1

INTRODUCTION

This introductory chapter provides a quick preview of the book contents, with the main purpose of defining what a macromodel is, motivating why macromodels are important, and overviewing the most important requirements that macromodels should meet in order to be of practical use. There are no derivations in this chapter, and any results that are stated here are not proved but only illustrated through simple examples. Details will be discussed in the following chapters of this book.

1.1 WHY MACROMODELING?

Many real-world problems are too complex to be modeled in full detail starting from first principles. In fact, the processing time and the memory requirements for a *direct* simulation of a fully detailed system, such as a chip-package-board electronic structure or a high-voltage power system, are prohibitive on any computer. For this reason, common engineering flows are based on divide and conquer approaches. Different devices and subblocks, which comprise the system, are characterized and modeled independently, with a level of accuracy that meets the application requirements. These individual models are then suitably interconnected for system-level analyses, usually in the form of time-domain simulations, allowing a full system study with an acceptable computational effort. The extraction of the models, that is, the *macromodeling* task, is of course an essential step in the overall procedure.

We understand by the term *macromodel* a reduced-complexity behavioral description of a device or a collection of devices. Macromodels are inherently approximate, since

Passive Macromodeling: Theory and Applications, First Edition. Stefano Grivet-Talocia and Bjørn Gustavsen.

their construction deliberately neglects some aspects that are deemed unimportant for the system behavior. Nonetheless, macromodels must be accurate enough to allow appropriate decisions by designers based on the results of subsequent numerical simulations. Depending on the application, macromodels may have to fulfill additional properties. Among these, the most common is passivity, which arises as a fundamental constraint when representing structures that are unable to generate energy, such as electrical interconnects; hence, the title of this book, *"Passive Macromodeling"*.

Many different approaches to macromodeling are available. Two popular classes of techniques, sometimes denoted as *white-box* and *gray-box*, assume a model structure that reproduces or mimics the physical topology of the system that the macromodel intends to represent. For instance, a set of physical conductors may be represented by a network of resistances (representing ohmic losses in each conductor), capacitances (representing charge accumulation at conductors in close proximity), and possibly coupled inductances (representing magnetic coupling between conductors). Either an automated extraction software or an expert designer is required for the construction of such models. This task may, however, become very difficult when the topology of the underlying physical structure becomes very complex. Even worse, the internal structure of the device to be modeled may be only partially known to the engineer who is in charge of building the macromodel. In such cases, exploiting an incomplete knowledge of the system in the model development is very difficult and likely to fail.

This book develops a complementary *black-box* approach. We seek for models that reproduce the behavior of a physical structure with respect to its input and output characteristics, observed from well-defined external terminals. No information on the internal structure of the actual system is exploited in the construction of the macromodel. This implies that there is usually no direct link between the internal topological and dynamical structure of the model and that of the physical system. A typical application scenario involves the availability of a limited number of time- or frequency-domain responses, obtained by a direct measurement or through a commercial field solver. Macromodels are constructed by fitting the parameters of a suitable model class to this data. The following example illustrates this point on a simple two-port circuit block.

Let us consider the two-port circuit element depicted in Figure 1.1(a), assumed to be the reference (true) system. We evaluate the corresponding 2×2 admittance matrix

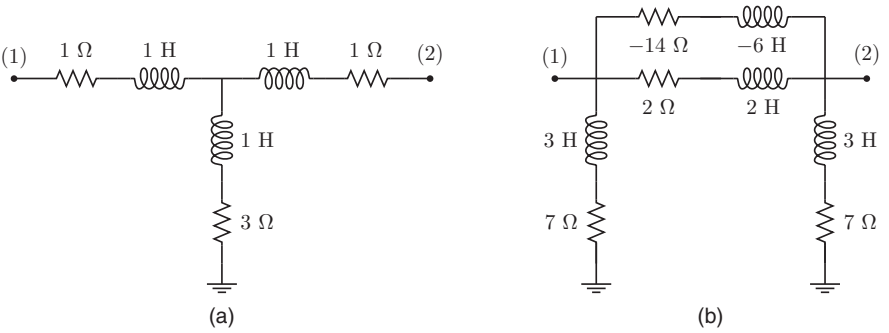

Figure 1.1 Original two-port circuit (a) and its synthesized model (b).

$\mathbf{Y}(s)$ defined at ports (1) and (2). The elements of this matrix are rational functions of the complex frequency s, which can be written as

$$Y_{11}(s) = Y_{22}(s) = \frac{2s+4}{3s^2+10s+7}, \quad Y_{12}(s) = Y_{21}(s) = \frac{-s-3}{3s^2+10s+7}\,. \qquad (1.1)$$

Starting from (1.1) and applying one of the (black-box) circuit synthesis methods, which will be discussed later in Chapter 11, leads to the equivalent circuit depicted in Figure 1.1(b). Since no information on the original topology was used in the synthesis, this equivalent circuit looks very different from the original. Some circuit elements are even negative. Yet, the behavior of the two circuit realizations as observed from ports (1) and (2) is identical, as a frequency sweep of the admittance matrix entries depicted in Figure 1.2 confirms.

One may ask the question whether one circuit realization is preferable to the other. As a general guideline, whenever some information is available on the original system, this should be exploited in the construction of the model. Referring to the aforementioned example, if we know that the original system is comprised of three inductors connected in a "wye" configuration, then it is clear that the circuit topology in Figure 1.1(a) is preferable. Even if the value of the circuit elements is not known, it can be easily

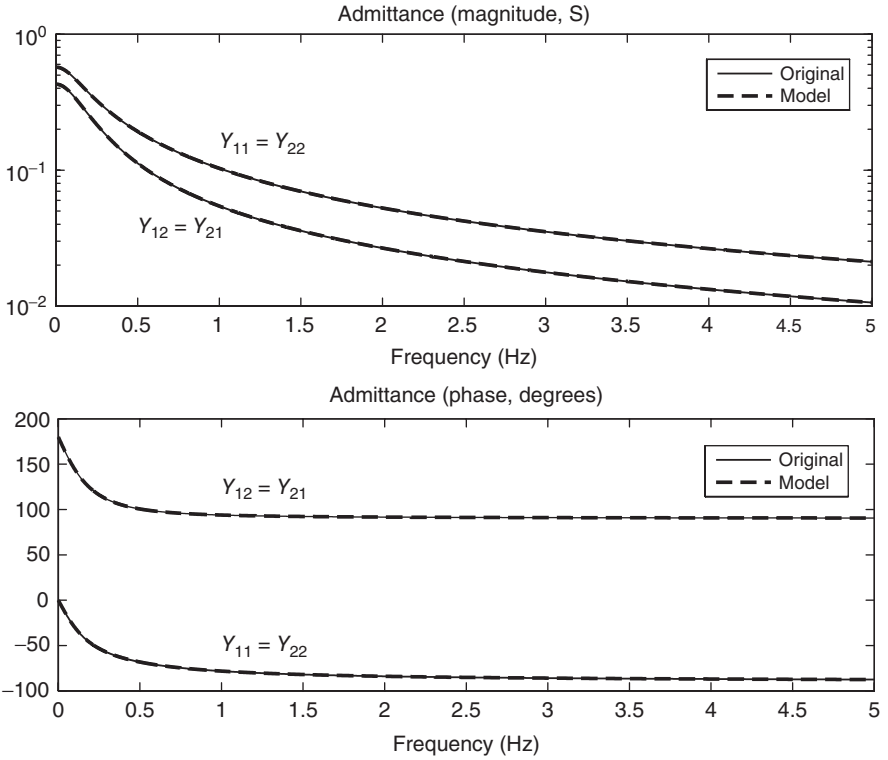

Figure 1.2 Admittance responses of the two circuit realizations shown in Figure 1.1.

estimated from measured port responses through a fitting process. However, when the number of connected elements in the original system is very large, as in the case of parasitic networks, with possibly many internal nodes that are not available externally, guessing the original topology and estimating the component values from external responses is not feasible. In this case, one should abandon the hope for a topology-aware model and resort to a pure black-box approach. If relevant for the application at hand, one may further seek for circuit realizations of the macromodel that do not include negative elements. This is indeed possible, as discussed later in Chapter 11.

1.2 SCOPE

In this book, we restrict ourselves to the treatment of systems and devices that are *linear* in their input–output behavior. This may at first sound restrictive, as nonlinear devices are found in many real-world systems. However, for most electronic and high-voltage power applications, most of the overall complexity is due to electromagnetic coupling or interaction between various system parts. Sometimes this interaction is unwanted and labeled as *parasitic*, while sometimes it is required and designed for proper system functioning. Such effects are inherently linear in the vast majority of applications. Therefore, linear macromodeling can be viewed as a process that replaces a high-complexity network of parasitics, or more generally a complex electromagnetic structure, with a lower complexity model at a well-defined set of interface ports. Additional devices with a nonlinear behavior can be modeled separately and included later in the same time-domain simulation, by suitably interconnecting all individual blocks at their interface ports. This modular approach is straightforward with computer simulation programs such as Simulation Program with Integrated Circuit Emphasis (SPICE) or Electromagnetic Transients Program (EMTP). Modeling of nonlinear devices requires techniques that are fundamentally different from those of linear theory, and as such it falls outside the scope of this book.

We further restrict ourselves to univariate modeling, by assuming only frequency or time as the free (independent) variable, although the topic of multivariate macromodeling is emerging as a more and more important field of research and application. Multivariate macromodeling involves introducing in the model expressions one or more additional variables, such as material or device design parameters. Preservation of this dependence in behavioral macromodels is extremely valuable for designers, who need to determine the best parameter configuration that allows suitable system performance metrics to be met, possibly after an optimization loop. Multivariate macromodeling is still a somehow immature field, with only partial results and sometimes inefficient or inaccurate algorithms available. For this reason, we have decided to not discuss multivariate macromodeling except for one introductory section in Chapter 14.

There are various other reasons why macromodeling approaches are important or prove useful in applications, some of which are listed as follows.

Inclusion of frequency-dependent effects. Many devices are characterized by strong and possibly complicated frequency-dependent effects. The series impedance of a transmission line is a simple example, where the frequency

dependence induced by skin and proximity effects results in a complex transcendental function of frequency. With the exception of a few idealized cases, it may be very difficult, if not impossible, to capture these effects in a time-domain simulation without resorting to some kind of macromodeling or approximation procedure.

Modeling from measurements. Sometimes a device is known only through time-domain or frequency-domain measurements of its input–output responses. This situation is quite common when the device is not developed in-house but acquired from a vendor, in which case no knowledge of the internal structure, geometry, or materials is available. The *black-box* macromodeling techniques that we discuss in this book are ideally suited for this scenario, allowing extraction of efficient simulation models from measured responses (e.g., impedance or voltage–current waveforms). Based on such models, system-level simulation and optimization becomes simple and efficient.

Hiding proprietary information. Macromodeling represents a given device with a closed-form (rational) representation of its transfer function, or with an equivalent state-space model. The parameters in these representations (pole–residue pairs or state-space matrices) are derived through a mathematical procedure and are therefore not related to the topology of the underlying system. Only the external port behavior is retained or approximated, not the internal structure. This approach automatically permits to hide proprietary information, so that the resulting macromodel can be freely shared without disclosing sensitive details. This is true even when the macromodel is synthesized as an equivalent circuit netlist, since the various elements that are produced are obtained through a mathematical synthesis process and are again not related to the internal structure of the device being modeled.

Fast behavioral characterization. Macromodels can be used for interpolation. This feature turns out to be quite useful when the characterization of a complex structure, for example, the evaluation of the response at one frequency using a field solver, is computationally expensive. When a macromodel is available, possibly derived from a small set of frequency samples, the determination of the response at many other frequency points can be achieved simply by evaluating the closed-form expression of the macromodel. Further, if the macromodel construction is embedded in the characterization loop and suitably interfaced with the field solver, it can be used as an adaptive sampling engine that automatically selects a minimal subset of frequency points that are strictly necessary for the model extraction. Several examples will be presented in Chapter 13.

Extrapolation. The aforementioned interpolation properties can also be extended to the extrapolation of a transfer function to the entire complex plane. The response of a closed-form macromodel is a function of the complex frequency (Laplace) variable s, so that its evaluation can be performed anywhere in the complex plane. This enables a fast computation of transient responses using numerical inverse Laplace transform (NILT), which has some inherent advantages with respect to the standard inverse Fourier transform and for which efficient algorithms exist.

Fast time-domain simulation. Macromodels are essentially closed-form expressions that match the device responses with a prescribed accuracy. When macromodels are based on rational approximations in the complex frequency domain, the conversion between frequency and time domains is straightforward, due to the analytic inversion properties of the Laplace transform. This in turn enables including the macromodels in transient simulators based on time stepping, as recursive convolution blocks. The latter are particularly efficient when compared to other alternative transient simulation approaches.

Subnetwork equivalencing. Macromodeling enables divide and conquer approaches for handling large circuits, networks, and systems. Even if the direct simulation of a large-scale system is feasible, a much faster simulation is sometimes possible by identifying one or more subsystems and replacing them with suitable reduced-complexity macromodels matching their input–output responses. The automation of this procedure is a key enabling factor for improving scalability and capacity of modern circuit solvers.

1.3 MACROMODELING FLOWS

Black-box macromodels can be constructed following many different approaches, depending on what kind of information is available on the original system. These approaches lead to different "flows", intended as well-defined sequences of steps leading to the final deliverable model. We introduce the most common flows next, leaving a more detailed motivation and discussion to Chapter 4.

1.3.1 Macromodeling via Model Order Reduction

Let us consider Figure 1.3. In this scenario, the starting point is a detailed knowledge of geometry and material properties of the structure under modeling. This information is typically available in computer-aided design (CAD) data files elaborated by designers.

The first step in this flow involves the electrical or electromagnetic description of the system behavior, so that all functional and parasitic effects of interest are properly accounted for. One has to properly define what are the interface ports for the system and decide what are the variables that will play the role of inputs (excitations) and outputs (responses). Typically, the dual variables at each port (voltage/current or incident/reflected scattering waves) are split, with one acting as input and the other acting as output. In this phase, appropriate boundary conditions are also established. Then, the geometry is suitably meshed, and Maxwell's equations are formulated in integral or differential form and discretized on this mesh. In particular, only the space dependence is discretized, whereas the time (or frequency) dependence is left continuous. The result of this process may be available either as a possibly large system of ordinary differential equations (ODEs) or differential algebraic equations (DAEs), or as a large-scale equivalent circuit. These alternative forms provide the input to the actual macromodeling stage. Note that whenever the system behavior is formulated as a large-scale circuit, the standard modified nodal analysis (MNA) method can be applied to construct an equivalent DAE or ODE system.

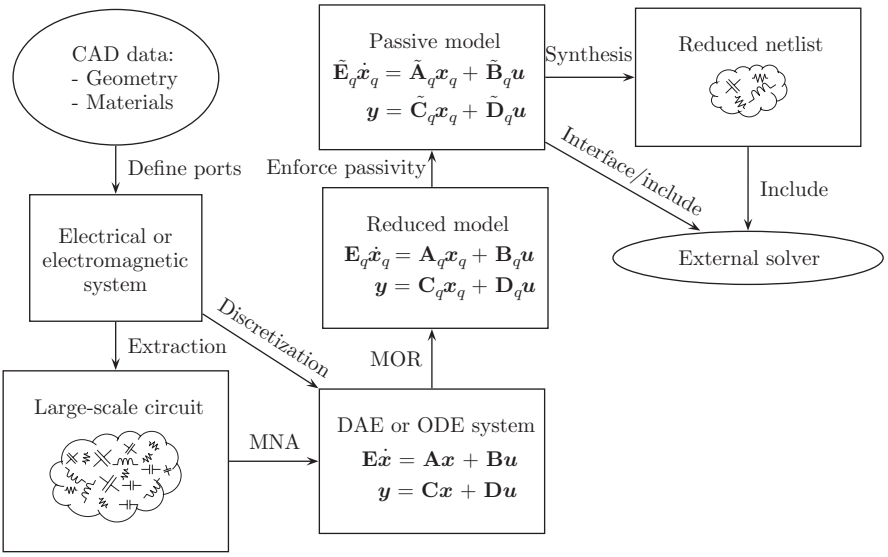

Figure 1.3 Macromodeling flow based on model order reduction.

Macromodeling via model order reduction (MOR) starts from the aforementioned *internal* description of the underlying system, and attempts to compress the existing large-scale model or circuit by reducing its size and complexity, while preserving its input–output characteristics. Many different techniques are available. Pure topological reduction methods process directly the equivalent circuit using graph theoretical tools, in order to exploit (approximate) equivalences and reduce the number of components. These methods will not be discussed in this book. When starting instead from an ODE or DAE description, several approximation methods can be used to project the state-space system onto a lower dimensional subspace while preserving input–output accuracy. Moment matching and truncated balancing methods belong to this class. In some cases, these approaches are able to preserve passivity, such as the well-known algorithm PRIMA. In some other cases, an optional passivity check and enforcement step may be applied. As a last step, the model is included in an external solver for later simulations.

MOR approaches are not the main subject of this book. However, due to their importance in many application areas, we dedicate Chapter 5 to an introduction to the basic and most commonly used algorithms, also providing a number of key references.

1.3.2 Macromodeling from Field Solver Data

This second macromodeling flow is probably the most common in several application areas. We refer to Figure 1.4, from which we see that the starting point is the same as for the MOR-based approaches, that is, a detailed knowledge of geometry and material properties of the true system. What makes the difference here is the type of field solver that is used to discretize the structure and extract its electrical or electromagnetic responses. For the MOR case, the solver must be "open," so that we

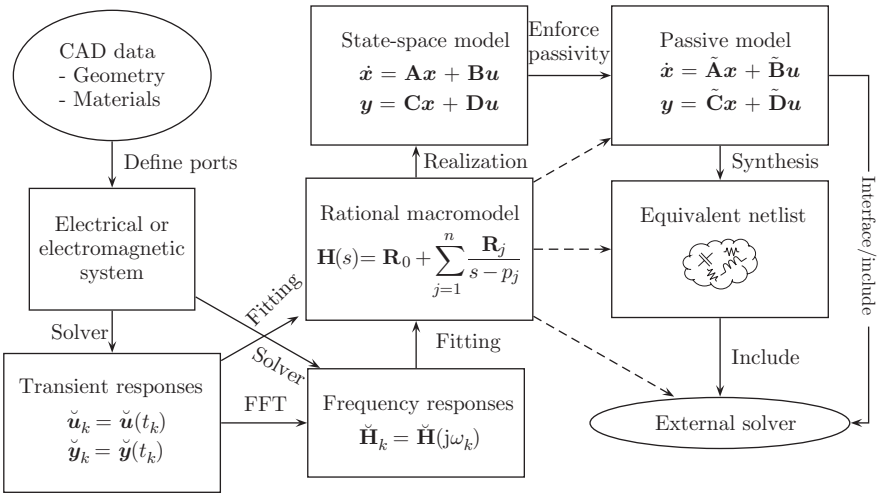

Figure 1.4 Macromodeling flow based on computed responses by full-wave solvers.

can export the internally discretized ODE or DAE systems. This situation is common for solvers that are developed in-house either at academic and research institutions or in large companies. However, most general-purpose commercial solvers are deliberately "closed," since exporting intermediate processing and discretization results might disclose sensitive details on the embedded proprietary algorithms. Such solvers only allow exporting the results of the computations, either in the form of transient responses excited by user-defined stimuli (e.g., in the case of solvers based on the finite difference time-domain method or similar algorithms) or in the form of sampled frequency responses (e.g., in the case of solvers based on finite element or method of moments). In other words, once imported and processed by the solver, the system becomes a black box, characterized only through input–output responses.

A black-box macromodeling algorithm is required in this flow. We can distinguish between frequency-domain techniques, which operate mostly by applying rational curve fitting to the transfer matrix samples, and time-domain methods, which construct the macromodel by directly processing transient excitations and responses as obtained by time-domain solvers. Alternatively, transient solver outputs may be processed by standard numerical Fourier transform techniques to derive estimates of the frequency responses, which can then be processed by frequency-domain macromodeling schemes. In all cases, the macromodeling process may produce a nonpassive model. Therefore, as discussed later in this chapter, a passivity check and enforcement step is usually performed after a state-space conversion, but before the synthesis of the macromodel in a form that is compatible with later system-level simulations. Alternatively, passivity enforcement and synthesis can be performed directly from a pole–residue representation, without going through a state-space realization.

1.3.3 Macromodeling from Measured Responses

The last macromodeling flow that we discuss is truly black box. With reference to Figure 1.5, we assume that the device under modeling is not available in the form of

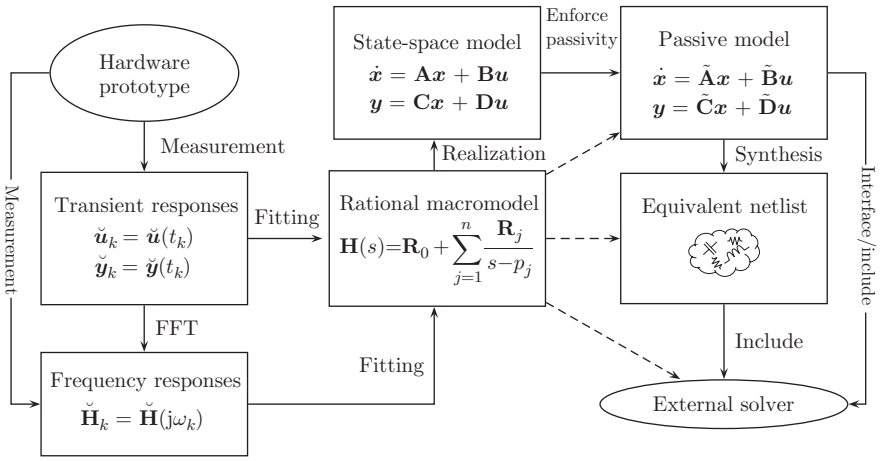

Figure 1.5 Macromodeling flow based on measured responses.

a CAD model, but as real hardware, either as a prototype or as final product. In this situation, one can characterize the structure through direct measurement. The most common approach is to perform the measurement in the frequency domain through a vector network analyzer (VNA), obtaining a set of frequency samples of the scattering matrix of the device. An alternative is to perform measurements in the time domain, for example, using time-domain reflectometry (TDR), by launching an excitation pulse as input and recording the resulting response as output. Such transient data can be processed by a time-domain macromodeling algorithm.

We see that this flow is very similar to the flow depicted in Figure 1.4. The only difference is the replacement of a "virtual" measurement performed by a numerical field solver with a *real* measurement performed on hardware. These two flows form the main motivation for this book, which presents various algorithms for processing input–output responses of a black-box system, in order to obtain an accurate, passive, and efficient behavioral macromodel. The following sections offer a preview of the main macromodeling steps.

1.4 RATIONAL MACROMODELING

The construction of behavioral macromodels follows a two-step procedure. First, the model class is designed in order to be representative of the systems under investigation. Second, a numerical procedure is applied for the estimation of the model parameters, so that the model responses match those of the target system. For the specific case of Linear Time-Invariant (LTI) systems, we can devise the model structure directly in the frequency (Laplace) domain, by choosing a particular functional form for the model transfer function $H(s)$. The most common choice is to let $H(s)$ be a rational function of the complex frequency s, which is equivalent to the assumption that the model dynamic behavior is expressed in the time domain as a set of ordinary differential equations (ODEs), with time t being the independent variable. This equivalence, as well as a description of the various forms in which the ODEs can be formulated, will be discussed later in Chapter 3.

Most readers will be familiar with rational functions $H(s)$ described as the ratio of two polynomials

$$H(s) = \frac{N(s)}{D(s)} = \frac{b_0 + b_1 s + b_2 s^2 + \cdots + b_m s^m}{1 + a_1 s + a_2 s^2 + \cdots + a_n s^n} \tag{1.2}$$

with prescribed numerator and denominator degrees m and n, respectively. The rational function (1.2) can be recast into a pole–zero form

$$H(s) = \alpha \frac{(s - z_1)(s - z_2) \cdots (s - z_m)}{(s - p_1)(s - p_2) \cdots (s - p_n)},$$

where coefficients p_j and z_j are the poles and the zeros, respectively. A third equivalent representation is the pole–residue form

$$H(s) = r_0 + \frac{r_1}{s - p_1} + \frac{r_2}{s - p_2} + \cdots + \frac{r_n}{s - p_n}, \tag{1.3}$$

where the coefficients r_j are the residues and where we assumed $m = n$.

We will see that, when the system under investigation is electrically large, so that the effects of finite propagation speed are visible in its terminal responses, it may be appropriate to modify the aforementioned rational model form by embedding explicit delay terms. The latter are very simple in the Laplace domain, since a delay τ is represented by the exponential factor $e^{-s\tau}$. Chapter 12 will discuss various forms in which such delay operators can be combined with (1.3) when modeling distributed structures. Restricting for the moment our attention to the delayless case, it turns out that the model form that is exploited in the most successful macromodeling algorithms is the partial fraction form (1.3). This is mainly due to the superior robustness of the numerical algorithms that compute the model parameters, in this case, poles and residues.

As stated earlier, the most common macromodeling scenarios require the determination of the model parameters in the frequency domain, by matching the model response to a set of frequency samples obtained by measurement or numerical simulation of the system under investigation. Denoting as $\breve{H}_k$ the available frequency response data at frequency $s_k = \mathrm{j}\omega_k$, we can cast the macromodeling problem as a simple rational curve fitting, which finds the coefficients r_j and p_j such that

$$\breve{H}_k \approx H(\mathrm{j}\omega_k) = r_0 + \frac{r_1}{\mathrm{j}\omega_k - p_1} + \frac{r_2}{\mathrm{j}\omega_k - p_2} + \cdots + \frac{r_n}{\mathrm{j}\omega_k - p_n} \tag{1.4}$$

holds for all available frequency points $k = 1, \cdots, K$. Alternative formulations allow to obtain the model parameters from sampled time-domain responses, as will be discussed in Chapters 6 and 7.

A direct curve fitting based on a least squares (LS) formulation of (1.4) leads to a nonlinear and nonconvex optimization, whose solution becomes very challenging when the model order n exceeds few units. A number of alternative methods will be described in this book for solving (1.4). We will see that the most effective approaches will be variants of the so-called vector fitting (VF) method. The VF algorithm is amazingly

robust and accurate, yet very easy to implement. We give here a short preview, deferring a detailed derivation and discussion to Chapters 7 and 8.

The main VF idea builds on the simple consideration that any rational function can be written as a ratio of rational functions

$$H(s) = \frac{c_0 + \sum_{j=1}^{n} \frac{c_j}{s-q_j}}{1 + \sum_{j=1}^{n} \frac{d_j}{s-q_j}}, \tag{1.5}$$

from which we see that the poles q_j, which are common to both numerator and denominator, cancel out completely. Additionally, we see from (1.5) that the poles p_j of $H(s)$ in (1.3) coincide with the zeros w_j of the denominator

$$\xi(s) = 1 + \sum_{j=1}^{n} \frac{d_j}{s-q_j} = \frac{\prod_{j=1}^{n}(s-w_j)}{\prod_{j=1}^{n}(s-q_j)}. \tag{1.6}$$

In (1.5) and (1.6), the common poles q_j are assumed to be known constants, while the associated numerator and denominator residues c_j and d_j are unknown. Applying now the fitting condition $\breve{H}_k \approx H(\mathrm{j}\omega_k)$ to (1.5) for $s = s_k = \mathrm{j}\omega_k$ and multiplying by $\xi(\mathrm{j}\omega_k)$ both sides of the resulting expression leads to

$$\left(1 + \sum_{j=1}^{n} \frac{d_j}{\mathrm{j}\omega_k - q_j}\right) \breve{H}_k \approx c_0 + \sum_{j=1}^{n} \frac{c_j}{\mathrm{j}\omega_k - q_j}, \quad k = 1, \cdots, K. \tag{1.7}$$

Since (1.7) is a linear combination of the unknowns c_j, d_j for each frequency ω_k, this is recognized as a linear least squares problem, whose solution is straightforward. As soon as coefficients d_j are computed, we use (1.6) to obtain the zeros w_j, which provide an estimate of the model poles p_j. The latter are finally used to solve a second linear LS problem expressed by (1.4), in order to obtain the model residues r_j. For better results, it is usually appropriate to iterate the procedure by restarting the scheme with a new set of "initial" poles defined as $q_j^{\nu} = w_j^{\nu-1}$, where ν denotes the iteration index. This process is commonly denoted as "pole relocation."

The VF scheme was introduced in this form in the late 1990s. Although rational modeling via complex curve fitting was already known and applied since many decades, earlier formulations were not robust enough for routine application to complex and possibly high-order systems. The superior performance of VF thus led to a fast diffusion in both academia and industry, making this scheme the method of choice for rational macromodeling.

1.5 PHYSICAL CONSISTENCY REQUIREMENTS

Rational macromodels are almost invariably required to comply with certain constraints associated with the "physical" properties of real-world systems. The most common properties are as follows.

Realness. Any physical system excited by a real-valued input signal responds with a real-valued output signal. This requirement translates into the simple transfer function condition $H^*(s) = H(s^*)$, where $*$ denotes the complex conjugate operator. For rational macromodels, this implies that all poles p_j and residues r_j in (1.3) are either real or come in complex conjugate pairs.

Causality. Any physical system that is originally at rest and that is excited by an input signal that is applied starting at some time t_0 must produce an output that is zero for $t < t_0$. In other words, the effect must follow the cause that produced it. This is the concept of causality. In the frequency domain, causality imposes an integral relation (the so-called *Kramers–Krönig* dispersion relations) between real and imaginary parts of the system transfer function, which form a Hilbert transform pair. In the particular case of rational macromodels, it turns out that these conditions are guaranteed if all model poles have a strictly negative real part.

Reciprocity. The vast majority of multiport electrical or electromagnetic systems arising in practical applications satisfy the reciprocity conditions, which imply certain symmetry properties of the transfer matrix $\mathbf{H}(s)$. For the most common impedance, admittance, or scattering representations, reciprocity implies $\mathbf{H}(s)^{\mathsf{T}} = \mathbf{H}(s)$.

Stability. Several definitions of stability exist that apply in different situations. For instance, the bounded-input bounded-output (BIBO) stability imposes a strong requirement on the system, which must produce a bounded signal as a response to a bounded excitation. For the particular case of rational macromodels, this requirement translates into a constraint on the location of the model poles, which must be confined into the left half of the complex plane (asymptotic stability). Other less stringent definitions are sometimes needed, as will be discussed in Chapter 2. Practically, all VF implementations include an explicit constraint that forces the model poles to be in the left half complex plane, since this condition guarantees at the same time asymptotic stability and causality.

Passivity. We can regard passivity as the most crucial and general property. A passive system is not able to generate energy on its own, under any condition. It is allowed to release power to the external environment, but only if some energy has been previously "stored" in the system. One can visualize this concept through an analogy with a water tank, from which one can extract water with a given flow rate only if it is not empty. We will see that a passive model must have a transfer matrix $\mathbf{H}(s)$ that is either *Positive Real* (in the case of impedance or admittance, in short *immittance* representations) or *Bounded Real* (in case of scattering representations). These conditions are first stated in Chapter 2 and developed in great detail for rational and state-space macromodels in Chapters 9 and 10. The theory that is elaborated in these chapters will show that passivity implies realness, causality, and stability (suitably defined). Restricting our focus on impedance or admittance systems, we will see that the following eigenvalue condition must hold:

$$\lambda_i \geq 0 \quad \forall \lambda_i \in \lambda(\mathbf{H}(\mathrm{j}\omega) + \mathbf{H}(\mathrm{j}\omega)^{\mathsf{H}}) \quad \forall \omega \in \mathbb{R}\,. \tag{1.8}$$

This condition may in fact be hard to check and even harder to enforce when building a macromodel. We can thus safely state that passivity enforcement is the most numerically challenging step in the overall macromodeling procedure, most often requiring a trade-off between optimality (in terms of accuracy) and efficiency.

We further elaborate the concept and the implications of passivity for a particular case of a one-port impedance $Z(s)$. This impedance is passive when

1. it is stable, so that one can define the frequency response $Z(\mathrm{j}\omega)$ as the steady-state voltage solution under a sinusoidal current excitation with unit amplitude and (angular) frequency ω;
2. the real part of $Z(\mathrm{j}\omega)$ is nonnegative at all frequencies,

$$Z(\mathrm{j}\omega) = R(\omega) + \mathrm{j}X(\omega)\,, \quad R(\omega) = \mathrm{Re}\{Z(\mathrm{j}\omega)\} \geq 0 \quad \forall \omega \in \mathbb{R}\,.$$

   This statement, which particularizes the more general condition (1.8) to the scalar one-port case, is certainly familiar to the reader, who expects that a positive resistance is not able to generate energy;
3. its real part is an even function of frequency, $R(\omega) = R(-\omega)$, and its imaginary part is an odd function of frequency, $X(\omega) = -X(-\omega)$. This is an immediate consequence of the realness condition.

Let us now assume that $R(\omega)$ is negative at some reference frequency ω_0, so that $R_0 = R(\omega_0) < 0$ and condition 2 is violated. Figure 1.6 illustrates this situation on a simple test case. The active power absorbed by the impedance in sinusoidal steady-state (AC) conditions at frequency ω_0 reads $P_0 = R_0 I_{\mathrm{rms}}^2$, where I_{rms} denotes the root mean square (RMS) value of the current through the impedance. Clearly, also $P_0 < 0$, so that the impedance releases active power with a constant supply rate to the circuit block $\mathcal{B}$ to which it is connected.

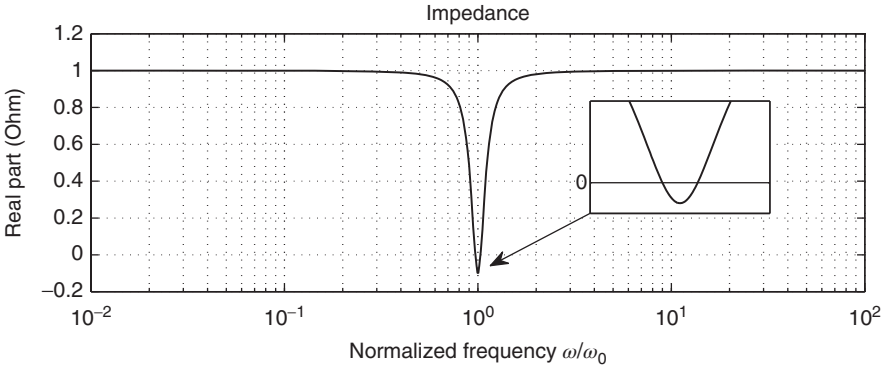

Figure 1.6 Real part of a nonpassive impedance $Z(s)$, with $\mathrm{Re}\{Z(\mathrm{j}\omega_0\} < 0$.

Assume $\mathcal{B}$ as a series connection of an inductor L and a capacitor C, with inductance and capacitance values such that $\omega_0^2 LC = 1$. The resulting circuit, depicted in Figure 1.7, resembles a simple RLC oscillator, with the exception that the resistance is replaced by the impedance $Z(s)$. We now excite the system through a series voltage source $e(t) = \sin(\omega_0 t)$, which is switched on at $t = 0$ and off again after three exact periods, at $t = 3T = 6\pi/\omega_0$. This source is characterized by a dominant spectral component at ω_0, which is injected in the circuit. The reactive components L and C, which are clearly passive devices, resonate at this frequency. Even if the source is switched off so that for $t > 3T$, the circuit is free-running, the impedance $Z(s)$ continues to inject power into the system. As a result, we obtain an unstable transient solution, as Figure 1.8 confirms.

This example illustrates the general fact that a system formed by an interconnection of several subsystems is not guaranteed to remain stable when at least one of its components is not passive. More generally, there exists a simple algorithmic procedure that, given any nonpassive component, synthesizes a destabilizing passive termination [164]. We therefore conclude that nonpassive models (of passive structures) should always be avoided, because system-level simulations based on such models may blow up with exponential rate, even if the terminations are passive. Conversely, any interconnection of individually passive subsystems is always guaranteed to be stable. We therefore see that passivity is of paramount importance under the standpoint of engineers or designers that are responsible for model generation and provision.

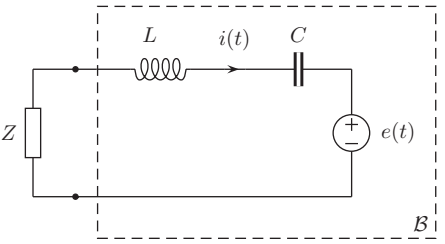

Figure 1.7 Template unstable circuit.

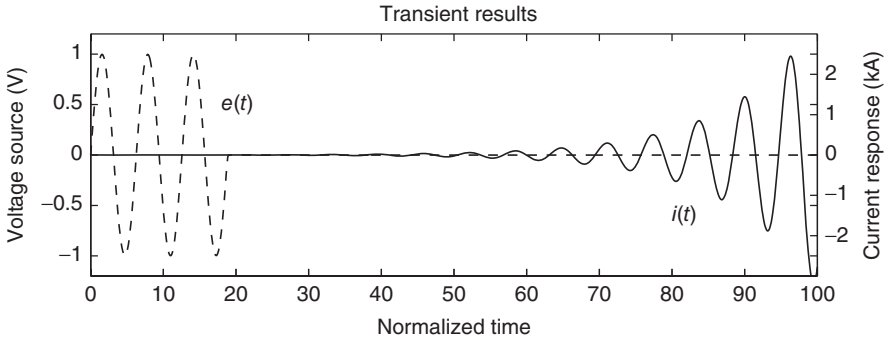

Figure 1.8 Unstable transient response computed from the circuit shown in Figure 1.7.

1.6 TIME-DOMAIN IMPLEMENTATION

One of the reasons why rational macromodels are so widely used in many engineering applications is the straightforward process that is required for their inclusion as subblocks in system-level transient simulations. The two main approaches that are usually adopted are illustrated here using a simple one-port and one-pole example, described by an admittance model

$$Y(s) = \frac{r}{s-p}\,, \tag{1.9}$$

with p and r real. A more complete treatment of the general case is postponed to Chapter 11.

Rational transfer functions characterize the class of lumped LTI circuits. Therefore, one may seek for a process that, starting from a mathematically derived rational macromodel, synthesizes an equivalent circuit whose transfer function matches exactly the macromodel expression. This is in fact a particular case of the general circuit synthesis theory, developed several decades ago and now well established. However, the focus is here not to build a physical circuit realization in hardware, but to produce an equivalent netlist that proves efficient when solved by simulation software such as SPICE or EMTP. Therefore, more flexibility can be exploited in the synthesis.

The extraction of an equivalent circuit that corresponds to (1.9) is straightforward. Considering that the corresponding impedance is

$$Z(s) = Y(s)^{-1} = \frac{s}{r} + \frac{-p}{r}\,,$$

we can represent the macromodel as a series connection of an inductance $L = 1/r$ and a resistance $R = -p/r$. If the model order is greater than 1, so that more partial fraction terms like (1.9) are present, the individual series RL blocks corresponding to each partial fraction are connected in parallel, leading to a Foster-like topology. In this realization, all circuit elements are positive only if the poles are real and negative and if the corresponding residues are positive. This may not always be true, in which case one should adopt a synthesis process that is suited to the particular solver that will be used for transient analysis. If this solver accepts negative elements, no other action is required. Otherwise, more sophisticated approaches are in order. In any case, as discussed in Chapter 11, there are many alternatives for the synthesis of general multiport immittance or scattering macromodels as equivalent netlists, thus allowing a direct macromodel inclusion in standard circuit simulators.

A second approach for transient analysis of interconnected systems that embed one or more macromodels is a direct discretization of the time-domain macromodel response. From (1.9), we can derive the corresponding impulse response through the inverse Laplace transform, obtaining

$$y(t) = \mathscr{L}^{-1}\{Y(s)\} = re^{pt}, \quad t \geq 0\,.$$

The current response due to a voltage excitation $v(t)$ is easily obtained through convolution

$$i(t) = y(t) * v(t) = \int_0^t y(t-\tau)v(\tau)\mathrm{d}\tau = \int_0^t re^{p(t-\tau)}v(\tau)\mathrm{d}\tau\,. \tag{1.10}$$

Upon time discretization, here assumed for simplicity with uniform time stepping $t_k = k\Delta t$, the current response samples $i_k = i(t_k)$ are given by

$$i_k \approx \Delta t \sum_{\ell=0}^{k} \hat{y}_{k-\ell} v_\ell\,, \tag{1.11}$$

where the definition of the discrete impulse response samples $\hat{y}_{k-\ell}$ depends on the particular quadrature rule used to discretize (1.10). This expression provides a discrete-time characteristic equation for the macromodel, which can be easily interfaced with nearly any transient solver.

Although the coefficients in (1.11) are known analytically and can be precomputed, this approach is almost never used in practice. In fact, since the kernel in the convolution integral is an exponential function, a simple trick allows to discretize and cast (1.10) as a recursive convolution

$$i_k \approx \alpha_1 i_{k-1} + \beta_0 v_k + \beta_1 v_{k-1}\,, \tag{1.12}$$

where the coefficients α_1, β_0, and β_1 can be precomputed based on r, p, and Δt, using a prescribed quadrature rule. This recursive form is much more efficient: only one past sample of both input and output signals is required at each time step, instead of the full set of past samples in (1.11). Various approaches, discussed in Chapter 11, are available to extend (1.12) to the general multiport case, to any input–output representation and to construct an appropriate interface with an external solver.

1.7 AN EXAMPLE

We consider here a simple real-world example, which illustrates the various steps of passive macromodeling that will be the subject of this book, with emphasis on the passivity requirements and their implications.

Let us consider the transmission line depicted in Figure 1.9. This structure consists of a simplified model of a low-frequency power transmission system, for which we want to simulate a transient voltage response excited by a constant unit voltage source E_0 that is applied at $t = 0$ at one of the line ends and then disconnected at $t = 0.1$ s. The other line end is left open.

The first step is to characterize the external port behavior of the transmission line. We opt for the admittance representation, which defines the relation between port currents and port voltages in the frequency domain and which is known to be well defined for this structure. The admittance matrix $\breve{\mathbf{Y}}(s)$ is calculated at a set of discrete frequency samples $s_k = \mathrm{j}\omega_k$ by means of analytical formulae, which include suitable terms accounting for skin effect losses in conductors. The admittance data

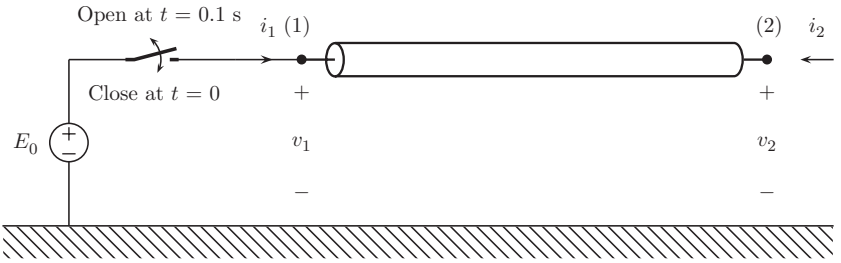

Figure 1.9 A transmission line system: single conductor (diameter 21.66 mm, DC per unit length resistance $R_{\rm DC} = 0.121\ \Omega$/km) of 2 km length, placed in air 18 m above a lossy ground (conductivity $\sigma_c = 0.1$ S/m).

samples $\breve{\mathbf{Y}}_k = \breve{\mathbf{Y}}(\mathrm{j}\omega_k)$ are regarded as an "exact" representation of the line behavior, albeit only at discrete frequencies. The thin solid line in Figure 1.10(a) depicts the frequency-dependent magnitude of $\breve{Y}_{11}$ and $\breve{Y}_{12}$.

The second step involves processing this admittance data to obtain a macromodel. We use the VF algorithm to obtain a model in the form

$$\mathbf{Y}(s) = \mathbf{R}_0 + \frac{\mathbf{R}_1}{s - p_1} + \frac{\mathbf{R}_2}{s - p_2} + \cdots + \frac{\mathbf{R}_n}{s - p_n}\,, \tag{1.13}$$

whose responses are compared to the original admittance data in Figure 1.10(a). We see that accuracy is excellent, since the model responses closely match the reference.

It turns out that this model is not passive, since a check based on (1.8) reveals that one of the eigenvalues λ_i is negative in a narrow frequency band close to DC ($s = 0$). This passivity violation occurs because the rational macromodel, although very accurate, is only an approximation of the true responses. The unavoidable approximation error is responsible for the passivity violation. Nevertheless, we perform the required transient analysis, obtaining the results depicted in Figure 1.10(b). At first look, these results might seem correct. However, if the transient simulation is extended, the detrimental effects of passivity violation become clearly visible. Figure 1.10(c) shows in fact that the transient results obtained from the nonpassive model blow up exponentially.

Understanding that passivity is a mandatory requirement, we generate a new passive model by perturbing the residues of the nonpassive model. Full details of this procedure are provided in Chapter 10. The transient analysis is then repeated using the obtained passive model. The results are depicted in Figure 1.10(b) and (c). We see that the early time responses of the two models are practically undistinguishable on this scale. However, we also see that the late-time behavior of the passive model remains bounded, as we expect from physical reasons. The instability has been completely removed.

1.8 WHAT CAN GO WRONG?

The transmission line example discussed in Section 1.7 demonstrated that when proper care is taken in all steps of macromodel construction, the results are expected to be accurate and the macromodel-based simulations will be fast and efficient. However, in

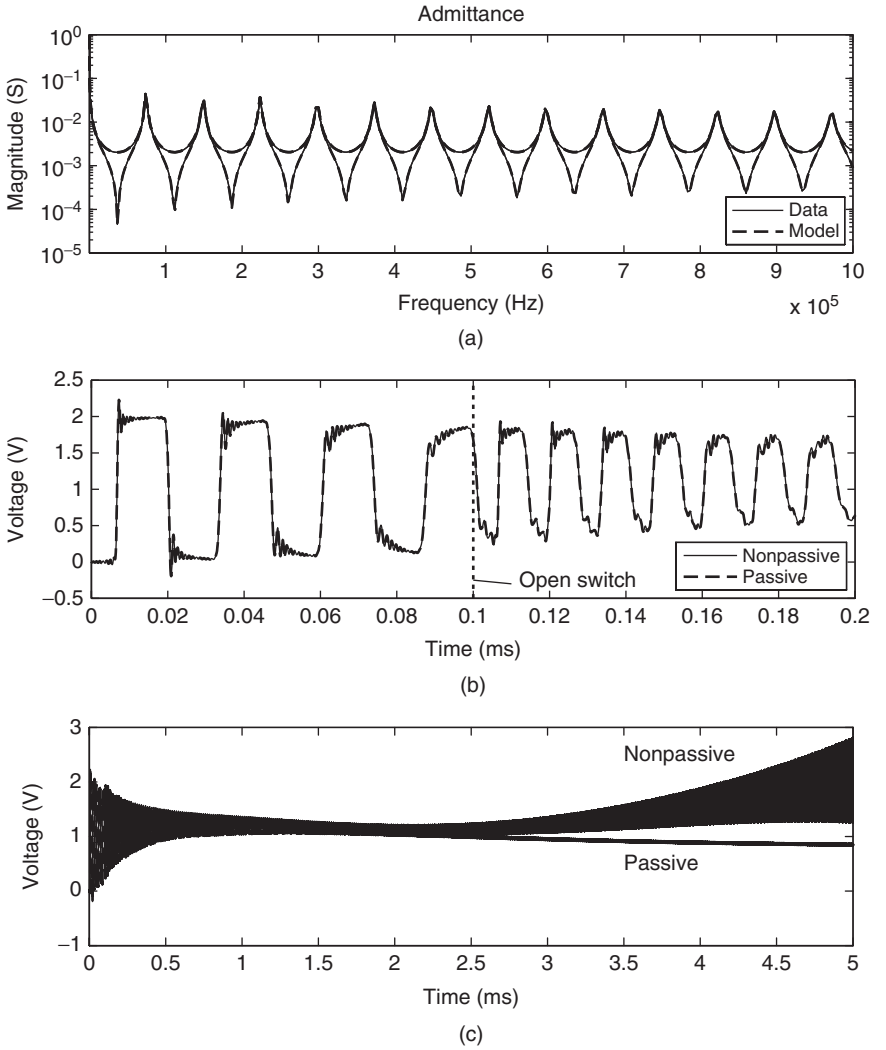

Figure 1.10 (a) Comparison between the exact frequency-dependent admittance responses of the transmission line shown in Figure 1.9 and a corresponding (nonpassive) rational macromodel. (b) Comparison between transient responses $v_2(t)$ obtained from a passive and a nonpassive macromodel. (c) For late time, the nonpassive model drives the simulation to instability.

some situations, the macromodeling procedure may fail. In this section, we review the most common reasons for such failures.

Let us focus on black-box macromodeling flows based on rational fitting of sampled frequency data. Rational functions are universal approximators. In principle, any set of finite data samples can be accurately fitted with a rational function, provided that a sufficient order is considered. Even a set of finite randomly selected complex numbers can be fitted with a rational function, in which case the order will be very large, and the intersample behavior of the interpolant will at best look "strange," possibly with "wild"

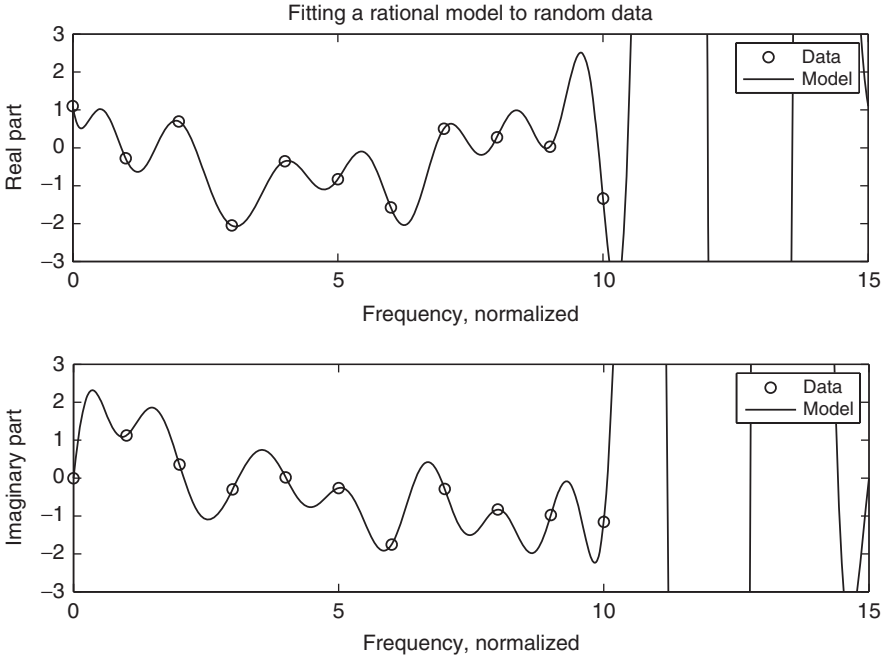

Figure 1.11 Rational approximation of a set of randomly generated complex data samples. The approximation is excellent when observed at the available frequency points. However, the rational function behavior between the samples and especially beyond the last frequency point exhibits very large oscillations.

oscillations. Figure 1.11 provides an illustrative example. This extreme case is included here to stress the fact that the rational fitting problem does not pose particular difficulties *per se*. In fact, difficulties mostly arise when trying to enforce the physical consistency requirements discussed in Section 1.5 to represent data samples that are not compatible in some sense with these constraints.

The most notable case occurs with obvious passivity violations in the original data samples. Let us reconsider the scalar impedance example discussed in Section 1.5, which is characterized by a real part $R(\omega)$ that is negative in a frequency interval (ω_1, ω_2); see Figure 1.6. Whatever fitting algorithm is adopted to construct a rational model $H(s)$, whenever the passivity constraint is enforced, the real part of the model transfer function $H(\mathrm{j}\omega) = H'(\omega) + \mathrm{j}H''(\omega)$ will be nonnegative, $H'(\omega) \geq 0$ for all ω. Therefore, there will be a mismatch $\Delta'(\omega) = |H'(\omega) - R(\omega)|$ that cannot be reduced below a minimum value, dictated by the extent of the passivity violation, that is, the minimum negative value of $R(\omega)$ in (ω_1, ω_2). This situation is illustrated in Figure 1.12, where the minimum distance Δ between the set of passive models and a given nonpassive system $\breve{H}(s)$ is emphasized. We have the following two alternatives.

1. Disregard the passivity constraint, and construct a macromodel $H_n(s)$ by focusing on its accuracy with respect to the given (nonpassive) data $\breve{H}_k = \breve{H}(\mathrm{j}\omega_k)$; in this case, the model will be not passive, shown as filled circle in Figure 1.12.

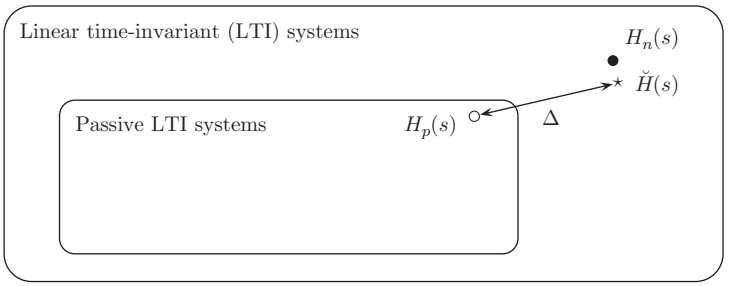

Figure 1.12 The distance between a passive model $H_p(s)$ (open circle) and a nonpassive system $\breve{H}(s)$ (star) cannot be smaller than a minimum amount Δ. Releasing the passivity constraint may lead to a nonpassive model $H_n(s)$ (filled circle) that closely matches the original system.

2. Enforce passivity of the model $H_p(s)$, paying the price of a less accurate fit with respect to the given data samples; this case is shown as open circle in Figure 1.12.

The presence of passivity violations in terms of a negative real part (for scalar systems) or some negative eigenvalues (1.8) for multiport systems may be due to several reasons. When dealing with measurements, noise and improper calibration or deembedding may lead to localized passivity violations. When simulated data obtained from field solvers are used, we should acknowledge that such responses are obtained through a discretization procedure, which is never exact. The unavoidable numerical approximation errors may in fact be the root cause for passivity violations. It is also not uncommon that such issues arise from bad solver settings from nonexpert users, nonphysical assumptions on material properties, and even trivial errors in data postprocessing. Such issues should not be resolved by postfixing inconsistent solver results or corrupt data, but by guaranteeing a proper usage of the solver, with adequate settings and with a physically consistent definition of the underlying electrical or electromagnetic system.

Nonetheless, it is a fact that exported data from field solvers may undergo hidden postprocessing steps that are aimed at "eliminating" any residual passivity violation. Most of the times, this postprocessing makes the inconsistencies in the data even worse. For instance, at the time of this writing, there were several commercial solvers on the market that artificially "clip" negative eigenvalues before making the data available to the user, as depicted in Figure 1.13. This procedure is detrimental, since the resulting responses will be inevitably flawed by another and more subtle type of inconsistency, a causality violation. Since proper understanding of these concepts requires some theoretical background that will be developed in Chapter 2, we only illustrate here the symptoms on a simple test case.

Let us consider again Figure 1.13, where two sampled frequency responses are represented: an original smooth but nonpassive response $\breve{H}(\mathrm{j}\omega_k)$, and a nonsmooth and (only) apparently passive response $\breve{H}_{\mathrm{clip}}(\mathrm{j}\omega_k)$, obtained by clipping its real part so that all negative values are redefined to some small positive values $\varepsilon > 0$. Figure 1.14 shows the results of a rational fitting applied to the two datasets, performed without imposing any passivity constraint, but enforcing model stability so that all poles have a negative

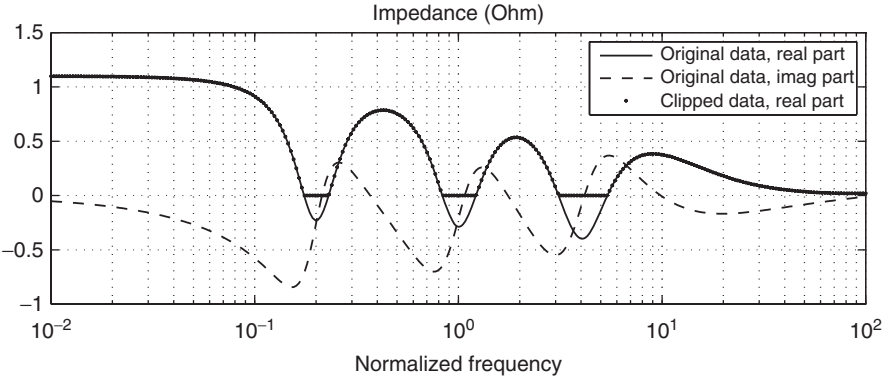

Figure 1.13 Real and imaginary parts of a nonpassive impedance (solid and dashed lines), and real part of the artificially "clipped" responses (dots).

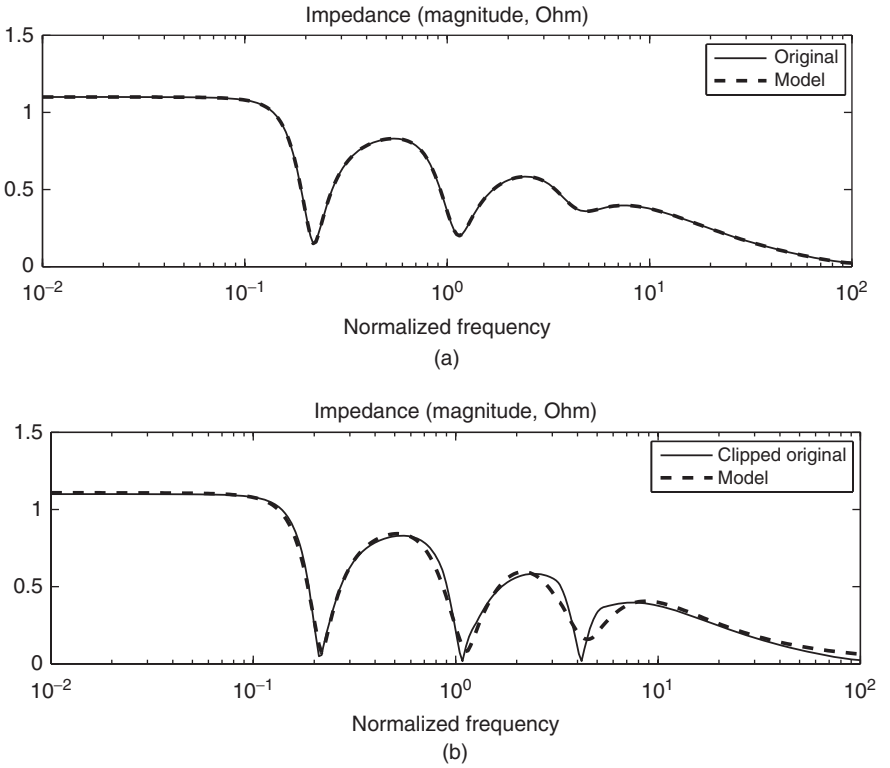

Figure 1.14 (a) Stable rational macromodel (8 poles) computed from original (nonpassive) impedance data. (b) Stable rational macromodel (16 poles) computed from clipped impedance data. (c) Evolution of fitting errors through iterations (the number next to each marker denotes the model order at the corresponding iteration).

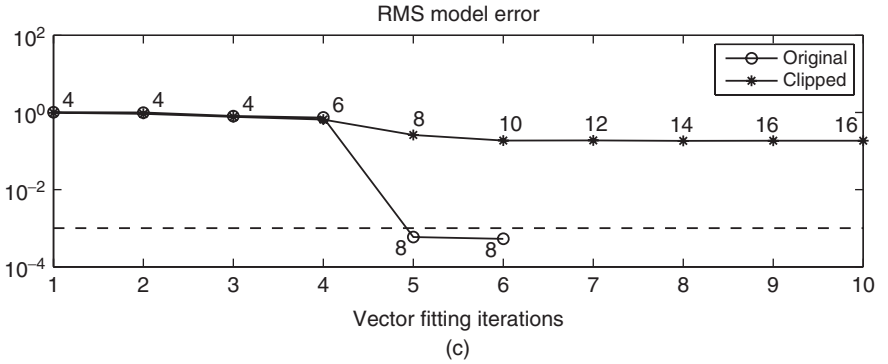

Figure 1.14 (*Continued*)

real part. The model computed from $\breve{H}(\mathrm{j}\omega_k)$ is very accurate, although nonpassive (this is in fact the same situation depicted by the filled circle in Figure 1.12). Conversely, the model computed from $\breve{H}_{\mathrm{clip}}(\mathrm{j}\omega_k)$ is very inaccurate. Moreover, the approximation error does not converge below the desired accuracy threshold $\delta = 10^{-3}$ even when increasing the model order through the iterations (see the progress of the model order used for each iteration, as indicated in panel (c)). We conclude that clipping the real part has induced a new kind of problem in the data, which cannot be fitted with a stable rational function. Based on the discussion in Section 1.5, this is a clear evidence of noncompliance with causality constraints.

In summary, we can state that rational macromodeling with or without passivity constraints is expected to be successful whenever the data from which the models are constructed are consistent with physical requirements. If this is not the case, probably the best approach would be to throw the inconsistent data away, to investigate what is the root cause that led to this inconsistency, and to regenerate the data after fixing the problem. Any step in the construction of macromodels must be well under control, and any heuristic postfix at any stage of the process is very dangerous and should be avoided.