{1}{R}\begin{pmatrix} p_1 & 0 \\ 0 & p_2 \end{pmatrix} , \quad \boldsymbol{b} = \begin{pmatrix} 1 \\ 1 \end{pmatrix} , \quad \boldsymbol{c} = \begin{pmatrix} \rho_1 \\ \rho_2 \end{pmatrix}^T , \tag{4.91}$$

where $p_{1,2}$ are

$$p_{1,2} = \frac{-3 \pm \sqrt{9-4}}{4} = \frac{-3 \pm \sqrt{5}}{2} \tag{4.92}$$

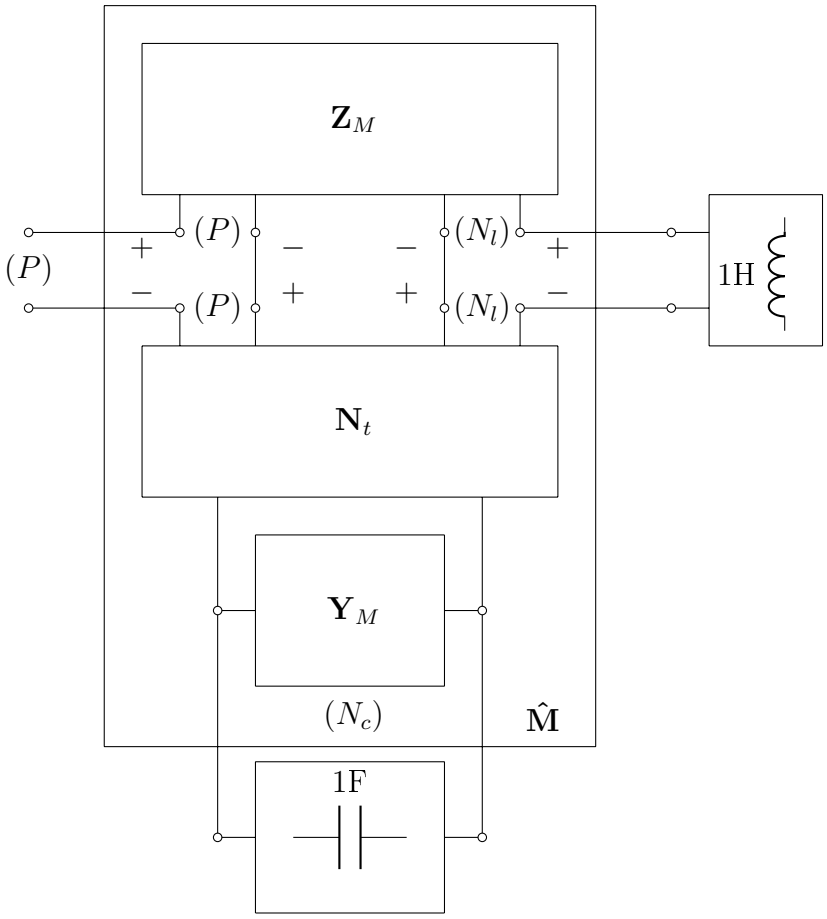

Figure 4.19: Interconnection of the three sub-networks $\mathbf{Z}_M$ (4.87) $\mathbf{Y}_M$ (4.89) and $\mathbf{N}_t$ (4.90), which closed on N_l unitary inductances and N_c unitary capacitances concludes the synthesis of the state-space model associated to the impedance transfer function (4.61). Each sub-network is synthesized using resistors and ideal transformers like described in Section 4.2.3, as an example the synthesis of $\mathbf{Z}_M$ is depicted in Figure 4.20. Note that each port in the figure denotes collectively a set of P, N_l, or N_c ports for the interface, inductance, and capacitance ports, respectively.

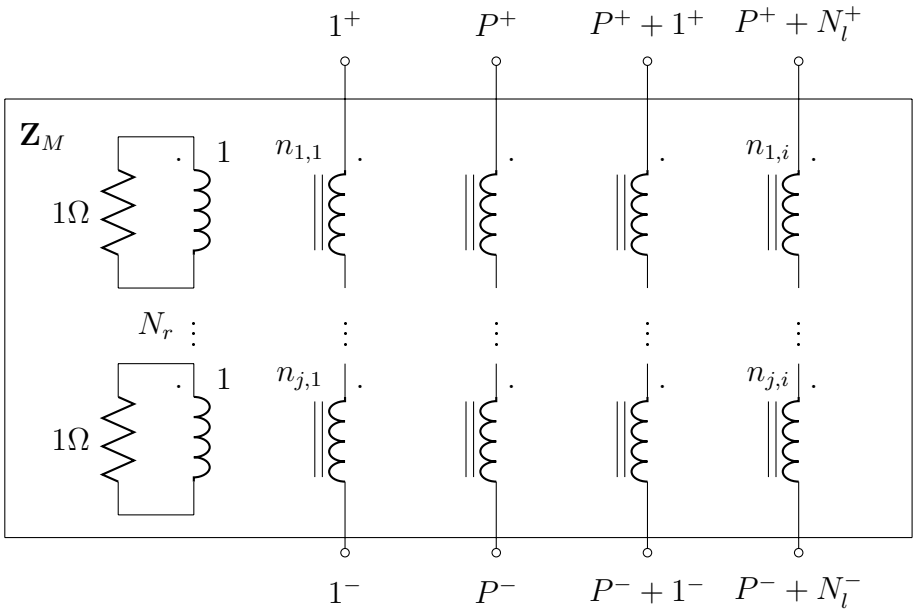

Figure 4.20: Belevitch (RT) synthesis for the sub-network $\mathbf{Z}_M$ resulting from Youla's procedure. According to (4.87): the N_r secondary ports of the ideal transformer described by $\mathbf{N}_z$ are closed on unitary resistors, while the first P primary ports realize the port of the impedance (4.61) and the last N_l primary ports are connected in series to unitary inductors. Only a few turns ratio values $n_{j,i}$, with $i = 1, \ldots, P + N_l$ and $j = 1, \ldots, N_r$, are considered for the multi-port ideal transformer to simplify the presentation of the circuit. P^+, N_l^+ and P^-, N_l^- are, respectively, positive and negative terminals for the $P + N_l$ ports of $\mathbf{Z}_M$.

and

$$\rho_{1,2} = \frac{5 \pm \sqrt{5}}{10}. \tag{4.93}$$

The conversion to a reciprocal and positive real balanced realization is straightforward: reciprocity is guaranteed by construction when dealing with a 1-port circuit, while to obtain a positive real balanced model it is sufficient to impose [162]

$$\mathbf{A} = \mathbf{A}^T, \tag{4.94}$$

$$\mathbf{B} = \mathbf{C}^T. \tag{4.95}$$

Since condition (4.94) is already met by (4.91), only (4.95) should be considered. It is easy to verify that the similarity matrix

$$\mathbf{T} = \begin{pmatrix} \sqrt{\rho_1} & 0 \\ 0 & \sqrt{\rho_2} \end{pmatrix} \tag{4.96}$$

transforms (4.91) into the positive real balanced realization

$$\hat{\mathbf{A}} = \frac{1}{R}\begin{pmatrix} p_1 & 0 \\ 0 & p_2 \end{pmatrix} \quad \hat{\boldsymbol{b}} = \begin{pmatrix} \sqrt{\rho_1} \\ \sqrt{\rho_2} \end{pmatrix} \quad \hat{\boldsymbol{c}} = \begin{pmatrix} \sqrt{\rho_1} \\ \sqrt{\rho_2} \end{pmatrix}^T. \tag{4.97}$$

Using the reactance extraction procedure the static hybrid matrix $\hat{\mathbf{M}}$ is found as

$$\hat{\mathbf{M}} = \begin{bmatrix} 0 & -\sqrt{\rho_1} & -\sqrt{\rho_2} \\ \sqrt{\rho_1} & -p_1/R & 0 \\ \sqrt{\rho_2} & 0 & -p_2/R \end{bmatrix}, \tag{4.98}$$

with the corresponding signature matrix

$$\mathbf{S}_M = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{bmatrix}. \tag{4.99}$$

The synthesis of the hybrid matrix $\hat{\mathbf{M}}$ is depicted in Figure 4.21. Please note that: since the impedance sub-network $\mathbf{Z}_M$ in (4.98) is null, the input port of $Z(s)$ is directly connected to the transformer sub-network described by the turns ration matrix $\mathbf{N}_t = [\sqrt{\rho_1} \quad \sqrt{\rho_2}]^T$. While the admittance sub-network $\mathbf{Y}_M$ is composed by the two admittances $-p_1/R$ and $-p_2/R$, directly connected in parallel to the unitary capacitors and to the secondary ports of the multi-port transformer described by $\mathbf{N}_t$.

The voltage output noise spectral density can be evaluated analytically like in Section 4.3.2 leading to the equation

$$\bar{V}_o^2(\omega) = 4K_bTR\left(-\frac{\rho_1^2 p_1}{\omega^2 R^2 C^2 + p_1^2} - \frac{\rho_2^2 p_2}{\omega^2 R^2 C^2 + p_2^2}\right). \tag{4.100}$$

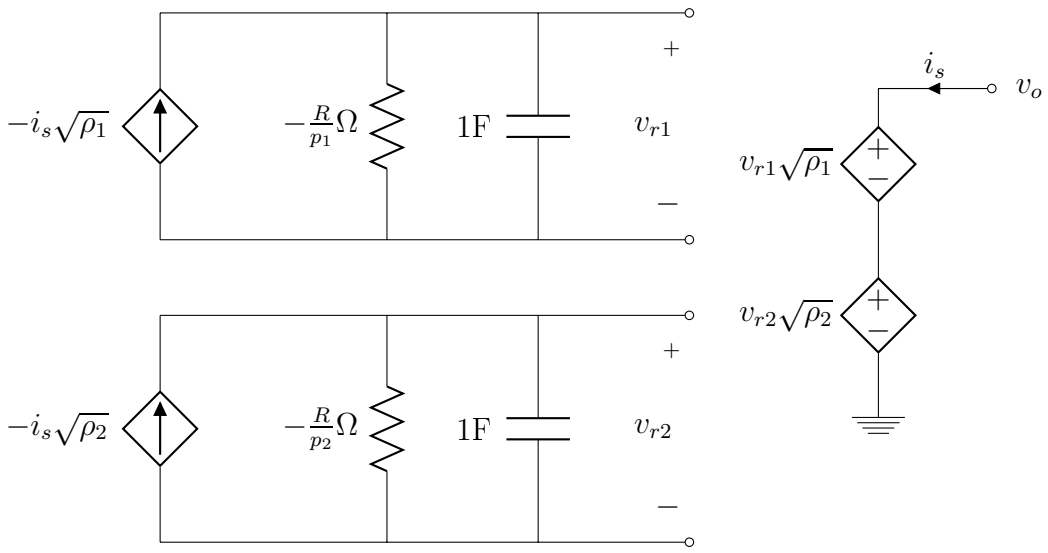

Figure 4.21: Youla's reactance extraction synthesis result from (4.98) for the RC ladder network example.

Equation (4.100) is compared in Figure (4.22) with the results obtained from Spice noise analysis using the original RC ladder network and Youla's synthesis based on (4.98). As expected the three curves overlap each other.

Circuit complexity

The main issue of Youla's reactance extraction synthesis lies in the complexity of the resulting network, indeed the number of elements scales as $\mathcal{O}(P^2N^2)$ compared with $\mathcal{O}(P^2N)$ for the direct synthesis method from Section 4.3.2. There are several reasons for such a big difference in the number of elements between the two synthesis methods, i.e.

- canonicity: as it was demonstrated in [119], it is impossible for a purely-reciprocal[6] passive synthesis to jointly use the minimum number of passive (resistors) and reactive (capacitors and inductors) elements. This means that a purely-reciprocal passive synthesis will always require more elements compared to a passive non-reciprocal one;

- noise-compliance: in order to preserve the noise response, the RT synthesis is used on the hybrid static matrix (4.80) resulting in three sub-

[6] A circuit composed only by RLCT elements was defined as purely-reciprocal in Section 4.2.1.

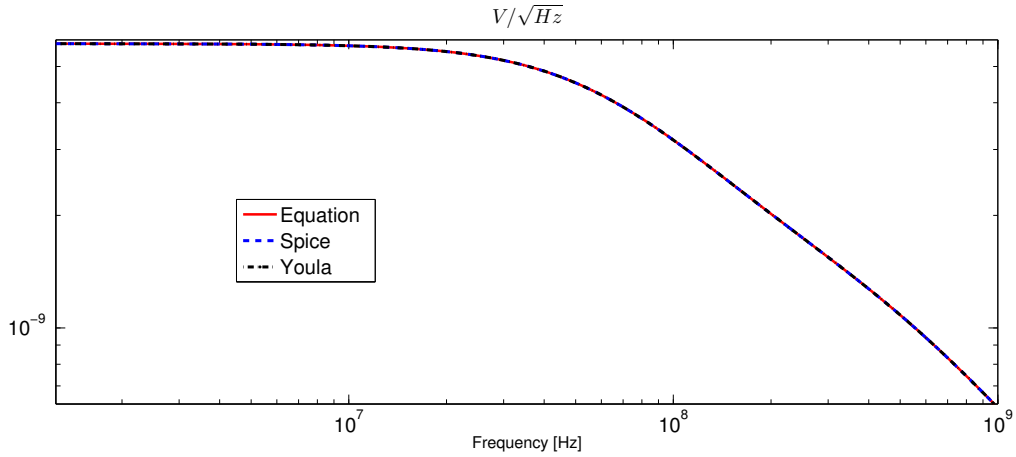

Figure 4.22: Comparison between noise analysis results obtained by means of analytical equation (4.100) (red line), Spice simulation using the circuit in Figure 4.21 (dot-dashed line) and Spice simulation on the original RC ladder network (blue dashed line).

networks of multi-port ideal transformers. The synthesis of those transformers requires a large number of controlled sources;

- sparsity: starting with a sparse model, i.e. Gilbert canonical form, will not result into a sparse reciprocal balanced realization because model structure is not preserved by the similarity transformation (4.67) in the general case.

As a consequence to the considerations above, the pure-reciprocity constraint will be relaxed in the following while preserving noise-compliance in the synthesis. Removing the constraint of pure-reciprocity leads to a synthesis with a complexity comparable to the direct synthesis method in Section 4.3.2.

4.3.4 Darlington's resistance extraction

Duo to the complexity of the resulting network, Youla's reactance extraction can only be used in practice for state-space models possessing a low dynamic order. As discussed in previous Section 4.3.3, this limitation is tightly related with the requirement of preserving a purely-reciprocal circuit in the synthesis process. Since the main concern in this work lies in attaining a noise-compliant synthesis, it is conceivable to remove the pure-reciprocity constraint in order to achieve a result of practical relevance. Youla's reactance extraction procedure could be modified in that sense, but the similarity

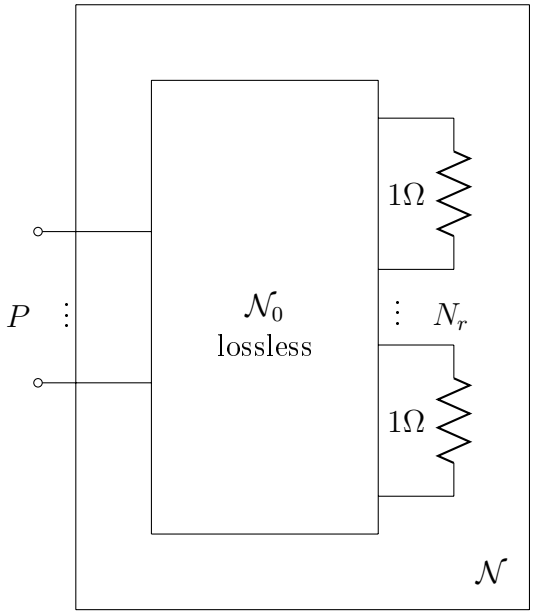

Figure 4.23: Darlington's Resistance extraction: lossless dynamic sub-network $\mathcal{N}_0$ closed on N_r resistors.

transformation (4.67) will not allow to preserve the structure of the state-space matrices in any case, thus loosing the sparsity pattern provided by Gilbert canonical form. Therefore a different approach is considered in this section, i.e. Darlington's Resistance extraction technique.

The classic Resistance extraction algorithm was proposed by Darlington in [147]. Given a linear and passive dynamic network $\mathcal{N}$, the main idea is: extract all the resistors from $\mathcal{N}$ thus obtaining a lossless dynamic network $\mathcal{N}_0$ like depicted in Figure 4.23. Since lossless networks are noise compliant by construction, even if non-reciprocal[7], the dynamic lossless sub-network $\mathcal{N}_0$ can be synthesized via the direct synthesis method from Section 4.3.2, leading to a circuit with a complexity scaling as $\mathcal{O}(P^2N)$. In this way a canonical noise-compliant synthesis can be obtained.

In the following the Resistance extraction technique will be applied to the passive state-space model

$$\dot{\boldsymbol{x}}(t) = \mathbf{A}\boldsymbol{x}(t) + \mathbf{B}\boldsymbol{i}(t)\,, \tag{4.101}$$

$$\boldsymbol{v}(t) = \mathbf{C}\boldsymbol{x}(t) + \mathbf{D}\boldsymbol{i}(t)\,, \tag{4.102}$$

[7]As discussed in Section 4.3.3, noise-compliance is related with the proper usage of controlled sources in the synthesis process: only lossless networks composed of controlled sources are noise compliant. Since reciprocity is not required, even non-reciprocal lossless networks will result into a noise-compliant circuit.

with input vector $\boldsymbol{i} \in \mathbb{R}^P$, output vector $\boldsymbol{v} \in \mathbb{R}^P$ and state-space matrices: $\mathbf{A} \in \mathbb{R}^{N\times N}$, $\mathbf{B} \in \mathbb{R}^{N\times P}$, $\mathbf{C} \in \mathbb{R}^{P\times N}$ and $\mathbf{D} \in \mathbb{R}^{P\times P}$. Similar results hold for scattering models [119].

State-space resistance extraction

The main objective of the resistance extraction method consists in the extraction of a lossless dynamic state-space model $\mathcal{N}_0$

$$\dot{\boldsymbol{x}}(t) = \mathbf{A}_0\boldsymbol{x}(t) + \mathbf{B}_0 \begin{bmatrix} \boldsymbol{i}(t) \\ \boldsymbol{i}_r(t) \end{bmatrix}, \tag{4.103}$$

$$\begin{bmatrix} \boldsymbol{v}(t) \\ \boldsymbol{v}_r(t) \end{bmatrix} = \mathbf{C}_0\boldsymbol{x}(t) + \mathbf{D}_0 \begin{bmatrix} \boldsymbol{i}(t) \\ \boldsymbol{i}_r(t) \end{bmatrix}, \tag{4.104}$$

which closed on N_r unitary resistors $\mathbf{I}_r$, i.e.

$$\boldsymbol{v}_r(t) = -\mathbf{I}_r\boldsymbol{i}_r(t)\,, \tag{4.105}$$

leads to the state-space of the starting passive dynamic network $\mathcal{N}$ described by (4.101)-(4.102). Please note that the lossless dynamic state-space realization (4.103) has the same dynamic order of the original dynamic network (4.101), i.e. $\mathbf{A}_0 \in \mathbb{R}^{N\times N}$, thus order minimality is preserved. The main difference between network $\mathcal{N}$ and $\mathcal{N}_0$ is in the number of ports. Indeed $\mathbf{D}_0 \in \mathbb{R}^{(P+N_r)\times(P+N_r)}$, where N_r is the number of resistors closing the N_r inputs $\boldsymbol{i}_r$ of the lossless sub-network, with $N_r = \text{rank}\{\mathbf{D} + \mathbf{D}^T\}$.

Unfortunately it is not possible to provide a simple circuit interpretation of the resistance extraction procedure, like it was done for the reactance extraction in Section 4.3.3, because the extraction of resistances from a state-space[8] model involves the solution of the quadratic system of equations

$$\mathbf{P}\mathbf{A} + \mathbf{A}^T\mathbf{P} = -\mathbf{L}\mathbf{L}^T\,, \tag{4.106}$$

$$\mathbf{P}\mathbf{B} = \mathbf{C}^T - \mathbf{L}\mathbf{W}\,, \tag{4.107}$$

$$\mathbf{D} + \mathbf{D}^T = \mathbf{W}^T\mathbf{W} \tag{4.108}$$

with $\mathbf{L} \in \mathbb{R}^{N\times N_r}$, $\mathbf{W} \in \mathbb{R}^{N_r\times P}$ and $\mathbf{P} \in \mathbb{R}^{N\times N}$.

According to [119], once the solution matrices $\mathbf{L}, \mathbf{W}, \mathbf{P}$ for (4.106)-(4.108) are known, the lossless state-space model $\mathcal{N}_0$ (4.103)-(4.104) can be directly

[8] The extraction of the resistors from the state-space model of $\mathcal{N}$ is equivalent to the identification of the lossless dynamical sub-network $\mathcal{N}_0$.

constructed as

$$\mathbf{A}_0 \doteq \frac{1}{2}(\mathbf{A}-\mathbf{A}^T)\,, \tag{4.109}$$

$$\mathbf{B}_0 \doteq \left[\frac{1}{2}(\mathbf{B}+\mathbf{C}^T)-\frac{1}{\sqrt{2}}\mathbf{L}\right] \doteq \mathbf{C}_0^T\,, \tag{4.110}$$

$$\mathbf{D}_0 \doteq \begin{bmatrix} \frac{1}{2}(\mathbf{D}-\mathbf{D}^T) & \frac{1}{\sqrt{2}}\mathbf{W}^T \\ -\frac{1}{\sqrt{2}}\mathbf{W} & \mathbf{0} \end{bmatrix}\,. \tag{4.111}$$

To verify that $\{\mathbf{A}_0,\mathbf{B}_0,\mathbf{C}_0,\mathbf{D}_0\}$ constitutes a lossless state-space model it is sufficient to check the relations

$$\mathbf{A}_0+\mathbf{A}_0^T = 0\,, \tag{4.112}$$

$$\mathbf{B}_0 = \mathbf{C}_0^T\,, \tag{4.113}$$

$$\mathbf{D}_0+\mathbf{D}_0^T = 0\,, \tag{4.114}$$

that are called the lossless PRL equations. The relation with the LMI in Table 4.3 (passivity column) is discussed in [119].

Please note that from (4.109) it is evident how the structure of matrix $\mathbf{A}$ is preserved by the resistance extraction process. It follows that Gilbert canonical form can be used in order to attain a low complexity synthesis like in Section 4.3.2. Moreover, if $\mathbf{D}=\mathbf{D}^T$ in (4.111), which is usually the case if the starting impedance model describes the immittance of a linear reciprocal circuit, $\mathbf{D}_0$ will only have the outer block-diagonal element different from zero, i.e. $\mathbf{W}$.

Once the lossless state-space model (4.109)-(4.111) is known from the solution of the PRL, the synthesis is straightforward, being a simple application of the direct synthesis method on (4.103)-(4.104), with the last N_r ports closed on unitary resistors.

Positive Real Lemma solution

In the solution of the PRL (4.106)-(4.108), leading to $\mathbf{L}$ and $\mathbf{W}$ in (4.109)-(4.111), the attention is restricted here to the case in which

$$\mathbf{D}+\mathbf{D}^T \succ 0\,, \tag{4.115}$$

i.e. the state-space model (4.101)-(4.102) is strictly asymptotically passive. According to [119], under the hypothesis (4.115), the solution of the PRL (4.106)-(4.108) is obtained by the following steps

1. stating from a minimal and passive state-space realization $\{\mathbf{A},\mathbf{B},\mathbf{C},\mathbf{D}\}$, solve the Algebraic Riccati Equation for the positive definite matrix $\mathbf{P}$, i.e.

$$\mathbf{A}^T\mathbf{P}+\mathbf{P}\mathbf{A}+\left(\mathbf{P}\mathbf{B}-\mathbf{C}^T\right)\mathbf{R}^{-1}\left(\mathbf{B}^T\mathbf{P}-\mathbf{C}\right)=0\,, \tag{4.116}$$

   using standard methods [173, 174];

2. obtain an internally passive model using the similarity transformation defined by matrix $\mathbf{P}^{1/2}$;

3. perform the decomposition

$$\mathbf{D}+\mathbf{D}^T=\mathbf{W}^T\mathbf{W}\,; \tag{4.117}$$

4. decompose $\mathbf{W}$ as

$$\mathbf{W}=\mathbf{V}\mathbf{R}^{1/2}\,; \tag{4.118}$$

   with $\mathbf{V}$ orthogonal;

5. form $\mathbf{L}$ using the equation

$$\mathbf{L}=(\mathbf{P}^{1/2}\mathbf{B}-\mathbf{P}^{-1/2}\mathbf{C}^T)\mathbf{R}^{-1/2}\mathbf{V}^T \tag{4.119}$$

   where $\mathbf{B}$ and $\mathbf{C}$ come from the starting state-space in step 1, $\mathbf{R}$ and $\mathbf{V}$ are in (4.118).

This concludes the resistance extraction process.

Example

The simple RC circuit example considered in Section 4.3.2 and 4.3.3 is proposed here using the resistance extraction synthesis. Starting from the positive real state space realization obtained in 4.3.3, i.e.

$$\hat{\mathbf{A}}=\frac{1}{R}\begin{pmatrix}p_1 & 0\\ 0 & p_2\end{pmatrix}\quad \hat{\boldsymbol{b}}=\begin{pmatrix}\sqrt{\rho_1}\\ \sqrt{\rho_2}\end{pmatrix}\quad \hat{\boldsymbol{c}}=\begin{pmatrix}\sqrt{\rho_1}\\ \sqrt{\rho_2}\end{pmatrix}^T \tag{4.120}$$

with $p_{1,2}=\frac{-3\pm\sqrt{9-4}}{4}=\frac{-3\pm\sqrt{5}}{2}$ and $\rho_{1,2}=\frac{5\pm\sqrt{5}}{10}$, the conversion to an internally passive realization is performed based on the steps outlined before. Since the state-space matrix $\mathbf{D}$ is zero in (4.120), the solution matrix $\mathbf{W}$ in the PRL (4.106)-(4.108) is zero as well. If follows that (4.106) and (4.107) are decoupled and (4.107) reduces to $\mathbf{B}=\mathbf{C}^T$, which is satisfied by (4.120).

Table 4.4: This table summarizes the most relevant features of the synthesis methods presented in this work: direct synthesis (RCCS) from Section 4.3.2, Reactance extraction from Section 4.3.3 and Resistance extraction from Section 4.3.4. The constant factors differentiating the complexity of the RCCS synthesis with the Resistance extraction are such that $y < x \in \mathbb{R}$.

	RCCS	Reactance extraction	Resistance extraction
Pure-Reciprocity	no	yes	no
Complexity	$\mathcal{O}(yNP^2)$	$\mathcal{O}(N^2P^2)$	$\mathcal{O}(xNP^2)$
Noise-compliant	no	yes	yes

The solution matrix $\mathbf{L}$ in (4.106) can be obtained in this case by a direct application of the Cholesky factorization, leading to

$$\mathbf{L} = \begin{bmatrix} \sqrt{\frac{2p_1}{R}} & 0 \\ 0 & \sqrt{\frac{2p_2}{R}} \end{bmatrix} . \tag{4.121}$$

The state-space model associated to the lossless sub-network $\mathcal{N}_0$ in the resistance extraction synthesis framework results from a direct application of (4.109)-(4.111) based on (4.120) leading to

$$\mathbf{A}_0 = 0 , \tag{4.122}$$

$$\mathbf{B}_0 = \begin{bmatrix} \sqrt{\rho_1} & -\sqrt{\frac{p_1}{R}} & 0 \\ \sqrt{\rho_2} & 0 & -\sqrt{\frac{p_2}{R}} \end{bmatrix} = \mathbf{C}_0^T , \tag{4.123}$$

$$\mathbf{D}_0 = 0 . \tag{4.124}$$

Since the state-space matrices (4.122)-(4.124) define a lossless network, the RCCS synthesis method discussed in Section 4.3.2 can be still used while preserving noise-compliance. The resistance extraction procedure is completed closing the last two ports of the lossless sub-network $\mathcal{N}_0$ on unitary resistors like depicted in Figure 4.24. The result of the frequency dependent noise analysis performed in Spice are depicted in Figure 4.25. This result confirm the noise-compliance property of the proposed synthesis strategy.

4.3.5 Dynamic synthesis results and comparison

In this section simulation time results and complexity are compared for the state-space synthesis methods previously described. Table 4.4 highlights the

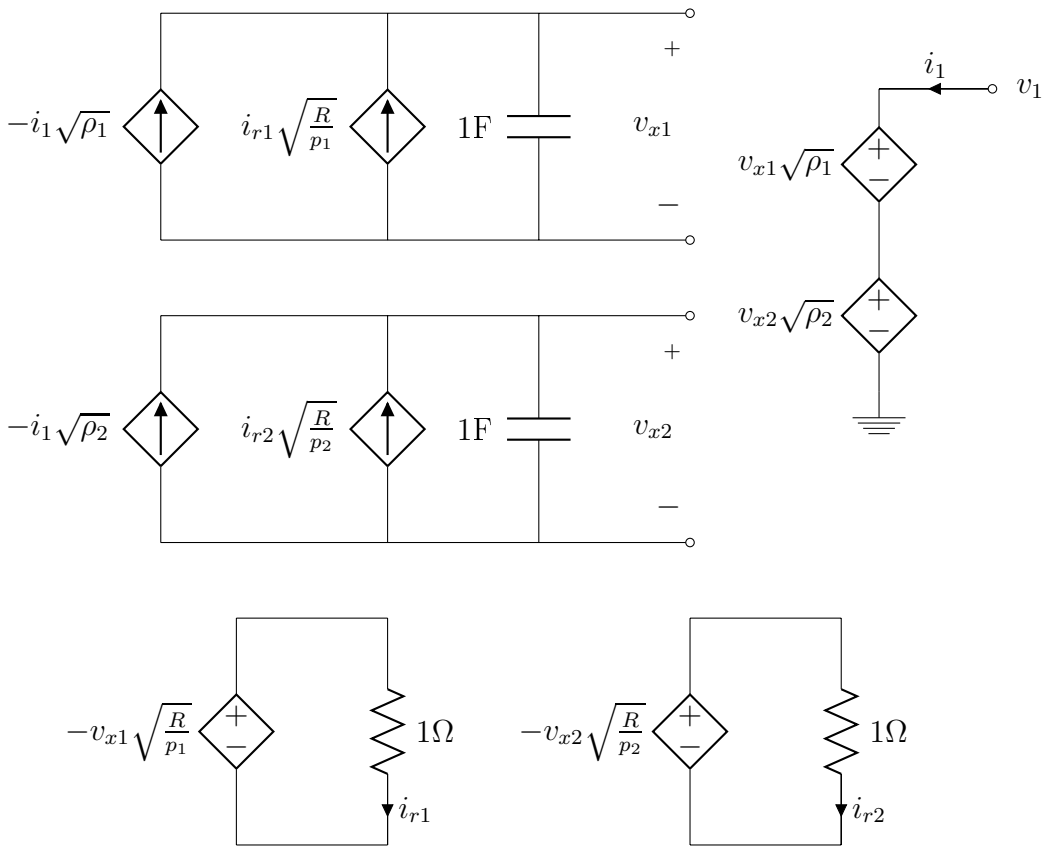

Figure 4.24: Darlington's resistance extraction synthesis resulting from (4.122)-(4.124) for the RC ladder network example.

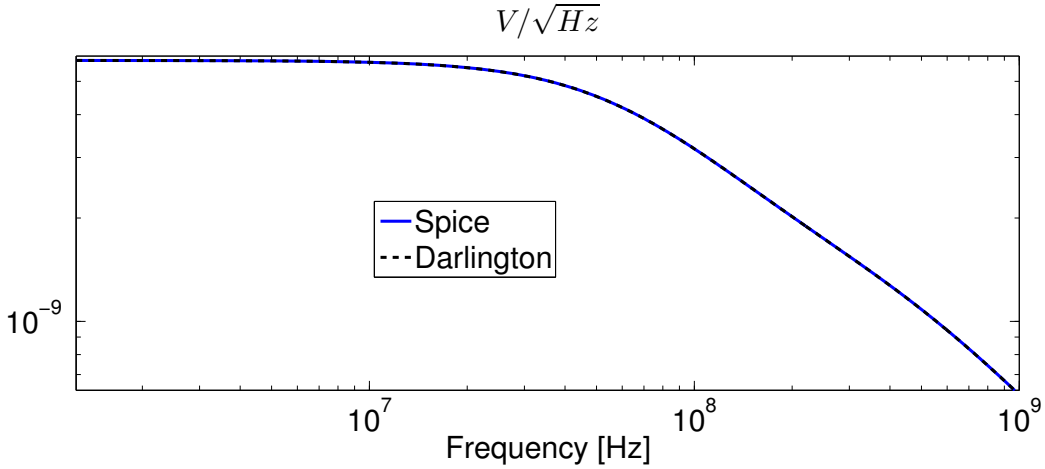

Figure 4.25: Comparison between noise analysis results obtained by means of Spice simulation using the circuit in Figure 4.24 (blue dashed line) and Spice simulation on the original RC ladder network (red line).

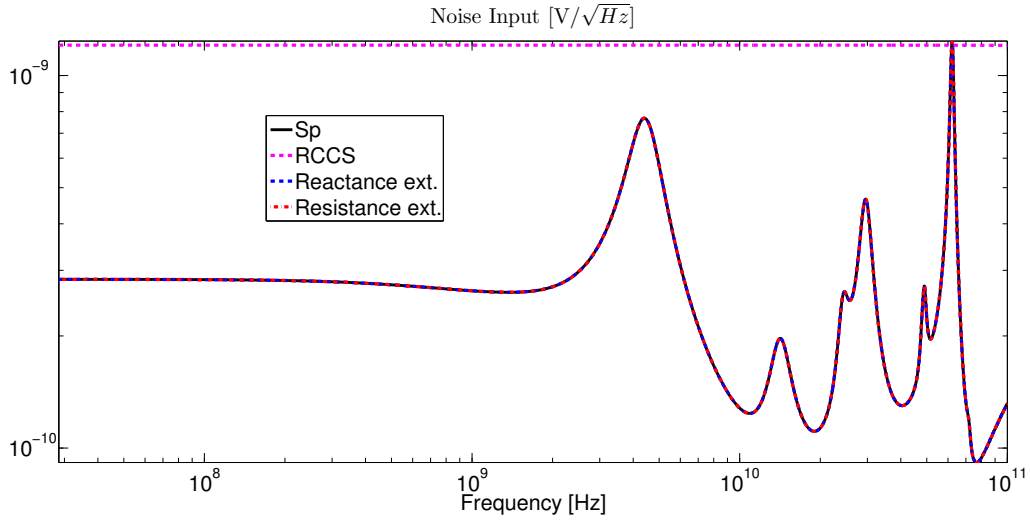

Figure 4.26: Noise analysis result for the first example in Table 4.5 (2 ports, order 20). As can be seen the results obtained from Reactance (blue dashed) and Resistance (red dashed) extractions-based synthesis methods match (to machine precision) the result of the noise analysis obtained from Spice using the raw S-parameters (Sp, black continuous line).

most relevant features of each synthesis method.
In a similar way to the static synthesis case, dynamic synthesis methods are validated through test cases conducted on artificial state-space models, obtained from the commercial software IdEM from IdemWorks, and on state-space models derived from real designs using the identification procedures discussed in Chapter 2. Frequency dependent noise responses are compared with the results of Spice-based noise simulations based on [176].

Automated testing procedure

A simple automated testing procedure is used to asses accuracy and complexity for the synthesis methods presented in this chapter. Several passive state-space models, with ports count in $[2, 100]$ and order in $[20, 600]$, are automatically generated using the software IdEM from IdemWorks and converted to Spice netlists. A small subset of those test cases is reported here with a detailed comparison of the number of network elements required by each synthesis method. The following Tables 4.5-4.6 compare the number of: nodes, capacitors, current controlled sources, inductors, resistors and voltage controlled sources used in the direct synthesis (RCCS), Reactance extraction and the resistance extraction methods. Moreover S-parameters simulations

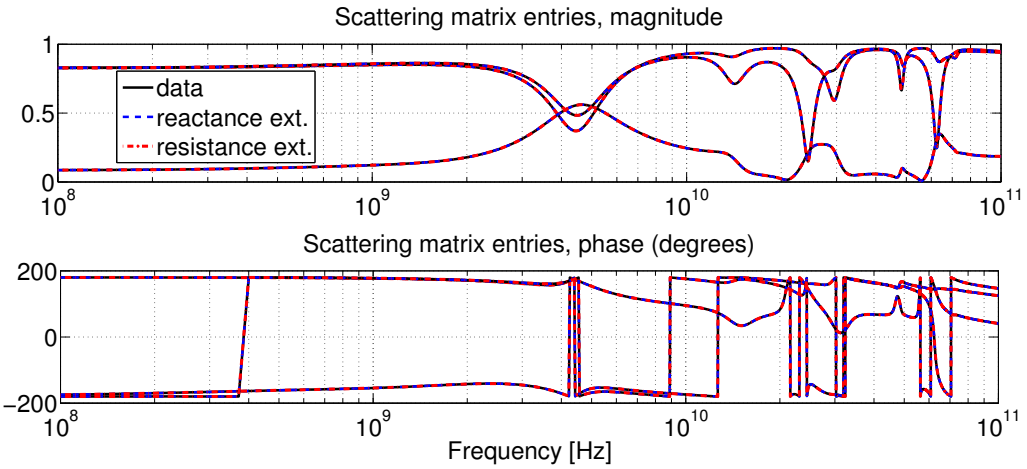

Figure 4.27: S-parameters for the first example in Table 4.5. As can be seen the results obtained from Reactance (blue dashed) and Resistance (red dashed) extractions-based synthesis methods match the original raw Scattering parameters with great accuracy.

were performed in Spice using the netlists resulting from each synthesis and execution times are reported in the tables as well. The accuracy of the S-parameters resulting from reactance and resistance extraction methods is demonstrated in Figure 4.27. Noise analysis results are depicted in Figure 4.26 for the first test case reported in Table 4.5. Note that the results concerning the RCCS synthesis refers to the Spice netlists obtained from the commercial software IdEM. The results collected in Table 4.5 confirm that the number of circuit elements scales as summarized in Table 4.4. When the number of ports is larger or comparable to the order of the state-space model, like in the examples of Table 4.6, resistance and reactance extraction methods have a comparable number of elements.

Tests derived from hardware designs

From the big set of real design test cases considered, two of particular relevance are considered in this section to compare the performances of the synthesis methods. The number of network elements is compared for both test cases in Table 4.7.

The first example is based on the state-space model for the centrally involved LC-tank coil of a RF Digitally Controlled Oscillator (DCO). DCOs can be tuned very accurately: their noise behaviour is a key figure of merit and requires therefore accurate noise modeling of all involved design parts. Thus modeling of the centrally involved LC-tank coil is a good benchmark

Table 4.5: Automatic test cases results. The number of network elements: capacitors (cap), current controlled sources (cccs), inductors (ind), resistors (res), voltage controlled voltage sources (vcvs), voltage controlled current sources (vccs), is compared for each synthesis method. As a global estimate of network complexity the total number of circuit elements is also reported. The simulation time refers to the execution time of S-parameter analysis in a Spice solver.

order: 20 ports: 2	RCCS	Reactance extraction	Resistance extraction
nodes	44	90	30
cap	40	10	20
cccs	–	208	24
ind	–	10	–
res	40	56	6
vcvs	–	34	4
vccs	144	208	142
total	224	526	196
time	150ms	370ms	146ms
order: 64 ports: 4			
nodes	72	869	84
cap	64	124	64
cccs	–	20934	186
ind	–	124	–
res	64	493	12
vcvs	–	338	8
vccs	368	20934	820
total	568	42947	1090
time	300ms	1m35s	730ms

Table 4.6: Automatic test cases results. The number of network elements: capacitors (cap), current controlled sources (cccs), inductors (ind), resistors (res), voltage controlled voltage sources (vcvs), voltage controlled current sources (vccs), is compared for each synthesis method. As a global estimate of network complexity the total number of circuit elements is also reported. The simulation time refers to the execution time of S-parameter analysis in a Spice solver.

order: 600 ports: 30	RCCS	Reactance extraction	Resistance extraction
nodes	660	2550	750
cap	600	300	600
cccs	–	153915	17808
ind	–	300	–
res	600	1590	90
vcvs	–	960	60
vccs	19740	153915	55002
total	21600	310980	73560
time	20s	20min	2min
order: 100 ports: 241			
nodes	1400	1463	741
cap	1200	121	241
cccs	–	51051	28895
ind	–	120	–
res	1200	902	300
vcvs	–	561	200
vccs	131400	51051	77575
total	135200	103806	107211
time	5min	10min	10min

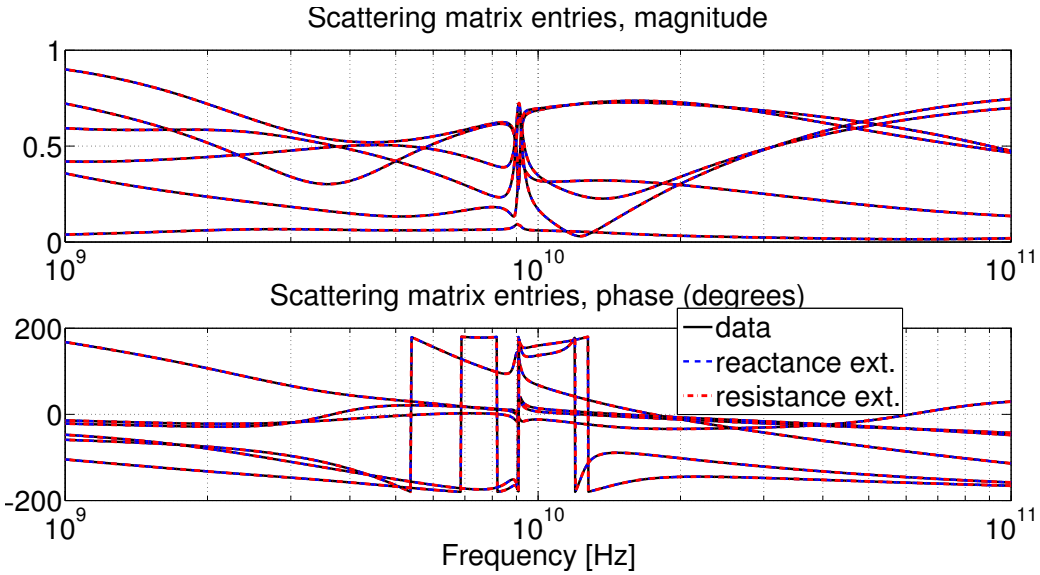

Figure 4.28: S-parameters for the LC-tank coil example. As can be seen the results obtained from Reactance (blue dashed) and Resistance (red dashed) extractions-based synthesis methods match the original raw Scattering parameters with great accuracy. A small part of the frequency response is shown since the response below $1GHz$ is very flat.

for the noise compliant synthesis. The accuracy of the synthesis methods in the calculation of the S-parameters is demonstrated in Figure 4.28. Noise analysis results are depicted in Figure 4.29.

The second example in Table 4.7 consider a 2-port base band filter block used in the receiver chain of a 3G transceiver. Also in this case noise-compliance is of paramount importance. All the components in a receiver chain are carefully designed in order to reduce noise contributions thus preserving the weak signal from the antenna. S-parameters accuracy is confirmed in Figure 4.30, while selected noise analysis results are reported in Figure 4.31.

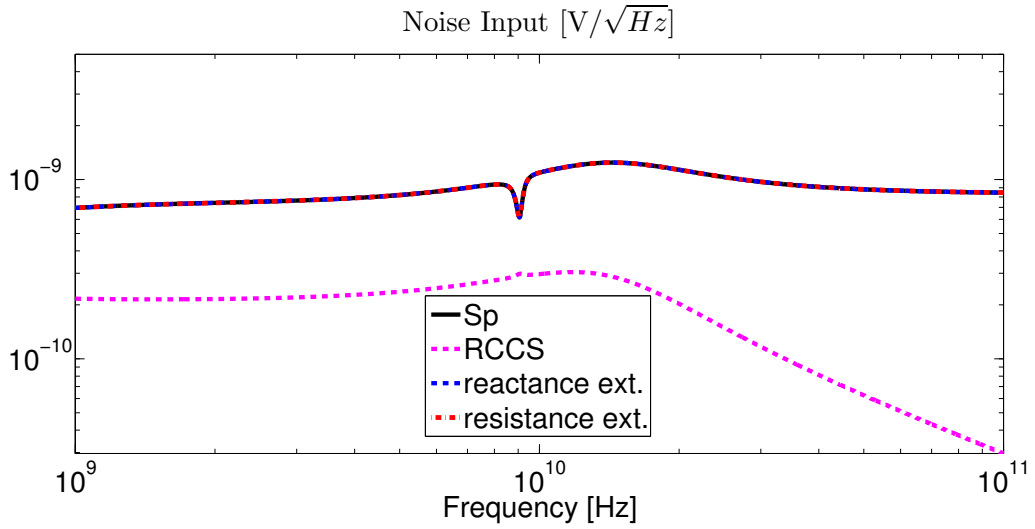

Figure 4.29: Noise analysis result for the LC-tank coil example. As can be seen the results obtained from Reactance (blue dashed) and Resistance (red dashed) extractions-based synthesis methods match (to machine precision) the result of the noise analysis obtained from Spice using the raw S-parameters (Sp, black continuous line). A small part of the frequency response is shown since the response below $1GHz$ is very flat.

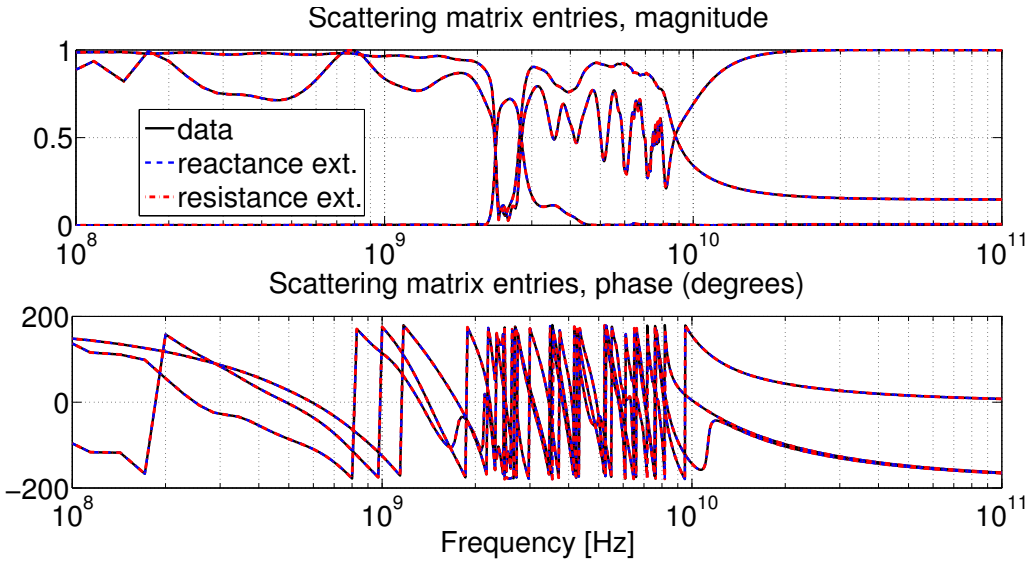

Figure 4.30: S-parameters for the base band filter example. As can be seen the results obtained from Reactance (blue dashed) and Resistance (red dashed) extractions-based synthesis methods match the original raw Scattering parameters with great accuracy.

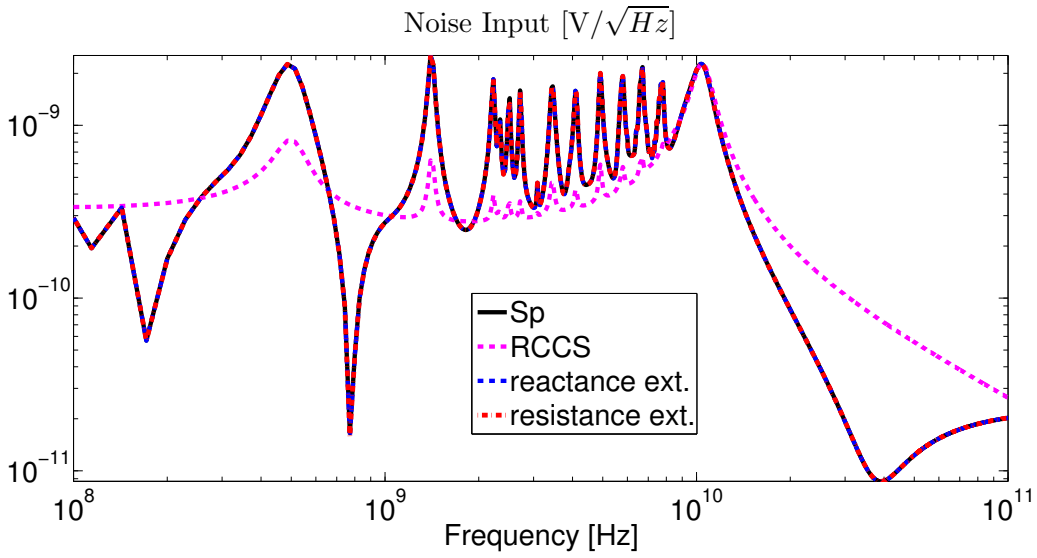

Figure 4.31: Noise analysis result for the base band filter example. As can be seen the results obtained from Reactance (blue dashed) and Resistance (red dashed) extractions-based synthesis methods match (to machine precision) the result of the noise analysis obtained from Spice using the raw S-parameters (Sp, black continuous line).

Table 4.7: Results summary for the two test cases based on real designs. The number of network elements: capacitors (cap), current controlled sources (cccs), inductors (ind), resistors (res), voltage controlled voltage sources (vcvs), voltage controlled current sources (vccs), is compared for each synthesis method. As a global estimate of network complexity the total number of circuit elements is also reported. The simulation time refers to the execution time of S-parameter analysis in a Spice solver.

order: 350 ports: 25	RCCS	Reactance extraction	Resistance extraction
nodes	400	1406	475
cap	350	177	350
cccs	–	53096	8631
ind	–	173	–
res	350	293	75
vcvs	–	540	50
vccs	9925	53096	26817
total	11025	107915	35923
time	20s	10min	1min
order: 248 ports: 2			
nodes	252	869	258
cap	248	124	248
cccs	–	20934	245
ind	–	124	–
res	248	493	6
vcvs	–	338	4
vccs	874	20934	1731
total	1622	–	2234
time	0.5s	1min	1.2s

4.4 Conclusions

Two noise compliant synthesis methods for linear behavioral macromodels based on the classical techniques have been presented: one preserves all the physical properties of the original system requiring a large number of network elements; the second misses to preserve reciprocity of the circuit blocks involved in the synthesis (while preserving noise behaviour) but requires less elements. Relying solely on the use of network elements possessing a proper noise model in SPICE based solvers, the proposed strategies are able to reproduce properly the noise behaviour of the system. The accuracy of the results obtained from the noise analysis was assessed by comparing the proposed synthesis with standard methods [176].

Noise compliance and network complexity have been the constraints of interest. Unfortunately those constraints can not jointly attain the optimum in the available synthesis. Indeed the direct state-space synthesis results into a network whose number of elements scales linearly with model order but that is not noise compliant, while Youla's synthesis is noise compliant but the number of network elements scales quadratically with model order. The best trade-off is provided by Darlington's resistance extraction, which is noise compliant, but in order to have a complexity comparable with the direct synthesis method requires the use of non-reciprocal elements. Therefore the best suited synthesis method should be selected depending on the application: for reduced order models of IC interconnects and parasitic networks for RF, SoC/SiP applications noise compliance is a must and Darlington's synthesis is the best choice. When dealing with large-scale packages, PCBs and transmission lines, the direct synthesis is best suited to tackle large order models but the resulting network will not be noise-compliant. Youla's synthesis is appropriate for those cases in which network reciprocity of all network component is needed.

The availability of a noise compliant network synthesis can be of paramount importance in analog behavioural modeling for devices and complete building blocks. Noise-preserving modeling is a must for simulation-based design and design verification purposes of complex analog systems. The methodology proposed here is an important step toward the migration of hard IP blocks into firm IP blocks, like discussed in Section (1.3.1).

Conclusions

This thesis proposed several improvements to various important steps in the verification flow of SoC/SiP components. All the issues tackled in this work raised from practical needs: fast identification and validation of component with large ports count, complexity reduction in system level simulations involving non-linear analog RF components and creation of versatile IP blocks to be used in a high proficiency IP reuse-based modern design flow.

For the identification of behavioural models associated to linear devices having hundreds of ports, an innovative algorithm was proposed and tested on several test cases of practical relevance. Combined with standard identification methods like Vector Fitting, the new procedure presented in this work attains speed-ups of two order of magnitudes in comparison with standard identification methods. Accuracy is completely under control and physical properties like passivity and stability can be easily enforced on commodity hardware relaying on a robust reformulation of common techniques.

Accurate small-signal models for RF analog non-linear components were derived considering that: several RF non-linear blocks are designed in order to behave in a linear way (no signal distortion or generation of spurious harmonics) under application-defined operating conditions, therefore non-linear effects are negligible and a small-signal model is accurate enough to mimic the response of the system. A simple and effective correction strategy was proposed in order to overcome the lack of accuracy at DC. The resulting models can substitute complex non-linear RF blocks resulting in a tremendous reduction of simulation time for system level simulations.

The small-signal models can then be extended by means of standard techniques to parameterized models. Application-defined operating conditions, temperature and other design parameters can be considered in order to enhance the applicability range. The application of the proposed methodology on real design test cases confirmed the quality of this approach.

Last but not the least, linear macromodels identified using the aforementioned techniques are converted (synthesized) in standard Spice netlists. In comparison with the most common synthesis methods, particular care was

devoted in this work in order to provide low-complexity Spice circuits reliable in all simulation scenarios. The availability of such versatile models converted to Spice netlists is the enabling factor for the migration of IP blocks from firm to hard form leading to a more robust design/verification flow.

The solutions proposed in this thesis met quality standards and expectations of the host institution, i.e. Intel Mobile Communications, and will likely become relevant part of a professional verification/design flow.

Appendix A

Notation, acronyms and symbols

Though already introduced in the text, notation, acronyms and symbols used in the thesis are summarized here for convenience of the reader.

Notation

constant identified by capital case letters (either Latin or Greek); example A. Some letters are reserved for particular definitions, like: number of circuit ports P.

index identified by lower case letters (either Latin or Greek); example a.

matrix identified by bold capital fonts; example $\mathbf{X}$. Some letters are reserved for particular applications, like $\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}$ which are associated to state-space models and $\mathbf{I}$, which is the identity matrix.

vector identified by bold lower case fonts; example $\boldsymbol{x}$.

Acronyms

3G	third Generation; refereed to Mobile communications standards.
A/D-D/A	Analog/Digital-Digital/Analog.
AC	Alternating Current.
AMS	Analog-Mixed Signal.

BB	Base Band.
BR	Bounded Real.
CAD	Computer Aided Design.
CB	Circuit Block.
CMOS	Complementary Metal-Oxide-Semiconductor.
CPU	Central Processing Unit.
DC	Direct Current.
DSP	Digital Signal Processing.
EDA	Electronic Design Automation.
GPU	Graphics Processing Unit.
GSK	Generalized Sanathanan-Koerner.
HDL	Hardware Description Language.
IC	Integrated Circuit.
IP	Intellectual Property.
ITRS	International Technology Roadmap for Semiconductors.
LDO	Low Drop-Out regulator.
LNA	Low Noise Amplifier.
LP	Low Power.
LTFM	Linear Transfer Function Model.
LTI	Linear Time Invariant.
MCM	Multi-Chip Module, alternative name for SiP.
NoC	Network on Chip.
OA,OpAmp,OPA	Operational Amplifier.
P-LTI	Parameterized-Linear Time Invariant.
PAS	Parallel Adaptive Sampling.
PC	Personal Computer.
PCB	Printed Circuit Board.
PDN	Power Delivery Network.

PI	Power Integrity.
PLL	Phase-Locked-Loop.
PR	Positive Real.
PWL	Piece-Wise-Linear.
RF	Radio Frequency.
SI	Signal Integrity.
SIA	Semiconductor Industry Association.
SiP	System in Package.
SK	Sanathanan-Koerner.
SoC	System on Chip.
SU	Speed Up.
SVD	Singular Values Decomposition.
TSV	Through-Silicon Via.
VF	Vector Fitting.

List of symbols

$\forall$	For all.
$\mathcal{O}(\cdot)$	Big O notation. Describes the limiting behaviour of a function when the argument tends towards a particular value or infinity.
$\sharp\Omega$	Cardinality (number of elements) in set Ω.
$\lceil x \rceil$	Maps the real number x to the smallest following integer.
x^*	Complex conjugate of x.
$\lambda(\mathbf{X})$	Set of all eigenvalues of matrix $\mathbf{X}$.
$\emptyset$	Empty set.
$\exists, \nexists$	There exists, there does not exist.
$\lfloor x \rfloor$	Maps the real number x to the largest previous integer.
$\mathbf{X}^H$	Conjugate-transpose of matrix $\mathbf{X}$.

$\mathbf{I}_n$	Identity matrix of dimension n.
iff	If and only if.
$\mathbb{I}\mathrm{m}\{x\}$	Imaginary part of complex number x.
$\jmath$	Imaginary unit.
$\in$	Is an element of.
$\cap$	Set intersection.
$\otimes$	Kronecker product.
$\mathrm{mat}(\boldsymbol{x})$	Transforms the vector $\boldsymbol{x}$ in a matrix $\mathbf{X}$ stacking sub-blocks of $\boldsymbol{x}$ as columns of $\mathbf{X}$ with proper dimensions.
$\max \Omega$	Maximum, the largest element of set Ω.
$\min \Omega$	Minimum, the smallest element of Ω.
$\mathrm{mod}(x)$	Remainder of division $x/2$.
$\mathbf{X} \prec 0, \mathbf{X} \preceq 0$	$\mathbf{X}$ is negative (semi)definite.
$\mathbf{X} \succ 0, \mathbf{X} \succeq 0$	$\mathbf{X}$ is positive (semi)definite.
$\prod$	Product.
$\mathbf{X}^\dagger$	Moore-Penrose pseudoinverse of matrix $\mathbf{X}$.
$\mathbb{R}\mathrm{e}\{x\}$	Real part of complex number x.
$\sigma(\mathbf{X})$	Set of all singular values of matrix $\mathbf{X}$.
$\|\mathbf{X}\|_2$	Spectral norm of matrix $\mathbf{X}$, defined as $\max \sigma(\mathbf{X})$.
s.t.	Subject to.
$\subset, \subseteq$	Is a subset of.
$\sum$	Sum.
$\mathbb{C}$	Set of complex numbers.
$\|\mathbf{X}\|_F$	Frobenius norm of matrix $\mathbf{X}$.
$\mathbb{R}$	Set of real numbers.
$\mathrm{tr}(\mathbf{X})$	The trace of a square matrix $\mathbf{X}$ is defined as the sum of the elements on the main diagonal of $\mathbf{X}$.
$\mathbf{X}^T$	Transpose of matrix $\mathbf{X}$.

$\cup$	Set union.
$\text{vec}(\mathbf{X})$	Vector stacking all columns of matrix $\mathbf{X}$.

Appendix B

The Vector Fitting algorithm

The starting point for the identification of a rational macromodel is a set of samples from the system frequency response of a P-port LTI device:

$$\mathbf{H}_l \in \mathbb{C}^{P\times P}, \quad \{\omega_l\} \quad l = 1, \ldots, L. \tag{B.1}$$

The model resulting from the application of the VF algorithm will be in pole-residue form

$$\mathbf{H}(s) = \mathbf{D} + \sum_{n=1}^{N} \frac{\mathbf{R}_n}{s - p_n}. \tag{B.2}$$

The main goal is to minimize the approximation error in a generic norm

$$\min \|\mathbf{H}(s_l) - \mathbf{H}_l\| \ \forall l \tag{B.3}$$

Since the model (B.2) requires the identification of poles p_n and residues $\mathbf{R}_n$, the resulting minimization problem (B.3) will be non-linear. As discussed in Chapter 2, Section 2.2, Vector Fitting (VF) uses the Generalized Sanathanan-Koerner (GSK) Iteration to avoid dealing with the non-linear minimization problem (B.3) thus converting it into a sequence of linear problems.

The original version of the VF algorithm as proposed by [62] is now presented as a step-by-step procedure:

1. Starting poles selection: choose an arbitrary[1] set of poles q_j with $j = 1, \ldots, N$;

[1] Some hints on how to choose the starting poles can be found in [62].

2. Using the poles q_j define two rational functions

$$\sigma \mathbf{H}^i(s) = \mathbf{D}^i + \sum_{n=1}^{N} \frac{\mathbf{N}_n^i}{s - q_n^i}, \tag{B.4}$$

$$\sigma^i(s) = 1 + \sum_{n=1}^{N} \frac{d_n^i}{s - q_n^i}. \tag{B.5}$$

Those will be numerator and denominator of the estimated model at iteration i, i.e.

$$\mathbf{H}^i(s) = \frac{\sigma \mathbf{H}^i(s)}{\sigma^i(s)}; \tag{B.6}$$

3. based on the GSK iteration construct a linear system using the equation

$$\sigma \mathbf{H}^i(\jmath\omega_l) \simeq \mathbf{H}_l \sigma^i(\jmath\omega_l), \tag{B.7}$$

and solve it in the least-square sense for $\mathbf{N}_n^i$ and d_n^i;

4. poles relocation: update the starting set of poles in (B.4) and (B.5) using the zeros of $\sigma^i(s)$ and force them to have negative real part (poles flipping);

5. iterate steps (2)-(4), $i = i + 1$, until $\sigma^i(s) \to 1$. This is the main iteration for the identification of model poles.

6. When the poles relocation procedure converged, set the poles of (B.4) as $p_n = q_n^i$.

7. Finally solve the equation

$$\mathbf{D}^i + \sum_{n=1}^{N} \frac{\mathbf{R}_n}{s - q_n^i} \simeq \mathbf{H}_l \tag{B.8}$$

in least-square sense to identifies the residues $\mathbf{R}_n$ of the final macro-model.

Appendix C

RC-example state-space derivation

The state-space equations associated to the impedance transfer function Z_{out} from the RC circuit of Figure 4.12 can be extracted by direct inspection, considering the RC network in Figure C.1.

$$\begin{cases} C\dot{v}_1 &= -\frac{v_1}{R} - \frac{v_1 - v_o}{R}, \\ C\dot{v}_o &= i_s + \frac{v_1 - v_o}{R}. \end{cases} \tag{C.1}$$

A simple rearrangement of the equations in (C.1), i.e.

$$\begin{cases} \dot{v}_1 &= -\frac{2}{RC}v_1 + \frac{1}{RC}v_o, \\ \dot{v}_o &= \frac{1}{RC}v_1 - \frac{1}{RC}v_o + \frac{i_s}{C}, \end{cases} \tag{C.2}$$

makes possible to identify the state vector

$$\boldsymbol{x} = \begin{pmatrix} v_1 \\ v_o \end{pmatrix}, \tag{C.3}$$

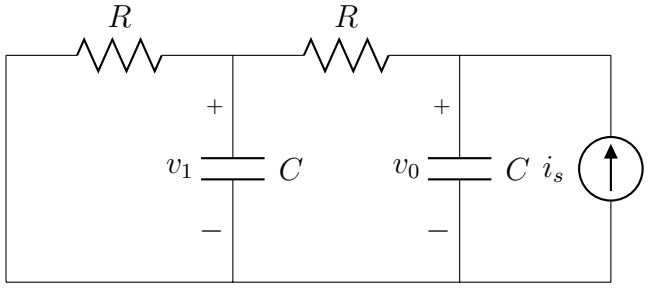

Figure C.1: RC ladder circuit for the identification of a state-space model using the direct inspection method.

and the matrices (normalized by C) associated to the state variables

$$\mathbf{A} = \frac{1}{R}\begin{pmatrix} -2 & 1 \\ 1 & -1 \end{pmatrix} \quad \boldsymbol{b} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \quad \boldsymbol{c} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}^T \quad \boldsymbol{d} = 0 \tag{C.4}$$

leading to the system of differential equations

$$\begin{cases} \dot{\boldsymbol{x}} &= \mathbf{A}\boldsymbol{x} + \boldsymbol{b}i \\ v_o &= \boldsymbol{c}\boldsymbol{x}. \end{cases} \tag{C.5}$$

with transfer function

$$Z(s) = \boldsymbol{d} + \boldsymbol{c}(sC\mathbf{I} - \mathbf{A})^{-1}\boldsymbol{b}. \tag{C.6}$$

The state-space equations obtained using the direct inspection method can be converted to an equivalent state-space form, i.e. the Gilbert canonical form [170]. In this representation the state-space matrix $\tilde{\mathbf{A}}$ is a (block) diagonal matrix with the eigenvalues of the original $\mathbf{A}$ matrix (C.4) on the main diagonal. As a consequence the Gilbert canonical form is equivalent to the decomposition of the transfer function (C.6) in the sum of rational functions, i.e.

$$\xi(u) = \frac{2+u}{u^2+3u+1} = \frac{\rho_1}{u-p_1} + \frac{\rho_2}{u-p_2}, \tag{C.7}$$

where $u = RCs$, $\xi = Z(s)/R$ and $p_{1,2}$ are the poles of $\xi(u)$ (zeros of $u^2 + 3u + 1$), while $\rho_{1,2}$ are the residues of $\xi(u)$

$$\rho_{1,2} = \xi(p_{1,2}). \tag{C.8}$$

Therefore the poles can be calculated from $u^2 + 3u + 1$

$$p_{1,2} = \frac{-3 \pm \sqrt{9-4}}{4} = \frac{-3 \pm \sqrt{5}}{2} \tag{C.9}$$

and the residues follows from (C.8)

$$\rho_{1,2} = \frac{5 \pm \sqrt{5}}{10}\,. \tag{C.10}$$

A simple way to construct a diagonal canonical form consists in collecting the residues in the state-space matrix $\tilde{\boldsymbol{c}}$, the poles on the main diagonal of $\tilde{\mathbf{A}}$ and ones in $\tilde{\boldsymbol{b}}$, i.e.

$$\tilde{\mathbf{A}} = \frac{1}{R}\begin{pmatrix} p_1 & 0 \\ 0 & p_2 \end{pmatrix} \quad \tilde{\boldsymbol{b}} = \begin{pmatrix} 1 \\ 1 \end{pmatrix} \quad \tilde{\boldsymbol{c}} = \begin{pmatrix} \rho_1 \\ \rho_2 \end{pmatrix}^T \tag{C.11}$$

corresponding to

$$Z(s) = \frac{R\rho_1}{sCR - p_1} + \frac{R\rho_2}{sCR - p_2} \tag{C.12}$$

where the physical dimensions are consistent.

Bibliography

[1] A. S. Grove, *Only the Paranoid Survive: How to Exploit the Crisis Points That Challenge Every Company*. New York: Random House, 2010 (cit. on p. 1).

[2] G. E. Moore, "Cramming more components onto integrated circuits", *Electronics*, vol. 38, no. 8, pp. 1–4, 1965 (cit. on p. 1).

[3] R. E. Jones, R. Chatterjee, and S. K. Pozder, "Technology and application of 3d interconnect", in *Proceedings of IEEE International Conference Integrated Circuit Design and Technology*, Austin, TX, 2007, pp. 176–180 (cit. on p. 2).

[4] S. K. Pozder and R. E. Jones, "Status and outlook", in *Wafer Level 3-D ICs Process Technology*. Springer, 2008, ch. 15 (cit. on p. 2).

[5] A. Papanikolaou, D. Soundris, and R. Radojcic, "Introduction to three-dimensional integration", in *Three Dimensional System Integration: IC Stacking Process and Design*. Springer, 2011, ch. 1 (cit. on p. 2).

[6] K. S. Yeo, M. A. Do, and C. C. Boon, *Design of CMOS RF Integrated Circuits and Systems*. World Scientific, 2010 (cit. on p. 2).

[7] J. R. Baker, *CMOS, Mixed-Signal Circuit Design*, ser. IEEE Press Series on Microelectronic Systems. Wiley, 2009 (cit. on p. 3).

[8] P. W. Hen-Sum, "Nanoscale cmos", *Proceedings of the IEEE*, vol. 87, no. 4, pp. 537 –570, 1999 (cit. on p. 3).

[9] S. W. Director, W. Maly, and A. J. Strojwas, *VLSI Design for Manufacturing: Yield Enhancement*. Springer, 1989 (cit. on p. 3).

[10] Y. Sun, *Test and Diagnosisi of Analogue, Mixed-signal and RF Integrated Circuits: The system on chip approach*, Y. Sun, Ed., ser. Circuits, Devices and Systems. The Institution of Engineering and Technology, 2008 (cit. on p. 3).

[11] A. Fayed and M. Ismail, *Adaptive techniqies for mixed signal systems on chip*, ser. The International Series in Engineering and Computer Science. Springer, 2006 (cit. on p. 3).

[12] L.-T. Wang, C. Stroud, and N. Touba, *System-on-Chip Test Architectures: Nanometer Design for Testability*. Elsavier, 2008 (cit. on p. 3).

[13] U. Alvarado, G. Bistué, and I. Adín, *Low Power RF Circuit Design in Standard CMOS Technology*, ser. Lecture Notes in Electrical Engineering. Springer, 2011, vol. 104 (cit. on pp. 3, 5).

[14] G. Schrom and S. Siegfried, "Ultra-low-power cmos technologies", in *International Semiconductor Conference*, vol. 1, Sinaia, Romania, 1996, pp. 237–246 (cit. on p. 3).

[15] I. Novak, Ed., *Power Distribution Network Design Methodologies*, 2008 (cit. on p. 4).

[16] A. V. Mezhiba and E. G. Friedman, *Power Distribution Networks in High Speed Integrated Circuits*. Kluwer Academic, 2004 (cit. on p. 4).

[17] M. Swaminathan, J. Kim, I. Novak, and J. Libous, "Power distribution networks for system-on-package: status and challenges", *Advanced Packaging, IEEE Transactions on*, vol. 27, no. 2, pp. 286–300, 2004 (cit. on p. 4).

[18] L. Smith, R. Anderson, D. Forehand, T. Pelc, and T. Roy, "Power distribution system design methodology and capacitor selection for modern cmos technology", *Advanced Packaging, IEEE Transactions on*, vol. 22, no. 3, pp. 284–291, 1999 (cit. on p. 4).

[19] R. Vishweshwara, R. Venkatraman, H. Udayakumar, and N. Arvind, "An approach to measure the performance impact of dynamic voltage fluctuations using static timing analysis", in *VLSI Design, 2009 22nd International Conference on*, 2009, pp. 519–524 (cit. on p. 4).

[20] J. P. Libous, "High performance asic chip power distribution design and analysis, short course", in *Topical Meeting on Electrical Performance of Electronic Packaging*, 2002 (cit. on p. 4).

[21] B. Amelifard and M. Pedram, "Design of an efficient power delivery network in an soc to enable dynamic power management", in *Low Power Electronics and Design (ISLPED), 2007 ACM/IEEE International Symposium on*, 2007, pp. 328–333 (cit. on p. 4).

[22] S. Kang and Y. Lebledici, *CMOS Digital Integrated Circuits*. McGraw-Hill Companies, 1999 (cit. on p. 4).

[23] C. Duan, B. J. LaMeres, and S. P. Khatri, *On and Off-Chip Crosstalk Avoidance in VLSI Design*. Springer, 2010 (cit. on p. 4).

[24] S. Khatri, A. Mehrotra, R. K. Brayton, A. Sangiovanni-vincentelli, and R. H. J. M. Otten, "A novel vlsi layout fabric for deep sub-micron applications", in *Design Automation Conference*, 1999, pp. 507–510 (cit. on p. 4).

[25] P. P. Sotiriadis and A. Chandrakasan, "Low power bus coding techniques considering inter-wire capacitance", in *Custom Integrated Circuits Conference*, Orlando, FL, 2000, pp. 507–510 (cit. on p. 4).

[26] L. He, S. Elassaad, and Y. Shi, *System-in-package: Electrical and Layout Perspectives*, ser. Foundations and Trends in Electronic Design Automation. Now, 2010, vol. 4 (cit. on p. 5).

[27] S. Haykin, *Communication systems*. John Wiley e Sons, 2001 (cit. on p. 5).

[28] J. B. Kuo and K.-W. Su, *CMOS VLSI Engineering: Silicon-on-Insulator (SOI)*. Kluwer Academic, 1998 (cit. on p. 5).

[29] A. Papanikolaou, D. Soudris, and R. Radojcic, Eds., *Three Dimensional System Integration: IC Stacking Process and Design*, Springer, 2011 (cit. on p. 5).

[30] A. Sheibanyrad, F. Pétrot, and A. Jantsch, Eds., *3D Integration for NoC-based SoC Architectures*, Springer, 2011 (cit. on p. 5).

[31] L. P. Carloni, P. Pande, and Y. Xie, "Networks-on-chip in emerging interconnect paradigms: advantages and challenges", in *Proceedings of the 2009 3rd ACM/IEEE International Symposium on Networks-on-Chip*, ser. NOCS '09, Washington, DC, USA: IEEE Computer Society, 2009, pp. 93–102 (cit. on p. 5).

[32] D. A. Mathaikutty and S. Shukla, *Metamodeling-driven IP Reuse for SoC Integration and Microprocessor Design*. Artech House, 2009 (cit. on pp. 5, 6, 14).

[33] E. M. Grant and H. Chang, Eds., *Winning the SoC Revolution: Experiences in Real Design*, Kluwer Academic, 2003 (cit. on pp. 6, 8).

[34] M. Keating and B. Pierre, *Reuse Methodology Manual*. Kluwer Academic, 2002 (cit. on p. 8).

[35] H. Chang, L. Cooke, M. Hunt, G. Martin, A. McNelly, and L. Todd, *Surviving the SoC Revolution: A Guide to Platform-Based Design*. Kluwer Academic, 1999 (cit. on p. 9).

[36] R. Saleh, S. Wilton, S. Mirabbasi, A. Hu, M. Greenstreet, G. Lemieux, P. P. Pande, C. Grecu, and A. Ivanov, "System-on-chip: reuse and integration", *Proceedings of the IEEE*, vol. 94, no. 6, pp. 1050 –1069, 2006 (cit. on p. 9).

[37] M. Keating, *The Simple Art of SoC Design: Closing the Gap between RTL and ESL*. Springer, 2011 (cit. on p. 9).

[38] K. Kundert, H. Chang, D. Jefferies, G. Lamant, E. Malavasi, and F. Sendig, "Design of mixed-signal systems-on-a-chip", *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol. 19, no. 12, pp. 1561–1571, 2000 (cit. on p. 9).

[39] G. Gielen and R. Rutenbar, "Computer-aided design of analog and mixed-signal integrated circuits", *Proceedings of the IEEE*, vol. 88, no. 12, pp. 1825–1854, 2000 (cit. on p. 9).

[40] N. Madrid, E. Peralias, A. Acosta, and A. Rueda, "Analog/mixed-signal ip modeling for design reuse", in *Design, Automation and Test in Europe, 2001. Conference and Exhibition 2001. Proceedings*, 2001, pp. 766–767 (cit. on p. 9).

[41] L. Zheying, L. Li, and Y. Jiren, "A study on analog ip blocks for mixed-signal soc", in *ASIC, 2003. Proceedings. 5th International Conference on*, vol. 1, 2003, 564–567 Vol.1 (cit. on p. 9).

[42] M. Hamour, R. Saleh, S. Mirabbasi, and A. Ivanov, "Analog ip design flow for soc applications", in *Proceedings of the 2003 IEEE International Symposium on Circuits and Systems*, vol. vol.4, 2003, pp. 676–9 (cit. on p. 9).

[43] N. Codreanu, C. Ionescu, P. Svasta, I. Plotog, and V. Vulpe, "Accurate 3d modelling and simulation of advanced packages and vertical stacked dice", in *Electronics System-Integration Technology Conference, 2008. ESTC 2008. 2nd*, 2008, pp. 857–862 (cit. on pp. 10, 11).

[44] F. Awwad, M. Nekili, V. Ramachandran, and M. Sawan, "On modeling of parallel repeater-insertion methodologies for soc interconnects", *Circuits and Systems I: Regular Papers, IEEE Transactions on*, vol. 55, no. 1, pp. 322–335, 2008 (cit. on p. 10).

[45] P. Collins, "How are we going to test soc's on a pcb?", in *Test Conference, 2005. Proceedings. ITC 2005. IEEE International*, 2005, 1 pp.–1264 (cit. on p. 10).

[46] H. Tohya and N. Toya, "A novel design methodology of the on-chip power distribution network enhancing the performance and suppressing emi of the soc", in *Circuits and Systems, 2007. ISCAS 2007. IEEE International Symposium on*, 2007, pp. 889–892 (cit. on p. 10).

[47] K. Salah, A. EL-Rouby, H. Ragai, and Y. Ismail, "3d/tsv enabling technologies for soc/noc: modeling and design challenges", in *Microelectronics (ICM), 2010 International Conference on*, 2010, pp. 268–271 (cit. on p. 10).

[48] T. Bashkow, "The a matrix, new network description", *Circuit Theory, IRE Transactions on*, vol. 4, no. 3, pp. 117–119, 1957 (cit. on pp. 10, 17, 102).

[49] G. Pick, "Über die beschränkungen analytischer funktionen, welche durch vorgegebene funktionswerte bewirkt werden", *Annals of Mathematics*, vol. 77, pp. 7–23, 1916 (cit. on p. 10).

[50] R. Nevanlinna, "Über beschränkte funktionen die in gegebenen punkten vorgeschreibene funktionswerte bewirkt werden", *Anal. Acad. Sci. Fenn., Ser. A.*, vol. 13, pp. 1–71, 1919 (cit. on p. 10).

[51] P. Delsarte, Y. Genin, and Y. Kamp, "The nevanlinna-pick problem for matrix-valued functions", *Journal on Applied Mathematics*, vol. 36, no. 1, pp. 47–61, 1979 (cit. on p. 10).

[52] P. I. Fedcina, "Descriptions of solutions of the tangential nevanlinna-pick problem", *Akademii Nauk Armyanskoj*, vol. 60, pp. 37–42, 1975 (cit. on p. 10).

[53] A. Tannenbaum, "Spectral nevanlinna-pick interpolation theory and robust stabilization", in *Decision and Control, 1987. 26th IEEE Conference on*, vol. 26, 1987, pp. 1635–1636 (cit. on p. 10).

[54] C. Coelho, J. Phillips, and L. Miguel Silveira, "Passive constrained rational approximation algorithm using nevanlinna-pick interpolation", in *Design, Automation and Test in Europe Conference and Exhibition, 2002. Proceedings*, 2002, pp. 923–930 (cit. on p. 10).

[55] J. A. Ball, I. Gohberg, and I. Rodman, *Interpolation of Rational Matrix Functions*, ser. Operator Theory: Advances and Applications. Birkhäuser, 1990, vol. 45 (cit. on p. 10).

[56] K. Löwner, "Uber monotone matrix funktionen", *Math. Z.*, vol. 38, pp. 177–216, 1934 (cit. on p. 12).

[57] V. Belevitch, "Interpolation matrices", Philips, Research Report 25, 1970, pp. 337–369 (cit. on p. 12).

[58] C. Beattie and S. Gugercin, "Realization-independent h2-approximation", in *Decision and Control (CDC), 2012 IEEE 51st Annual Conference on*, 2012, pp. 4953–4958 (cit. on p. 12).

[59] S. Lefteriu and A. Antoulas, "A new approach to modeling multiport systems from frequency-domain data", *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol. 29, no. 1, pp. 14–27, 2010 (cit. on p. 12).

[60] C. Sanathanan and J. Koerner, "Transfer function synthesis as a ratio of two complex polynomials", *Automatic Control, IEEE Transactions on*, vol. 8, pp. 56–58, 1963 (cit. on pp. 12, 20, 21, 81).

[61] E. C. Levy, "Complex-curve fitting", *Automatic Control, IRE Transactions on*, vol. AC-4, no. 1, pp. 37–43, 1959 (cit. on pp. 12, 20).

[62] B. Gustavsen and A. Semlyen, "Rational approximation of frequency domain responses by vector fitting", *Power Delivery, IEEE Transactions on*, vol. 14, no. 3, pp. 1052–1061, 1999, ISSN: 0885-8977 (cit. on pp. 12, 15, 20, 22, 23, 33, 51, 68, 71, 79, 86, 121, 163).

[63] S. Grivet-Talocia, "Package macromodeling via time-domain vector fitting", *Microwave and Wireless Components Letters, IEEE*, vol. 13, no. 11, pp. 472–474, 2003 (cit. on p. 12).

[64] S. Lefteriu and A. Antoulas, "On the convergence of the vector-fitting algorithm", *Microwave Theory and Techniques, IEEE Transactions on*, vol. 61, no. 4, pp. 1435–1443, 2013 (cit. on p. 12).

[65] S. Grivet-Talocia and M. Bandinu, "Improving the convergence of vector fitting for equivalent circuit extraction from noisy frequency responses", *Electromagnetic Compatibility, IEEE Transactions on*, vol. 48, no. 1, pp. 104–120, 2006 (cit. on p. 12).

[66] H Padé, "Sur la répresentation approchée d'une fonction par des fractions rationelles", Master's thesis, École normale supérieure, 1892 (cit. on p. 12).

[67] V. Zakian, "Rational approximants to the matrix exponential", *Electronics Letters*, vol. 6, 814 Ű 815, 1970 (cit. on p. 12).

[68] M. Celik, O. Ocali, M. Tan, and A. Atalar, "Pole-zero computation in microwave circuits using multipoint pade approximation", *Circuits and Systems I: Fundamental Theory and Applications, IEEE Transactions on*, vol. 42, no. 1, pp. 6–13, 1995 (cit. on p. 12).

[69] N. Xin-Peng, Z. Ping, W. You-hua, and Z. Hai-jiao, "Research for approximation algorithm of high-frequency model of transformer windings", in *Control and Decision Conference, 2009. CCDC '09. Chinese*, 2009, pp. 5502–5506 (cit. on p. 12).

[70] P. Feldmann and R. Freund, "Efficient linear circuit analysis by pade approximation via the lanczos process", *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol. 14, no. 5, pp. 639–649, 1995 (cit. on p. 12).

[71] R. Shi and J. Macgregor, "A framework for subspace identification methods", in *American Control Conference, 2001. Proceedings of the 2001*, vol. 5, 2001, pp. 3678–3683 (cit. on p. 12).

[72] M. Bergamasco and M. Lovera, "Continuous-time predictor-based subspace identification using laguerre filters", *Control Theory Applications, IET*, vol. 5, no. 7, pp. 856–867, 2011 (cit. on p. 12).

[73] H. Oku and T. Fujii, "Direct subspace model identification of lti systems operating in closed-loop", in *Decision and Control, 2004. CDC. 43rd IEEE Conference on*, vol. 2, 2004, pp. 2219–2224 (cit. on p. 12).

[74] I. Jamaludin, N. Wahab, M. Rahmat, and S. Sahlan, "Online subspace identification methods for mimo model", in *Control, Systems Industrial Informatics (ICCSII), 2012 IEEE Conference on*, 2012, pp. 22–25 (cit. on p. 12).

[75] A. Dounavis, R. Achar, and M. Nakhla, "Efficient passive circuit models for distributed networks with frequency-dependent parameters", *Advanced Packaging, IEEE Transactions on*, vol. 23, no. 3, pp. 382–392, 2000 (cit. on p. 13).

[76] A. Chinea, P. Triverio, and S. Grivet-Talocia, "Delay-based macromodeling of long interconnects from frequency-domain terminal responses", *Advanced Packaging, IEEE Transactions on*, vol. 33, no. 1, pp. 246–256, 2010 (cit. on p. 13).

[77] F.-B. Yeh, C.-C. Lin, and H.-N. Huang, "Parametric nevanlinna-pick interpolation theory", in *Proc. Natl. Sci. Counc. ROC (A)*, 1998 (cit. on p. 13).

[78] R. A. Maher and R. Samir, "Robust stability of a class of unstable systems under mixed uncertainty", *J. Control Sci. Eng.*, vol. 2011, 28:28–28:28, 2011 (cit. on p. 13).

[79] S. Lefteriu, A. Antoulas, and A. C. Ionita, "Parametric model order reduction from measurements", in *Electrical Performance of Electronic Packaging and Systems (EPEPS), 2010 IEEE 19th Conference on*, 2010, pp. 193–196 (cit. on p. 13).

[80] P. Triverio, M. Nakhla, and S. Grivet-Talocia, "Parametric macromodeling of multiport networks from tabulated data", in *Electrical Performance of Electronic Packaging, 2007 IEEE*, 2007, pp. 51–54 (cit. on pp. 13, 79, 80).

[81] P. Triverio, S. Grivet-Talocia, and M. Nakhla, "On the construction of uniformly stable multivariate interconnect macromodels", in *Signal Propagation on Interconnects, 2009. SPI '09. IEEE Workshop on*, 2009, pp. 1–4 (cit. on pp. 13, 81, 85).

[82] P. Triverio, M. Nakhla, and S. Grivet-Talocia, "Passive parametric macromodeling from sampled frequency data", in *Signal Propagation on Interconnects (SPI), 2010 IEEE 14th Workshop on*, 2010, pp. 117–120 (cit. on pp. 13, 66, 79, 80).

[83] F. Ferranti, "Parameterized macromodeling and model order reduction of high-speed interconnects", PhD thesis, Ghent University (Belgium), 2011 (cit. on p. 13).

[84] F. Ferranti, M. Nakhla, G. Antonini, T. Dhaene, L. Knockaert, and A. Ruehli, "Parameterized model order reduction of delayed systems using an interpolation approach with amplitude and frequency scaling coefficients", in *Signal and Power Integrity (SPI), 2012 IEEE 16th Workshop on*, 2012, pp. 57–60 (cit. on p. 13).

[85] L. Daniel, O. C. Siong, L. S. Chay, K. H. Lee, and J. White, "A multiparameter moment-matching model-reduction approach for generating geometrically parameterized interconnect performance models", *Trans. Comp.-Aided Des. Integ. Cir. Sys.*, vol. 23, no. 5, pp. 678–693, 2006, ISSN: 0278-0070 (cit. on p. 13).

[86] T. Bui-Thanh, K. Willcox, and O. Ghattas, "Model reduction for large-scale systems with high-dimensional parametric input space", *SIAM Journal on Scientific Computing*, vol. 30, pp. 3270–3288, 2008 (cit. on p. 13).

[87] B. Bond and L. Daniel, "A piecewise-linear moment-matching approach to parameterized model-order reduction for highly nonlinear systems", *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol. 26, no. 12, pp. 2116–2129, 2007 (cit. on pp. 13, 14).

[88] M. Buchholz and S. Werner, "Recursive subspace identification of linear parameter-varying systems", in *American Control Conference (ACC), 2012*, 2012, pp. 3408–3414 (cit. on p. 13).

[89] P. M. O. Gebraad, J. Van Wingerden, G. Van der Veen, and M. Verhaegen, "Lpv subspace identification using a novel nuclear norm regularization method", in *American Control Conference (ACC), 2011*, 2011, pp. 165–170 (cit. on p. 13).

[90] S. Olivadese, P. Brenner, and S. Grivet-Talocia, "Dc-compliant small-signal macromodels of non-linear circuit blocks", in *Signal and Power Integrity (SPI), 2013 17th IEEE Workshop on*, 2013, pp. 1–4 (cit. on pp. 13, 82).

[91] P. Drazin, *Nonlinear Systems*. Cambridge university press, 1992 (cit. on p. 14).

[92] T. E. Stern, "Piecewise-linear network theory", Research Laboratory of Electronics, Massachusetts Institute of Technology (MIT), Tech. Rep., 1956 (cit. on p. 14).

[93] L. Chua and A.-C. Deng, "Canonical piecewise-linear modeling", *Circuits and Systems, IEEE Transactions on*, vol. 33, no. 5, pp. 511–525, 1986 (cit. on p. 14).

[94] T. A. M. Kevenaar and D. Leenaerts, "A comparison of piecewise-linear model descriptions", *Circuits and Systems I: Fundamental Theory and Applications, IEEE Transactions on*, vol. 39, no. 12, pp. 996–1004, 1992 (cit. on p. 14).

[95] Y. Zhang, S. Sankaranarayanan, and F. Somenzi, "Piecewise linear modeling of nonlinear devices for formal verification of analog circuits", in *Formal Methods in Computer-Aided Design (FMCAD), 2012*, 2012, pp. 196–203 (cit. on p. 14).

[96] K. Gharaibeh, "Behavioral modeling of nonlinear power amplifiers using threshold decomposition-based piece wise linear approximation", in *Radio and Wireless Symposium, 2008 IEEE*, 2008, pp. 755–758 (cit. on p. 14).

[97] D. I. Long, "Behavioural modelling of mixed-signal circuits using pwl waveforms", in *Mixed-Signal AHDL/VHDL Modelling and Synthesis (Ref. No: 1997/331), IEE Colloquium on*, 1997, pp. 2/1–2/6 (cit. on p. 14).

[98] D. Deschrijver, M. Mrozowski, T. Dhaene, and D. De Zutter, "Macromodeling of multiport systems using a fast implementation of the vector fitting method", *Microwave and Wireless Components Letters, IEEE*, vol. 18, no. 6, pp. 383–385, 2008 (cit. on pp. 15, 22).

[99] A. Chinea and S. Grivet-Talocia, "A parallel vector fitting implementation for fast macromodeling of highly complex interconnects", in *Electrical Performance of Electronic Packaging and Systems (EPEPS), 2010 IEEE 19th Conference on*, 2010, pp. 101–104 (cit. on p. 15).

[100] A. Chinea, S. Grivet-Talocia, S. Olivadese, and L. Gobbato, "High-performance passive macromodeling algorithms for parallel computing platforms", *Components, Packaging and Manufacturing Technology, IEEE Transactions on*, vol. 3, no. 7, pp. 1188–1203, 2013 (cit. on p. 15).

[101] S. Grivet-Talocia and A. Ubolli, "A comparative study of passivity enforcement schemes for linear lumped macromodels", *Advanced Packaging, IEEE Transactions on*, vol. 31, no. 4, pp. 673–683, 2008 (cit. on p. 15).

[102] ——, "On the generation of large passive macromodels for complex interconnect structures", *Advanced Packaging, IEEE Transactions on*, vol. 29, no. 1, pp. 39–54, 2006 (cit. on pp. 15, 54).

[103] S. Grivet-Talocia, "On passivity characterization of symmetric rational macromodels", *Microwave Theory and Techniques, IEEE Transactions on*, vol. 58, no. 5, pp. 1238–1247, 2010 (cit. on pp. 15, 119).

[104] ——, "An adaptive sampling technique for passivity characterization and enforcement of large interconnect macromodels", *Advanced Packaging, IEEE Transactions on*, vol. 30, no. 2, pp. 226–237, 2007 (cit. on pp. 15, 33, 39, 48, 52, 54, 57).

[105] L. Gobbato, A. Chinea, and S. Grivet-Talocia, "A parallel hamiltonian eigensolver for passivity characterization and enforcement of large interconnect macromodels", in *Design, Automation Test in Europe Conference Exhibition (DATE), 2011*, 2011, pp. 1–6 (cit. on p. 15).

[106] T.-Y. Wu, S. Gharahi, and J. Abraham, "An area efficient on-chip static ir drop detector/evaluator", in *Circuits and Systems, 2009. ISCAS 2009. IEEE International Symposium on*, 2009, pp. 2009–2012 (cit. on p. 16).

[107] B. McMillan, "Introduction to formal realizability theory", Bell Labs, Tech. Rep., 1952 (cit. on pp. 18, 102).

[108] H. H. Rosenbrock, *State-Space and Multivariable Theory.* Thomas Nelson and Sons, 1970 (cit. on p. 18).

[109] V. Belevitch, *Classical network theory.* Holden-Day, 1968 (cit. on pp. 19, 102, 103, 107, 109, 110, 112, 113).

[110] V. L. C. F. Golub G. H., *Matrix computations*, Third edition. Johns Hopkins Press, 1996 (cit. on pp. 21, 24, 130).

[111] N. J. Higham, *Accuracy and Stability of Numerical Algorithms*, 2nd. SIAM, 2002 (cit. on pp. 21, 113).

[112] P. Triverio, "Self consistent, efficient and parametric macromodels for high-speed interconnects design", PhD thesis, Politecnico di Torino, 2009 (cit. on pp. 21, 66, 79, 80).

[113] C. Eckart and G. Young, "The approximation of one matrix by another of lower rank", English, *Psychometrika*, vol. 1, no. 3, pp. 211–218, 1936, ISSN: 0033-3123 (cit. on p. 25).

[114] R. M. Larsen, "Computing the svd for large and sparse matrices", Stanform University, Tech. Rep., 2000 (cit. on pp. 25, 31, 32).

[115] P. Triverio, S. Grivet-Talocia, M. Nakhla, F. Canavero, and R. Achar, "Stability, causality, and passivity in electrical interconnect models", *Advanced Packaging, IEEE Transactions on*, vol. 30, no. 4, pp. 795–808, 2007 (cit. on pp. 28, 33, 66).

[116] F. C. Van Loan, "The ubiquitous kronecker product", *Journal of Computational and Applied Mathematics*, vol. 123, pp. 85–100, 2000 (cit. on p. 29).

[117] B. Gustavsen, *Vector fitting code*, Matlab source code (cit. on pp. 30, 51).

[118] M. R. Wohlers, *Lumped and Distributed Passive Networks.* Academic Press, 1969 (cit. on p. 33).

[119] D. B. O. Anderson and V. S., *Network Analysis and Synthesis*, E. Cliff, Ed. Prentice Hall, 1973 (cit. on pp. 33, 103, 116, 118, 119, 129, 131, 132, 134, 135, 139, 142, 143).

[120] S. Boyd and L. O. Chua, "On the passivity criterion for lti n-ports", *International Journal of Circuit Theory and Applications*, vol. 10, no. 4, pp. 323–333, 1982 (cit. on p. 33).

[121] S. Grivet-Talocia, "Passivity enforcement via perturbation of hamiltonian matrices", *Circuits and Systems I: Regular Papers, IEEE Transactions on*, vol. 51, no. 9, pp. 1755–1769, 2004 (cit. on pp. 33, 39).

[122] S. Boyd, *Convex Optimization*. Cambridge University Press, 2004 (cit. on p. 36).

[123] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan, *Linear matrix inequalities in system and control theory*. SIAM, 1994 (cit. on p. 36).

[124] J. F. Sturm, *Using sedumi 1.02, a matlab toolbox for optimization over symmetric cones*, 1998 (cit. on pp. 37, 119, 132).

[125] J. H. Wilkinson, *The algebraic eigenvalue problem*. Oxford University Press, 1965 (cit. on pp. 40, 51).

[126] T. Kailath, *Linear systems*. Prentice-Hall, 1980 (cit. on pp. 41, 85, 118).

[127] B. Raines and R. Rojas, "Wideband characteristic mode tracking", *Antennas and Propagation, IEEE Transactions on*, vol. 60, no. 7, pp. 3537–3541, 2012 (cit. on p. 52).

[128] P. Triverio, S. Grivet-Talocia, and M. Nakhla, "A parameterized macromodeling strategy with uniform stability test", *Advanced Packaging, IEEE Transactions on*, vol. 32, no. 1, pp. 205–215, 2009 (cit. on pp. 66, 79, 80).

[129] E. Samuel, L. Knockaert, F. Ferranti, and T. Dhaene, "Guaranteed passive parameterized macromodeling by using sylvester state-space realizations", *Microwave Theory and Techniques, IEEE Transactions on*, vol. 61, no. 4, pp. 1444–1454, 2013 (cit. on pp. 66, 79, 80, 85).

[130] F. Ferranti, T. Dhaene, and L. Knockaert, "Compact and passive parametric macromodeling using reference macromodels and positive interpolation operators", *Components, Packaging and Manufacturing Technology, IEEE Transactions on*, vol. 2, no. 12, pp. 2080–2088, 2012 (cit. on pp. 66, 79, 80, 85).

[131] F. Ferranti, L. Knockaert, T. Dhaene, and G. Antonini, "Parametric macromodeling based on amplitude and frequency scaled systems with guaranteed passivity", *International Journal of Numerical Modelling: Electronic Networks, Devices and Fields*, vol. 25, no. 2, pp. 139–151, 2012 (cit. on pp. 66, 79, 80, 85).

[132] F. Ferranti, L. Knockaert, and T. Dhaene, "Passivity-preserving parametric macromodeling by means of scaled and shifted state-space systems", *Microwave Theory and Techniques, IEEE Transactions on*, vol. 59, no. 10, pp. 2394–2403, 2011 (cit. on pp. 66, 79, 80, 85).

[133] ——, "Guaranteed passive parameterized admittance-based macromodeling", *Advanced Packaging, IEEE Transactions on*, vol. 33, no. 3, pp. 623–629, 2010 (cit. on pp. 66, 79, 80, 85).

[134] F. Ferranti, L. Knockaert, T. Dhaene, and G. Antonini, "Passivity-preserving parametric macromodeling for highly dynamic tabulated data based on lur'e equations", *Microwave Theory and Techniques, IEEE Transactions on*, vol. 58, no. 12, pp. 3688–3696, 2010 (cit. on pp. 66, 79, 80, 85).

[135] L. O. Chua, C. A. Desoer, and E. S. Kuh, *Linear And Non Linear Circuits*. McGRaw-Hill, 1987 (cit. on pp. 67, 77).

[136] S. Grivet-Talocia, P. Brenner, and F. Canavero, "Fast macromodel-based signal integrity assessment for rf and mixed-signal modules", in *Electromagnetic Compatibility, 2007. EMC 2007. IEEE International Symposium on*, 2007, pp. 1–6 (cit. on pp. 68, 86).

[137] R. Behzad, *RF microelectronics*. Prentice Hall, 1998 (cit. on pp. 69, 70).

[138] B. Gustavsen and A. Semlyen, "A robust approach for system identification in the frequency domain", *Power Delivery, IEEE Transactions on*, vol. 19, no. 3, pp. 1167–1173, 2004 (cit. on p. 79).

[139] D. Deschrijver, B. Haegeman, and T. Dhaene, "Orthonormal vector fitting: a robust macromodeling tool for rational approximation of frequency domain responses", *Advanced Packaging, IEEE Transactions on*, vol. 30, no. 2, pp. 216–225, 2007 (cit. on p. 79).

[140] S. Grivet-Talocia, S. Acquadro, M. Bandinu, F. Canavero, I. Kelander, and M. Rouvala, "A parameterization scheme for lossy transmission line macromodels with application to high speed interconnects in mobile devices", *Electromagnetic Compatibility, IEEE Transactions on*, vol. 49, no. 1, pp. 18–24, 2007 (cit. on pp. 79, 80).

[141] K. Singhal and J. Vlach, "Symbolic analysis of analog and digital circuits", *Circuits and Systems, IEEE Transactions on*, vol. 24, no. 11, pp. 598–609, 1977 (cit. on p. 80).

[142] K. Zhou, J. Doyle, and K. Glover, *Robust and optimal control*. Prentice-Hall, 1996 (cit. on p. 85).

[143] S. Xu and J. Lam, *Robust Control and Filtering of Singular Systems*. Springer Berlin / Heidelberg, 2006 (cit. on p. 85).

[144] W. Cauer, *Synthesis of Linear Communication Networks*. McGraw-Hill Companies, 1958 (cit. on p. 102).

[145] R. Pauli, "The scientific work of wilhelm cauer and its key position at the transition from electrical telegraph techniques to linear systems theory", in *Cybernetics and systems 16th, European meeting on cybernetics and systems research*, 2002 (cit. on p. 102).

[146] R. Foster, "A reactance theorem", Bell System Journal, Tech. Rep., 1924 (cit. on p. 102).

[147] S. Darlington, "Synthesis of reactance 4-poles which produce prescribed insertion loss characteristics, including special applications to filter design", PhD thesis, Columbia university, 1940 (cit. on pp. 102, 141).

[148] W. Cauer, "Die verwirklichung von wechselstromwinderstanden vorgeschriebener frequenzabhaungigkeit", *Arch. Elektrotech.*, vol. 17, pp. 355–388, 1962 (cit. on p. 102).

[149] O. Brune, "Synthesis of a finite two-terminal network whose driving-point impedance is a prescribed function of frequency", PhD thesis, Massachusetts Institute of Technology (MIT), 1931 (cit. on p. 102).

[150] B. Tellegen, "Synthesis of passive resistanceless four-poles that may violate reciprocity condition", Philips, Tech. Rep. 3, 1948 (cit. on p. 102).

[151] R. Boot and R. J. Duffin, "Impedance synthesis without use of transformers", *Journal of Applied Physics*, vol. 20, p. 816, 1949 (cit. on pp. 102, 103).

[152] B. Tellegen, "The gyrator, a new electronic network element", Philips, Tech. Rep., 1948 (cit. on p. 102).

[153] B. D. O. Anderson and R. Newcomb, "Impedance synthesis via state-space techniques", *Electrical Engineers, Proceedings of the Institution of*, vol. 115, no. 7, pp. 928–936, 1968 (cit. on p. 102).

[154] D. Youla and P. Tissi, "N-port synthesis via reactance extraction - part i", in *IEEE International Convention*, 1966 (cit. on pp. 102, 103, 117, 128).

[155] R. E. Kalman, "On a new characterization of linear passive systems", in *First Allerton Converence on Circuit Theory*, 1963 (cit. on p. 103).

[156] B. McMillan, "Passive multiterminal networks without transformers", *Circuits and Systems I: Regular Papers, IEEE Transactions on*, vol. 54, no. 10, pp. 2315–2320, 2007 (cit. on pp. 103, 110, 112).

[157] E. Guillemin, R. Foster, L. Weinberg, I. Cederbaum, and G. Biorci, "The realization of n-port networks without transformers - a panel discussion: statements by discussants", *Circuit Theory, IRE Transactions on*, vol. 9, no. 3, pp. 202–208, 1962 (cit. on p. 103).

[158] A. Einstein, "Investigations on the theory of, the brownian movement", PhD thesis, University of Zurich, 1905 (cit. on p. 104).

[159] J. B. Johnson, "Thermal agitation of electricity in conductors", *Phys. Rev.*, vol. 32, pp. 97–109, 1 1928 (cit. on p. 104).

[160] H. Nyquist, "Thermal agitation of electric charge in conductors", *Phys. Rev.*, vol. 32, pp. 110–113, 1 1928 (cit. on p. 104).

[161] R. Saeks, *Generalized Networks*. Holt, Rinehart, and Winston, Inc., 1972 (cit. on pp. 107, 118).

[162] J. C. Willems, "Realization of systems with internal passivity and symmetry constraints", J. Franklin, Tech. Rep. 301, 1976 (cit. on pp. 107, 138).

[163] U. Bakshi and A. Bakshi, *Circuit Theory*. Technical Publications Pune, 2009 (cit. on p. 109).

[164] J. D. Irwin, *Basic Engineering Circuit Analysis*. Macmillan, 1987 (cit. on p. 109).

[165] I. Cederbaum, "Conditions for the impedance and admittance matrices of n-ports without ideal transformers", *Proceedings of the IEE - Part C: Monographs*, vol. 105, no. 7, pp. 245–251, 1958 (cit. on p. 110).

[166] F. Reza, "Synthesis without ideal transformers", *Journal of Applied Physics*, vol. 25, no. 6, pp. 807–808, 1954 (cit. on p. 110).

[167] O. P. D. Cutteridge, "Synthesis of linear n-terminal networks", *Proceedings of the IEE - Part C: Monographs*, vol. 109, no. 16, pp. 640–645, 1962 (cit. on p. 110).

[168] R. A. Horn and C. R. Johanson, *Matrix Analysis*. Cambridge University Press, 1990 (cit. on p. 113).

[169] M. Celick, L. Pileggi, and A. Odabasioglu, *IC Interconnect Analysis*. Kluwer Academic, 2002 (cit. on pp. 116, 120).

[170] E. G. Gilbert, "Controllability and observability of linear systems", *SIAM Journal of Control*, vol. 1, pp. 128–151, 1963 (cit. on pp. 118, 166).

[171] T. Reis and J. Willems, "A balancing approach to the realization of systems with internal passivity and reciprocity", *Systems & Control Letters*, vol. 60, no. 1, pp. 69 –74, 2011 (cit. on pp. 119, 131–133).

[172] G. Vasilescu, *Electronic Noise and Interfering Signals: Principles and Applications*. Springer, 1999 (cit. on p. 123).

[173] P. Lancaster and L. Rodman, *Algebraic Riccati Equations*. Clarendon Press, 1995 (cit. on pp. 130, 144).

[174] D. A. Bini, B. Iannazzo, and B. Meini, *Numerical Solution of Algebraic Riccati Equations*. SIAM, 2012 (cit. on pp. 130, 144).

[175] A. Laub, "A schur method for solving algebraic riccati equations", *Automatic Control, IEEE Transactions on*, vol. 24, no. 6, pp. 913–921, 1979 (cit. on p. 130).

[176] F. Harris, "On the use of windows for harmonic analysis with the discrete fourier transform", *Proceedings of the IEEE*, vol. 66, no. 1, pp. 51–83, 1978 (cit. on pp. 147, 155).