IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

IEEE ELECTRONICS PACKAGING SOCIETY SECTION

Received 21 November 2024, accepted 15 December 2024, date of publication 19 December 2024,
date of current version 6 January 2025.

Digital Object Identifier 10.1109/ACCESS.2024.3520388

RESEARCH ARTICLE

Data-Driven Modeling of Weakly Nonlinear Circuits via Generalized Transfer Function Approximation

ANTONIO CARLUCCI[1], (Member, IEEE), ION VICTOR GOSEA[2], AND STEFANO GRIVET-TALOCIA[1], (Fellow, IEEE)

[1]Department of Electronics and Telecommunications, Politecnico di Torino, Turin 10129, Italy
[2]Max Planck Institute for Dynamics of Complex Technical Systems, CSC Group, 39106 Magdeburg, Germany

Corresponding author: Antonio Carlucci (antonio.carlucci@polito.it)

ABSTRACT This paper presents an extension of the Vector Fitting algorithm with the purpose of constructing compact behavioral models of weakly nonlinear circuits starting from frequency-domain input-output data. Using the concept of generalized transfer function provided by Volterra series theory for nonlinear systems, the algorithm approximates a given dataset of generalized transfer function samples with a black-box multivariate rational model. The fitted model can then be recast into a bilinear state-space form for time-domain analysis. Practical extraction of the required data samples can be carried out by measurement or harmonic balance analysis available in commercial solvers. Examples demonstrating the accuracy and efficiency of the behavioral models include a Low-Dropout Regulator and a Low Noise Amplifier.

INDEX TERMS Behavioral model, nonlinear system, vector fitting, Volterra series, data-driven model, non-intrusive model reduction, generalized transfer function.

I. INTRODUCTION

Recent developments in electronic systems design have been driven by the need for powerful yet compact and energy-efficient devices. As manufacturing technology hits physical limits to transistor scaling, engineers have turned to alternative solutions where functional blocks of heterogeneous nature such as analog, power, and radio-frequency (RF) are tightly integrated and operate in close vicinity [1], [2]. This causes unintended electrical interactions due to parasitic coupling, whose impact should be predicted accurately through numerical simulation to ensure proper functioning. In addition, many analog circuit blocks exhibit some degree of nonlinear behavior, that might be either purposeful (e.g. in an RF mixer) or just an unwanted higher-order effect in a circuit that is meant to operate linearly (e.g. an amplifier).

Any numerical simulation and verification process requires appropriate models of devices, components and their interconnection networks. A brute-force approach based on a combination of first-principle equations (e.g., Maxwell's and/or Kirchhoff's equations) often yields intractable simulation models, both in terms of memory and time required to compute their solution. This is especially true when an entire electronic system made up of several coupled blocks has to be solved repeatedly in the time domain (transient analysis) for design verification or optimization. Therefore, a common trend in Electronic Design Automation (EDA) over the last few decades aims at reducing such complexity through behavioral models, intended as reduced-order compact descriptions of given original high-fidelity representations. More precisely, a behavioral model should reproduce reliably the input-output behavior of a physical device, that is the circuit response to given external stimuli such as voltage/current inputs or other driving signals.

A plethora of techniques to obtain compact models of electronic circuits are available. In particular, model reduction methods based on projection of the circuit equations have been first put forward for large-scale linear networks [3], [4], and later extended to nonlinear ones in [5] and [6]. Alternative techniques are those based on system identification [7],

The associate editor coordinating the review of this manuscript and approving it for publication was Ravi Mahajan.

which are data-driven algorithms that give a behavioral model starting from sampled input-output data describing the circuit behavior, without requiring access to the underlying system equations. Besides being an efficient way to derive low-complexity representations from behavioral data, the data-driven approach is the only option whenever a first-principles description of the system is unavailable (e.g. out of intellectual property issues) or difficult to use, for example in multi-physics simulations or when coupling lumped circuit representations with distributed elements (field-circuit simulations).

Among data-driven methods, the Vector Fitting (VF) algorithm introduced in [8] is an established solution for building reduced models of linear systems using transfer function samples obtained from real or virtual frequency-domain measurements. With VF, these samples are used to optimize the coefficients of a rational transfer function model in order to best approximate the data points. VF has found use in a diverse range of applications including modeling of electromagnetic structures (printed circuit boards, chip packages, microwave components), power systems [9], and even the cardiovascular flow [10]. See [11] for a review. However, despite the ubiquity of nonlinear phenomena in electronic circuits, a VF-like solution built on rational fitting for nonlinear systems is not available yet. This is indeed the gap that this paper intends to fill. We remark that research in this direction led to the so-called NL-VF algorithm [12], [13], whose purpose is however quite different: NL-VF does not provide a nonlinear dynamical model, but a linear representation of the map between a set of input excitation signals to the corresponding output responses, constructed as a rational transfer function in the Laplace domain.

Modeling of nonlinear systems has been thoroughly researched in various disciplines and application fields, using widely different approaches such as block-oriented models [14], [15] (e.g. Wiener-Hammerstein), NARMAX [16], Volterra series [17], [18], [19], Machine Learning [20], Koopman operator theory [21]. Among these, the framework of the Volterra series for nonlinear system analysis has attracted enormous interest in engineering fields for a long time since its introduction. This is because it provides a rigorous mathematical tool to represent nonlinear systems (or *functionals* in its original version) using a series expansion, with the same intuitive interpretation as the more familiar Taylor series used to locally approximate a function.

The Volterra formalism has been successfully used for several purposes including RF circuit modeling [22], [23], analog design [24], ADC (Analog-to-Digital Converter) predistortion [25]. A particularly appealing feature is that it allows extending the transfer function concept to the nonlinear domain as described in [26] and [27], thus offering a way to generalize methods that were originally conceived for linear systems only. Such generalizations exist for model reduction algorithms based on Krylov subspace projection [6], [28], [29], [30], H_2-optimal interpolation [31], [32], [33] and the Loewner identification framework [34], [35].

Behavioral modeling through the Volterra series has been addressed in several works. In [22], a mixed discrete-time/frequency-domain method is proposed to identify the Volterra kernels or transfer functions based on input-output measurements. In [36], the authors compute Volterra transfer functions efficiently starting from X-parameters data (for a detailed definition, see [36] and references therein). In these works and in most of the existing literature, the behavioral model is intended as the collection of Volterra kernel values, discretely sampled in time or frequency (i.e. non-parametric approach). This is because, in principle, these suffice to approximately find the model response to a given input by multidimensional convolution, as discussed e.g. in [37].

In this paper, a different (parametric) approach is taken: the Volterra formalism is applied to a given weakly nonlinear circuit to define an appropriate set of *Generalized Transfer Functions (GTF)*, whose frequency samples are easily computed or measured using known methods [36], [38], [39], [40], [41], [42]. Starting from such samples, we construct a dynamical model expressed as a set of nonlinear state equations. Such a model can then be solved by time-domain integration to find the output response. In addition to superior accuracy, the benefit of turning the raw transfer function data into a dynamical model instead of directly applying convolution is the possibility for this model to co-exist with other components that have a standard equation- or circuit-based description. In other words, the proposed models can be used as black-box generalized components in a system-level simulation, and not only in forward numerical evaluation of the output signals given some known excitation.

To build the proposed nonlinear behavioral model, we resort to a rational approximation of the GTFs, achieved through a generalization of the VF algorithm, which constitutes the main novel contribution of this work. In addition to the above rational transfer function model, the corresponding time-domain counterpart in the form of a bilinear state-space realization is provided. Such form is known to be a universal approximator of nonlinear dynamic maps [43]. This behavioral model can be easily synthesized into a SPICE netlist as a sequence of linear blocks (whose synthesis is detailed in [11]) interleaved with static nonlinearities.

To summarize, this work provides a comprehensive framework and an associated algorithmic solution to compute nonlinear behavioral models starting from input-output frequency-domain sampled responses. Such models are to be used to run efficient numerical simulations for design verification and/or optimization. The proposed approach is *non-intrusive*, as it does not require access to the governing equations of the device to be modeled. The only requirement is the ability to *measure* or *simulate* the original system via any (commercial) circuit solver (e.g. through harmonic balance or transient analysis, X-parameters, etc.).

The performance of the proposed approach is demonstrated in three test cases of practical relevance. The first proof of concept is a Low-Dropout regulator (LDO) circuit to demonstrate the accuracy and efficiency of the approach. The second example is a large-scale post-layout model of an LDO after parasitic extraction (in Cadence Virtuoso [44]). The third example is a Low-Noise Amplifier (LNA) with an accurate EM characterization modeled in Keysight ADS [45]. These examples show the applicability and practical advantages of the proposed modeling algorithm in terms of simulation speedup.

II. BACKGROUND AND NOTATION

A. NOTATION

In this paper, we use boldface italic letters to denote vectors ($\boldsymbol{x}$) and boldface symbols ($\mathbf{X}$) for matrices. The matrix transpose is $\mathbf{X}^{\mathrm{T}}$, the identity matrix of dimension n is $\mathbb{I}_n$, the n-dimensional vector of all ones is $\mathbf{1}_n$. The notation $\mathbf{A} \otimes \mathbf{B}$ is the Kronecker product between matrices $\mathbf{A}$ and $\mathbf{B}$. Calligraphic symbols such as $\mathcal{R}$ are tensors. The m-way tensor $\mathcal{R}$ is an m-dimensional matrix whose individual entries $r_{n_1,\dots,n_m}$ are indexed by a multi-index $(n_1, \dots, n_m)$, containing a positive integer for each dimension. Following [46], the symbol $\mathcal{R}_{(j)}$ is the matricization of $\mathcal{R}$ along dimension j, i.e. a matrix whose columns are the fibers of mode j. The mode-j tensor-matrix product $\mathcal{R} \times_j \mathbf{A}$ is obtained by multiplying the matrix $\mathbf{A}$ with the mode-j fibers of $\mathcal{R}$. In other words, $\mathcal{R} \times_j \mathbf{A}$ is the tensor whose matricization is $\mathbf{A}\mathcal{R}_{(j)}$. In addition, the vectorization operator $\mathrm{vec}\{\mathcal{R}\}$ or $\mathrm{vec}\{r_{n_1,\dots,n_m}\}$ returns a column vector collecting all entries $r_{n_1,\dots,n_m}$ of $\mathcal{R}$.

The letter s is the Laplace variable, the symbols f and $\omega = 2\pi f$ respectively indicate the frequency and angular frequency, and the imaginary unit is j.

With $\mathbb{P}_m$ we indicate the set of permutations of the first m natural numbers, with each element $\sigma \in \mathbb{P}_m$ being a one-to-one function from $\{1, \dots, m\}$ into itself.

B. VOLTERRA SERIES

A nonlinear one-port circuit can be represented as a nonlinear system, mapping some input signal $u(t) \in \mathbb{R}$ (e.g. input current) to the corresponding output signal $y(t) \in \mathbb{R}$ (e.g. voltage response). We restrict the discussion to time-invariant networks and denote the nonlinear map associated with the system G as $y(t) = G[u](t)$.

An established tool for input-output representation of nonlinear maps is the Volterra series [17], [19], that provides an expansion of the system response $y(t)$ in terms of multidimensional convolutions of the input $u(t)$ with the multivariate kernels $h_m(\tau_1, \dots, \tau_m)$, which are characteristic functions of the system to be represented, as follows

$$y(t) = y_0 + \sum_{m=1}^{\infty} \underbrace{\int_0^t h_m(\tau_1, \dots, \tau_m) u(t-\tau_1) \cdots u(t-\tau_m)\, \mathrm{d}\boldsymbol{\tau}}_{\triangleq G_m[u]}. \tag{1}$$

In (1), the symbol $\mathrm{d}\boldsymbol{\tau} \triangleq \mathrm{d}\tau_1 \cdots \mathrm{d}\tau_m$ indicates that the m-th summand is a multiple integral across the rectangular domain $[0, t]^m \ni (\tau_1, \dots, \tau_m)$. Apart from the affine constant y_0, the first summand (i.e. $m = 1$) is the response of the linear subsystem $G_1[u]$, which coincides with the well-known convolution formula valid for linear time-invariant (LTI) systems. In addition, further terms with $m \geq 2$ represent higher-order nonlinear contributions $y_m(t)$ from the *homogeneous* subsystems $G_m[u]$, so called because of the homogeneity property $G_m[\alpha u] = \alpha^m G_m[u]$, $\forall \alpha \in \mathbb{R}$. The Volterra series is locally valid around an expansion point u_0 (e.g. $u(t) = 0$) and is guaranteed to converge when the amplitude $|u(t)|$ is small enough. Therefore, a fruitful application of the Volterra series requires assuming that the underlying system is *weakly nonlinear*, meaning that the first few terms of the Volterra series are sufficient to approximate its input-output behavior [47]. In fact, although Volterra series can, in principle, approximate a large class of nonlinear systems [47], this might require prohibitively many terms. Hence, we focus on weakly nonlinear systems. We refer the Reader to [48] for a complete discussion regarding convergence.

Note that the linear subsystem $G_1[u]$, associated with the impulse response $h_1(\tau)$, is the small-signal approximation of the circuit, linearized around the operating point u_0. In large-signal conditions, it will not provide an accurate approximation of the system behavior, which would require taking higher-order terms into account.

For a given subsystem G_m, the choice of h_m is non-unique. However, ambiguity is removed if we assume (without loss of generality) that every h_m in (1) is a symmetric function, whose value is invariant under arbitrary permutation of its arguments. This leads to a unique definition of the *symmetric* kernel, which is also particularly convenient because its values can be sampled from input-output experiments (either numerical simulations or actual measurements), without requiring direct access to the first-principles equations underpinning the system to be modeled. For this reason, by h_m we indicate the symmetric kernels hereinafter.

An alternative representation is the *regular* kernel h_{reg} defined in [17, Ch.1], which corresponds to a different form of the convolution integrals in the series (1). Originally introduced for realization theory, it has been proficiently exploited in model reduction [34]. However, in practical applications, it is less effective for non-intrusive modeling purposes because regular TF values are not easily inferred from input-output data, as noted in [49] and [50].

The expansion (1) admits a frequency-domain counterpart through the (multivariate) Laplace transform. In fact, transforming h_m gives the degree-m GTF $H_m(s_1, \dots, s_m)$ defined as follows (see [51])

$$H_m(s_1, \dots, s_m) = \int_0^{\infty} \cdots \int_0^{\infty} h(\tau_1, \dots, \tau_m) e^{-\sum_{i=1}^{m} s_i \tau_i}\, \mathrm{d}\boldsymbol{\tau} \tag{2}$$

Samples $H_m(s_1, \ldots, s_m)$ can be extracted experimentally from steady-state responses under periodic excitation, through the *harmonic probing* method of [38], also discussed and enhanced in [39] and [42].

Laplace transform can also be applied to the m-th regular kernel $h_{\mathrm{reg},m}$ to obtain the regular transfer function (TF) $H_{\mathrm{reg},m}(s_1, \ldots, s_m)$. In this work, the regular TF is only used instrumentally because of the following useful link with the symmetric GTF [17],

$$H_m(s_1, \ldots, s_m) = \frac{1}{m!} \sum_{\sigma \in \mathbb{P}_m} H_{\mathrm{reg},m}(s_{\sigma(1)}, s_{\sigma(1)+\sigma(2)}, \ldots, s_{\sigma(1)+\cdots+\sigma(m)}). \quad (3)$$

A comprehensive overview of Volterra series theory can be found in [18].

C. BILINEAR SYSTEM

We consider the class of *bilinear systems* as a template for modeling nonlinearities. Bilinear systems are characterized by dynamic equations including products of inputs and states and admit the following state-space representation

$$\dot{\boldsymbol{x}} = \mathbf{A}\boldsymbol{x} + \mathbf{N}\boldsymbol{x}u + \boldsymbol{b}u \quad (4a)$$

$$y = \boldsymbol{c}^{\mathrm{T}}\boldsymbol{x} \quad (4b)$$

where $\boldsymbol{x}$ is the state vector, and the coefficients $\mathbf{A}, \mathbf{N} \in \mathbb{R}^{\nu \times \nu}$, and $\boldsymbol{b}, \boldsymbol{c} \in \mathbb{R}^{\nu}$ provide the state-space realization. Here, nonlinearity arises from the product $\boldsymbol{x}u$ in (4a).

Despite the seemingly restrictive bilinear structure, it has been shown that it has a *universal approximation* property because, under some causality and continuity assumptions, nonlinear input-output maps can be approximated arbitrarily well by a bilinear model [43]. Moreover, there are constructive ways to recast practically relevant nonlinear systems into a bilinear format either exactly or approximately using state variable transformations. For instance, the Carleman bilinearization [17], [52] is a general analytical procedure that yields a bilinear approximation starting from the state equations of a given input-affine system. In addition, exact lifting techniques can be applied in certain problem instances to obtain an equivalent quadratic, or quadratic-bilinear representation [6], [30]. However, these are intrusive in that they require knowing and being able to manipulate a set of nonlinear differential equations describing the system under analysis.

Following [17] and [31], we now recall the regular TFs of the bilinear system (4a)-(4b). Starting from $m = 1$, the degree-1 regular TF is

$$H^{\mathrm{bil}}_{\mathrm{reg},1}(s_1) = \boldsymbol{c}^{\mathrm{T}}(s_1\mathbb{I} - \mathbf{A})^{-1}\boldsymbol{b}. \quad (5)$$

For $m \geq 2$, the higher-order regular TFs are the following separable functions,

$$H^{\mathrm{bil}}_{\mathrm{reg},m}(s_1, \ldots, s_m) = \boldsymbol{c}^{\mathrm{T}}(s_m\mathbb{I} - \mathbf{A})^{-1}\mathbf{N}(s_{m-1}\mathbb{I} - \mathbf{A})^{-1} \mathbf{N} \cdots (s_1\mathbb{I} - \mathbf{A})^{-1}\boldsymbol{b}. \quad (6)$$

III. PROBLEM STATEMENT AND CONTRIBUTION

The purpose of this paper is to describe a numerical method to infer a bilinear dynamical model whose input-output behavior matches the observed response of a given weakly nonlinear system, known through sampled values of the GTF in a certain frequency range and up to a predefined degree M. To this aim, we introduce the dataset $\mathcal{D}$ containing, for each degree m ranging from 1 to M, a set of K_m points consisting in the m-tuples $\boldsymbol{s}_m^{(k)} \triangleq (s_1^{(k)}, \ldots, s_m^{(k)})$, with $k = 1, \ldots, K_m$, paired with the GTF samples $\check{H}_m(s_1^{(k)}, \ldots, s_m^{(k)}) = \check{H}_m^{(k)}$ evaluated at the respective points. In symbols,

$$\mathcal{D} = \left\{ \left(\boldsymbol{s}_m^{(k)}, \check{H}_m^{(k)} \right), m = 1, \ldots, M, \ k = 1, \ldots, K_m \right\}. \quad (7)$$

We pursue the objective of finding a bilinear model whose generalized transfer functions match the observed $\check{H}_m^{(k)}$ as closely as possible. The resulting algorithm could be part of a workflow as depicted in Fig. 1, where the initial dataset $\mathcal{D}$ is extracted from measurements of harmonic steady-state responses. Despite the demonstrated efficiency of transfer function fitting methods in deriving low-complexity black-box models from sampled data in the LTI context, a generalization of rational macromodeling (e.g. VF, Adaptive Antoulas-Anderson (AAA) [53]) to nonlinear circuits is not yet available and constitutes the gap addressed here. An exception to this is the algorithm of [54], which has quite a different purpose as it is designed for the special class of linear systems with quadratic output. Another related work is based on the Loewner framework, by means of using input-output data (e.g., samples of symmetric transfer functions). This addresses bilinear [55] but also quadratic identification [49], [50]. Additionally, [56, Sec. 4.3] addresses bilinear identification in the Loewner framework using symmetric transfer function data evaluated for particular combinations of the arguments (all equal). The proposed approach differs from [55] as the approximation process also takes into account GTFs of degree larger than two, which also contribute to determining the *dynamic part* of the proposed model (the matrix $\mathbf{A}$).

In this paper, we choose to focus on extending the VF algorithm because it has several desirable properties such as flexibility in the choice of cost function and ease of stability enforcement. To this aim, the foregoing sections describe in detail the novel contributions of this work, in particular:

- a derivation of the GTFs of a conveniently chosen bilinear model structure in pole-residue form in Sec. IV. The selected model format leverages a set of recursively defined rational basis functions that allow to cast the identification problem as a simple linear regression (Sec. IV-D).
- an extension of the VF algorithm for identifying poles of GTFs in Sec. V. Its main feature is a recursive strategy relying on a sequence of linear least-squares optimization steps, which are notoriously scalable and robustly solvable.

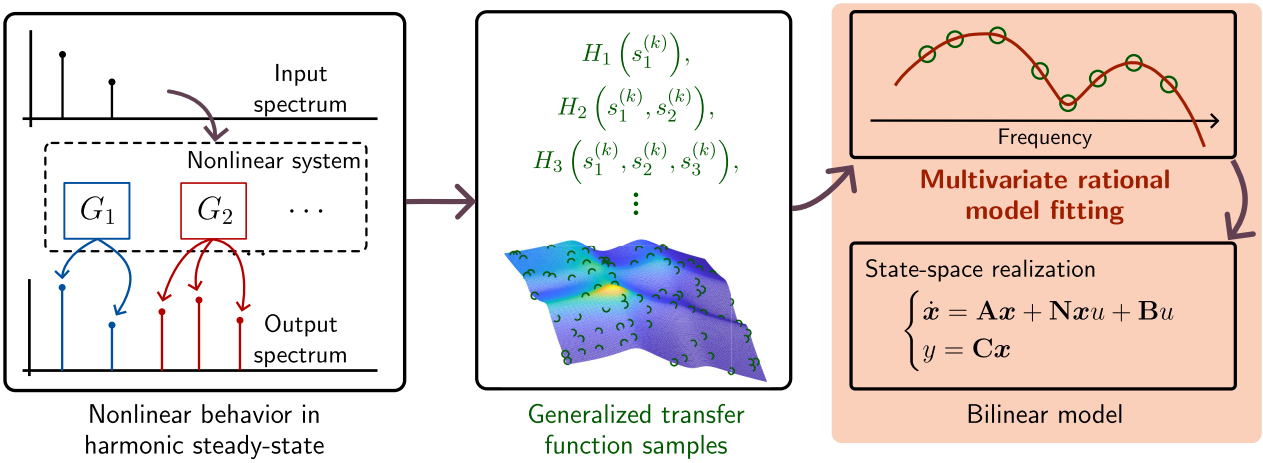


FIGURE 1. Outline of the nonlinear modeling methodology based on Volterra series, starting from input/output measurements up to a bilinear model built through rational approximation of GTF data samples.

- numerical examples of the approximation capability and practical advantages in Sec. VII. The benchmarks include an LDO voltage regulator and an LNA circuit.

Compared to the early results we reported in [57], this work contains substantial improvements and additions. In particular, the newly-introduced multivariate pole-residue expansion of Sec. IV provides a new model structure based on rational basis functions that is absent in [57]. This allows independent estimation of coefficients of different subsystems, thus avoiding an error accumulation problem affecting the approach of [57]. Consequently, the bilinear model realization described herein has been improved to accommodate this more general model format. As in [57], pole estimation relies on a sequence of weighted VF steps (Sec. V). However, this phase has been reworked to use a different choice of weighting vectors arising from the new model format.

IV. GENERALIZED TRANSFER FUNCTION FITTING

A. BILINEAR GTF

Let us begin with an analysis of the GTFs of the bilinear system defined by (4a)-(4b). Expanding the system around $u_0 = 0$, the degree-1 bilinear GTF $H_1^{\text{bil}}(s_1)$ corresponds to the small-signal linearization at $\boldsymbol{x} = 0$, and it coincides with the regular TF $H_{\text{reg},1}^{\text{bil}}(s_1)$. Higher-order symmetric GTFs H_m^{bil} are found by combining (6) and (3) to obtain the compact expression

$$H_m^{\text{bil}}(s_1, \ldots, s_m) = \frac{1}{m!} \sum_{\sigma \in \mathbb{P}_m} \boldsymbol{c}^{\mathrm{T}} \boldsymbol{X}_m(s_{\sigma(1)}, \ldots, s_{\sigma(m)}) \tag{8}$$

where the auxiliary function $\boldsymbol{X}_m(s_1, \ldots, s_m) \in \mathbb{C}^{\nu}$ is defined recursively starting with $\boldsymbol{X}_1(s_1) = (s_1\mathbb{I} - \mathbf{A})^{-1}\boldsymbol{b}$ according to the rule

$$\boldsymbol{X}_m(s_1, \ldots, s_m) = \left[\left(\textstyle\sum_{i=1}^{m} s_i\right)\mathbb{I} - \mathbf{A}\right]^{-1} \mathbf{N}\boldsymbol{X}_{m-1}(s_1, \ldots, s_{m-1}) \tag{9}$$

for $m \geq 2$. In fact, a simple substitution of (9) into (8) shows that it is equivalent to plugging (6) into the identity (3). This recursive definition of $\boldsymbol{X}_m$ is instrumental to the following developments.

Let us assume that $\mathbf{A}$ has simple eigenvalues $\lambda(\mathbf{A})$, so that $H_1^{\text{bil}}(s_1)$ has ν distinct poles $\{p_n\}_{n=1}^{\nu} = \lambda(\mathbf{A})$. Then its pole-residue expansion is a linear combination of partial fractions from the set $\Phi_1 = \{(s_1 - p_n)^{-1}\}_{n=1}^{\nu}$. Given that the pole-residue form offers a non-redundant parameterization of the system coefficients, which makes it particularly suited for model estimation tasks, we follow the same idea and attempt to formulate a pole-residue expansion for higher-degree GTFs. This is in contrast with previous works on bilinear system identification such as [34], which have focused on estimating a state-space realization or have considered pole-residue expansions of the regular kernel instead of the symmetric one [31]. Therefore, we now derive the set of multivariate rational basis functions spanning all possible H_m^{bil} defined by (8), even for higher degrees $m \geq 2$. To this aim, Eq. (9) suggests taking a recursive approach. Using a more general notation that will prove useful later, let us introduce a family of M pole sets $\{p_n^{(1)}\}_{n=1}^{\nu_1}, \ldots, \{p_n^{(M)}\}_{n=1}^{\nu_M}$. These appear in the definition of the following sequence

$$\varphi_{n_1} = \left(s_1 - p_{n_1}^{(1)}\right)^{-1}, \qquad 1 \leq n_1 \leq \nu_1 \tag{10a}$$

$$\varphi_{n_1,\ldots,n_m} = \left(s_1 + \cdots + s_m - p_{n_m}^{(m)}\right)^{-1} \varphi_{n_1,\ldots,n_{m-1}}, \qquad 1 \leq n_m \leq \nu_m. \tag{10b}$$

This definition is relevant because the family of pole sets can be chosen so that $\boldsymbol{X}_m$ belongs to the span of the following set Φ_m of rational basis functions,

$$\Phi_m = \{\varphi_{n_1,\ldots,n_m}(s_1, \ldots, s_m),\ 1 \leq n_j \leq \nu,\ \forall j = 1, \ldots, m\}. \tag{11}$$

In other words, any vector-valued function $\boldsymbol{X}_m$ compatible with the recursion (9) can be written as a linear combination

of the basis functions in the set Φ_m, provided that the basis poles sets are chosen suitably, i.e., in this case, they coincide with the system poles $\{p_n^{(m)}\}_{n=1}^{\nu_m} = \lambda(\mathbf{A})$, $\forall\, m = 1, \ldots, M$.

Based on (8), the basis functions for the corresponding expansion of the symmetric GTF H_m^{bil} are the symmetrization of individual elements of Φ_m,

$$\varphi^{\text{sym}}_{n_1,\ldots,n_m}(s_1, \ldots, s_m) = \frac{1}{m!} \sum_{\sigma \in \mathbb{P}_m} \varphi_{n_1,\ldots,n_m}(s_{\sigma(1)}, \ldots, s_{\sigma(m)}) \tag{12}$$

Note that the symmetrization consists in summing over all permutations σ of the m arguments and dividing by their number $m!$. The set of basis functions $\{\varphi^{\text{sym}}_{n_1,\ldots,n_m}\}$ can be seen as the *bilinear* analog of the partial fraction basis underpinning the model representation of the VF iteration.

Example 1 (Basis function set with two poles): To get an intuitive picture of the construction (10), consider the pole sets $\{p_1^{(1)}, p_2^{(1)}\}$, $\{p_1^{(2)}, p_2^{(2)}\}$, i.e. $\nu_1 = \nu_2 = 2$. The degree-1 basis functions are

$$\varphi_1(s_1) = \frac{1}{s_1 - p_1^{(1)}}, \quad \varphi_2(s_1) = \frac{1}{s_1 - p_2^{(1)}} \tag{13}$$

and the degree-2 ($m = 2$) ones are

$$\begin{aligned}
\varphi_{1,1}(s_1, s_2) &= \frac{1}{(s_1 + s_2 - p_1^{(2)})(s_1 - p_1^{(1)})}, \\
\varphi_{1,2}(s_1, s_2) &= \frac{1}{(s_1 + s_2 - p_2^{(2)})(s_1 - p_1^{(1)})}, \\
\varphi_{2,1}(s_1, s_2) &= \frac{1}{(s_1 + s_2 - p_1^{(2)})(s_1 - p_2^{(1)})}, \\
\varphi_{2,2}(s_1, s_2) &= \frac{1}{(s_1 + s_2 - p_2^{(2)})(s_1 - p_2^{(1)})}.
\end{aligned} \tag{14}$$

Each one of them can be symmetrized according to (12) by summing the function values at all possible permutations of the arguments (only two permutations in this case). For example, the symmetrization of $\varphi_{1,2}(s_1, s_2)$ is

$$\begin{aligned}
\varphi^{\text{sym}}_{1,2}(s_1, s_2) &= {}^1\!/_2[\varphi_{1,2}(s_1, s_2) + \varphi_{1,2}(s_2, s_1)] \\
&= \frac{1}{2} \frac{1}{s_1 + s_2 - p_2^{(2)}} \left(\frac{1}{s_1 - p_1^{(1)}} + \frac{1}{s_2 - p_1^{(1)}} \right)
\end{aligned} \tag{15}$$

Example 2 (GTFs of a bilinear system): This example illustrates the observation made in this section that the GTFs H_m^{bil} arising from a bilinear system can be expressed as a linear combination of the basis functions $\{\varphi^{\text{sym}}_{n_1,\ldots,n_m}\}$. Consider a simple bilinear system with realization $(\mathbf{A}, \mathbf{N}, \boldsymbol{b}, \boldsymbol{c})$ whose state matrix $\mathbf{A} \in \mathbb{R}^{2\times 2}$ has simple eigenvalues $\lambda(\mathbf{A}) = \{q_1, q_2\}$. Then, for some constant vectors $\boldsymbol{X}_{1,1}, \boldsymbol{X}_{1,2}$, the function $\boldsymbol{X}_1(s)$ can be expanded as follows

$$\boldsymbol{X}_1(s_1) = (s_1 - q_1)^{-1} \boldsymbol{X}_{1,1} + (s_1 - q_2)^{-1} \boldsymbol{X}_{1,2}. \tag{16}$$

Using (9) to find $\boldsymbol{X}_2(s_1, s_2)$, the factor $[(s_1 + s_2)\mathbb{I} - \mathbf{A}]^{-1}\mathbf{N}$ can also be expanded as

$$\begin{aligned}
[(s_1 + s_2)\mathbb{I} - \mathbf{A}]^{-1}\mathbf{N} = {} & (s_1 + s_2 - q_1)^{-1}\mathbf{N}_{1,1} \\
& + (s_1 + s_2 - q_2)^{-1}\mathbf{N}_{1,2}
\end{aligned} \tag{17}$$

for some constant matrices $\mathbf{N}_{1,1}$, $\mathbf{N}_{1,2}$. Hence, using (9),(16), (17), $\boldsymbol{X}_2(s_1, s_2)$ can be written as

$$\begin{aligned}
\boldsymbol{X}_2(s_1, s_2) = {} & \underbrace{(s_1 + s_2 - q_1)^{-1}(s_1 - q_1)^{-1}}_{\varphi_{1,1}(s_1,s_2)} \boldsymbol{X}_{2,11} \\
& + \underbrace{(s_1 + s_2 - q_1)^{-1}(s_1 - q_2)^{-1}}_{\varphi_{2,1}(s_1,s_2)} \boldsymbol{X}_{2,21} \\
& + \underbrace{(s_1 + s_2 - q_2)^{-1}(s_1 - q_1)^{-1}}_{\varphi_{1,2}(s_1,s_2)} \boldsymbol{X}_{2,12} \\
& + \underbrace{(s_1 + s_2 - q_2)^{-1}(s_1 - q_2)^{-1}}_{\varphi_{2,2}(s_1,s_2)} \boldsymbol{X}_{2,22}.
\end{aligned} \tag{18}$$

Hence, this shows $\boldsymbol{X}_2(s_1, s_2) \in \text{span}\, 8_2$, where the set 8_2 is $8_2 = \{\varphi_{1,1}, \varphi_{2,1}, \varphi_{1,2}, \varphi_{2,2}\}$. The pole sets defining 8_2 are $\{p_n^{(1)}\}_{n=1}^2$ and $\{p_n^{(2)}\}_{n=1}^2$, with $p_1^{(1)} = p_1^{(2)} = q_1$, $p_2^{(1)} = p_2^{(2)} = q_2$. Using (8) and the definition (12), one can also directly verify that the GTF $H_2^{\text{bil}}(s_1, s_2) \in \text{span}\{\varphi^{\text{sym}}_{n_1,n_2}\}_{n_1=1,n_2=1}^2$.

B. BLACK-BOX MODEL STRUCTURE

Section IV-A showed that the GTFs of a generic bilinear system (20) are obtained by a linear combination of a fundamental set of rational basis functions φ^{sym}'s. Therefore, we define the GTF model structure used for fitting the data samples $\mathcal{D}$ to be a linear combination of these φ^{sym}'s. Note that the basis depends on a yet unknown family of pole sets, whose identification will be addressed later. For our purposes, there is no harm in increasing the generality by allowing different pole sets for each degree, a choice adopted in the following definition.

Definition 1 (Bilinear GTF model): Given M sets of model poles $\left\{p_n^{(1)}\right\}_{n=1}^{\nu_1}, \ldots, \left\{p_n^{(M)}\right\}_{n=1}^{\nu_M}$, the bilinear GTF model $H_m(s_1, \ldots, s_m)$ of degree m is defined as

$$H_m(s_1, \ldots, s_m) = \sum_{n_1=1}^{\nu_1} \cdots \sum_{n_m=1}^{\nu_m} r_{n_1,\ldots,n_m} \varphi^{\text{sym}}_{n_1,\ldots,n_m}(s_1, \ldots, s_m) \tag{19}$$

where $r_{n_1,\ldots,n_m} \in \mathbb{C}$ are constant coefficients (termed generalized residues) and the multivariate functions $\varphi^{\text{sym}}_{n_1,\ldots,n_m}$ are defined in eqs. (12), (10).

Note that, differently from the analysis in Sec. IV-A, which proceeds from state-space to transfer function coefficients, Definition 1 goes backward and postulates a pole-residue-like expansion first. Although it is based on the same basis functions φ^{sym}, all generalized residues $r_{n_1,\ldots,n_m}$ are here regarded as independent constants. This will not be the case if one starts from a given $(\mathbf{A}, \mathbf{N}, \boldsymbol{b}, \boldsymbol{c})$, because the resulting expansion coefficients would be ratios of

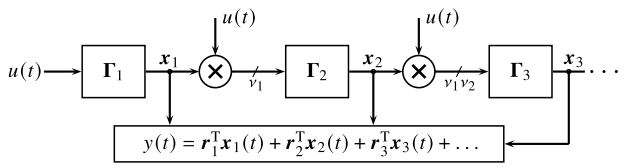


FIGURE 2. Block representation of the bilinear model structure (20), featuring a finite number of nonzero Volterra kernels.

polynomials depending on the entries of the realization matrices, hence more difficult to estimate and optimize.

Therefore, a procedure to find a bilinear state-space realization of the GTF model (19) should be explicitly provided. The fundamental theory of bilinear realizations has been thoroughly investigated, for instance, in [58] and [59], and an algorithm to extract state-space coefficients starting from a regular transfer function with factored denominator is described in [17, Ch. 4]. In the following, we adapt the procedure from [17] to the model (19) which, although not regular, can still be addressed analogously.

Proposition 1 (GTF model realization): For all degrees $m = 1, \dots, M$, the collection of transfer functions (19) admit the bilinear realization

$$\dot{\boldsymbol{x}}_1 = \mathbf{A}_1 \boldsymbol{x}_1 + \mathbf{1} u \tag{20a}$$

$$\dot{\boldsymbol{x}}_m = (\mathbf{A}_m \otimes \mathbb{I}_{N_{m-1}})\boldsymbol{x}_m + \mathbf{1}_{\nu_m} \otimes \boldsymbol{x}_{m-1} u, \quad 2 \le m \le M \tag{20b}$$

$$y_m = \boldsymbol{r}_m^{\mathrm{T}} \boldsymbol{x}_m, \quad 2 \le m \le M \tag{20c}$$

where $\mathbf{A}_m = \mathrm{diag}\{p_n^{(m)}\}_{n=1}^{\nu_m}$, $\boldsymbol{x}_m \in \mathbb{C}^{N_m}$, and $N_m = \nu_1 \cdots \nu_m$. In the output equations (20c), the vector $\boldsymbol{r}_m = \mathrm{vec}\{r_{n_1,\dots,n_m}\}$ collects the generalized residues of the degree-m GTF model (19). The overall output is the superposition of contributions from individual homogeneous subsystems, i.e.

$$y(t) = \sum_{m=1}^{M} y_m(t) = \sum_{m=1}^{M} \boldsymbol{r}_m^{\mathrm{T}} \boldsymbol{x}_m(t) \tag{21}$$

The system (20) results from gluing together a sequence of M subsystems from degree 1 up to M, each being in the modal realization. This construction is illustrated in Fig. 2, which clarifies how this bilinear system consists of a chain of linear blocks $\boldsymbol{\Gamma}_1, \boldsymbol{\Gamma}_2, \dots, \boldsymbol{\Gamma}_M$ interleaved with a multiplicative nonlinearity (static multiplication with the input u). The linear blocks correspond to individual equations (20a), (20b), as follows

$$\boldsymbol{\Gamma}_1(s) = (s\mathbb{I} - \mathbf{A}_1)^{-1}\mathbf{1}_{\nu_1}$$
$$\boldsymbol{\Gamma}_m(s) = [(s\mathbb{I} - \mathbf{A}_m)^{-1}\mathbf{1}_{\nu_m}] \otimes \mathbb{I}_{N_{m-1}}, \quad 2 \le m \le M. \tag{22}$$

Remark 1: Realization (20) has complex coefficients. In case H_m has real symmetry, with pole sets that are closed under complex conjugation, a change of coordinate is sufficient to turn (20) into an equivalent realization with real coefficients. Assume that the pole set $\{p_n^{(m)}\}_{n=1}^{\nu_m}$ is such that the first ν_m^{real} poles are real and the following $2\nu_m^{cplx}$ ones are complex conjugate pairs arranged adjacently, i.e.

$$p^{(m)}_{\nu_m^{real}+2h-1} = \bar{p}^{(m)}_{\nu_m^{real}+2h}, \quad h = 1, \dots, \nu_m^{cplx}$$

The transformation matrix

$$\mathbf{T}_m = blkdiag\left\{\mathbb{I}_{\nu_m^{real}},\ \mathbb{I}_{\nu_m^{cplx}} \otimes \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & \mathrm{j} \\ 1 & -\mathrm{j} \end{pmatrix}\right\}.$$

is such that $\mathbf{T}_m^{-1}\mathbf{A}_m\mathbf{T}_m \triangleq \mathbf{A}_m^{real}$ is real and in quasi-diagonal form (consisting of 2×2 diagonal blocks). Then the real realization follows from the change of basis

$$\boldsymbol{x}_m = (\mathbf{T}_m \otimes \cdots \otimes \mathbf{T}_1)\boldsymbol{x}_m^{real}, \qquad m = 1, \dots, M$$

in (20), where $\boldsymbol{x}_m^{real} \in \mathbb{R}^{N_m}$ is the new state vector. Consequently, the residue vector $\boldsymbol{r}_m$ is changed into $\boldsymbol{r}_m^{real} = (\mathbf{T}_m \otimes \cdots \otimes \mathbf{T}_1)^{\mathrm{T}}\boldsymbol{r}_m$. This transformed vector is real provided that the original $\boldsymbol{r}_m$ has the real symmetry property, i.e. generalized residues corresponding to element-wise conjugate sets of poles are conjugate.

Note that the time-domain simulation in the proposed approach can be carried out by first integrating (20a)-(20b) numerically up to $m = M$, and then recovering the output as in (21). The diagonal format of each $\mathbf{A}_m$ implies that the recursive convolution method [60] can be applied iteratively to solve for each $\boldsymbol{x}_m(t)$, a particularly convenient feature from the computational standpoint.

C. A MORE COMPACT REALIZATION

By analyzing (20) in frequency domain, one observes that the input-to-state map $u \to \boldsymbol{x}_m \in \mathbb{C}^{\nu_1 \cdots \nu_m}$ is associated with a symmetric transfer function

$$\boldsymbol{\Xi}_m(s_1, \dots, s_m) = \frac{1}{m!}\sum_{\sigma \in \mathbb{P}_m} \Psi_m(s_{\sigma(1)} + \cdots + s_{\sigma(m)}) \otimes \cdots \otimes \Psi_1(s_{\sigma(1)}) \tag{23}$$

where $\Psi_j(s) = (s\mathbb{I} - \mathbf{A}_j)^{-1}\mathbf{1}_{\nu_j}$, with $j = 1, \dots, m$. Simply put, $\boldsymbol{\Xi}_m$ is just the collection of the basis functions $\varphi^{\mathrm{sym}}_{n_1,\dots,n_m}$ in a column vector. However, by making the Kronecker product structure explicit, one can reinterpret the generalized residues $r_{n_1,\dots,n_m}$ as an m-way tensor $\mathcal{R}$ to recast (19) in tensor format

$$H_m(s_1, \dots, s_m) = \frac{1}{m!}\sum_{\sigma \in \mathbb{P}_m} \mathcal{R} \times_m \Psi_m(s_{\sigma(1)} + \cdots + s_{\sigma(m)})^{\mathrm{T}} \times_{m-1} \cdots \times_1 \Psi_1(s_{\sigma(1)})^{\mathrm{T}}. \tag{24}$$

This rewriting is useful because the tensor identity in [46, Prop. 3.7] allows to go back to a Kronecker product form involving the matricization of $\mathcal{R}$ along any of its modes, thus yielding several equivalent realizations. For instance, using the mode-m matricization, (24) becomes

$$H_m(s_1, \dots, s_m) = \frac{1}{m!}\sum_{\sigma \in \mathbb{P}_m} \Psi_m(s_{\sigma(1)} + \cdots + s_{\sigma(m)})^{\mathrm{T}}\mathcal{R}_{(m)} \cdot \Psi_{m-1}(s_{\sigma(1)} + \cdots + s_{\sigma(m-1)}) \otimes \cdots \otimes \Psi_1(s_{\sigma(1)}). \tag{25}$$

Considering the subsystem of highest modeled degree $m = M$, this simple algebraic manipulation shows that another realization of H_M is obtained by modifying the last output equation in (20c) and the last state equation in (20b) as follows

$$\begin{aligned}\dot{\boldsymbol{x}}_M &= \mathbf{A}_M^{\mathrm{T}}\boldsymbol{x}_M + \mathcal{R}_{(M)}\boldsymbol{x}_{M-1}u \\ y_M &= \mathbf{1}^{\mathrm{T}}\boldsymbol{x}_M\end{aligned} \tag{26}$$

This format is practically more convenient because the size of the last state vector $\boldsymbol{x}_M$ becomes ν_M instead of N_M. Therefore this is the preferred realization used in the subsequent numerical examples shown in Sec. VII.

D. FITTING BY LINEAR REGRESSION

The above developments set the premises for tackling GTF approximation through linear regression. In fact, if the basis poles defining $\varphi^{\mathrm{sym}}_{n_1,\dots,n_m}$ are again regarded as fixed constants (whose estimation is to be discussed later), H_m in eq. (19) has linear dependence upon the to-be-estimated model coefficients $\boldsymbol{r}_m = \mathrm{vec}\{r_{n_1,\dots,n_m}\}$. Using vector notation, $H_m(s_1,\dots,s_m) = \boldsymbol{\Xi}_m^{\mathrm{T}}(s_1,\dots,s_m)\boldsymbol{r}_m$, with $\boldsymbol{\Xi}_m$ from (23),

$$\boldsymbol{\Xi}_m^{\mathrm{T}}(s_1,\dots,s_m) = \left(\varphi^{\mathrm{sym}}_{1,\dots,1}\ \varphi^{\mathrm{sym}}_{2,1,\dots,1}\ \cdots\ \varphi^{\mathrm{sym}}_{\nu_1,\dots,\nu_m}\right), \tag{27}$$

where all entries are evaluated at $(s_1,\dots,s_m)$. Then the standard theory of linear least squares gives the optimal vector $\boldsymbol{r}_m^*$ minimizing the quadratic cost

$$J_m = \sum_{k=1}^{K_m}\left|H(\boldsymbol{s}_m^{(k)}) - \breve{H}_m^{(k)}\right|^2 \tag{28}$$

as $\boldsymbol{r}_m^* = \mathcal{C}_m^{\dagger}\breve{\boldsymbol{H}}_m$, where $\mathcal{C}_m^{\dagger}$ is the Moore-Penrose pseudoinverse of the matrix

$$\mathcal{C}_m = \left(\boldsymbol{\Xi}_m\left(\boldsymbol{s}_m^{(1)}\right)\ \cdots\ \boldsymbol{\Xi}_m\left(\boldsymbol{s}_m^{(K_m)}\right)\right)^{\mathrm{T}} \tag{29}$$

and $\breve{\boldsymbol{H}}_m \in \mathbb{C}^{K_m}$ collects all K_m degree-m GTF data samples. Upon finding the optimal $\boldsymbol{r}_m^*$, the bilinear model is obtained by applying the realization in Definition 1, where the generalized residues appearing in the output equation (21) are chosen as the optimal ones, i.e. $\boldsymbol{r}_m = \boldsymbol{r}_m^*$.

V. POLE ESTIMATION BY RATIONAL FITTING

In this section, we finally address the issue of determining the M pole sets that define the basis functions in (12). In order to review ideas that will be extended for our purposes, let us consider again the degree-1 GTF $H_1(s_1)$, a rational function in the variable s_1. The problem of fitting $H_1(s_1)$ to sampled data $\breve{H}_1^{(k)}$, so that

$$H_1(s_1^{(k)}) \approx \breve{H}_1^{(k)}, \qquad k = 1,\dots,K_1 \tag{30}$$

is known as a standard rational approximation problem that has been investigated thoroughly, with robust and well-developed algorithmic solutions such as the VF algorithm, which is briefly reviewed in the following. VF uses a barycentric representation of the rational model written as a ratio of two functions, namely a numerator $b(s_1)$ and a denominator $a(s_1)$,

$$H_1(s_1) = \frac{b(s_1)}{a(s_1)}. \tag{31}$$

These two auxiliary functions are rational with the same auxiliary poles $\{q_i\}_{i=1}^{\nu_1}$,

$$b(s_1) \triangleq \sum_{i=1}^{\nu_1}\frac{b_i}{s_1 - q_i}, \qquad a(s_1) \triangleq 1 + \sum_{i=1}^{\nu_1}\frac{a_i}{s_1 - q_i} \tag{32}$$

where for simplicity and without loss of generality we assume that $H_1(\infty) = 0$. The VF algorithm performs multiple iterations, each consisting of finding the optimal coefficients a_i, b_i that minimize the cost

$$\tilde{J}_1 = \sum_{k=1}^{K_1}\left|b(s_1^{(k)}) - \breve{H}_1^{(k)}a(s_1^{(k)})\right|^2. \tag{33}$$

The optimal a_i, b_i are obtained by linear least-squares minimization because $\tilde{J}_1$ is a convex quadratic loss function in these unknowns. The rationale behind this choice is that $\tilde{J}_1$ is a *linearized* version of the actual cost J_1 (model-data error) as defined in (28), whence it is obtained by neglecting the common denominator in J_1 after H_1 is replaced with its barycentric form $b(s_1)/a(s_1)$. The choice of auxiliary poles, initially a guess, is iteratively refined using the denominator coefficients according to the update rule $\{q_i\} \leftarrow \mathrm{zeros}\{a(s)\}$, that takes the new auxiliary poles for the next iteration to be the denominator zeros. At convergence, the denominator coefficients a_i tend to zero, $a(s) \to 1$, and the set of auxiliary poles converge to the model poles. Hence, a useful feature of VF is that the pole relocation phase yields a set of model poles that can be used to define a pole-residue expansion like (19).

We see that using VF to find a rational approximant of the degree-1 samples $\breve{H}_1^{(k)}$ automatically yields the first set of model poles $\{p_n^{(1)}\}$, which coincide with the auxiliary poles $\{q_i\}$ upon convergence. In order to employ the same pole estimation technique for higher degree GTFs, let us go back to (25) to observe that the GTF model H_m can be factored as

$$H_m(s_1,\dots,s_m) = \boldsymbol{F}_m(s_1+\cdots+s_m)\boldsymbol{Q}_m(s_1,\dots,s_m) \tag{34}$$

with $\boldsymbol{F}_m(s_1+\cdots+s_m) = \Psi_m(s_{\sigma(1)}+\cdots+s_{\sigma(m)})^{\mathrm{T}}\mathcal{R}_{(m)}$. Note that $\boldsymbol{F}_m$ is a $1\times N_{m-1}$ rational function with poles at $\{p_n^{(m)}\}$, univariate in the argument $s_1+\cdots+s_m$ and it factors out of the symmetric sum in (25) because its argument $(s_{\sigma(1)}+\cdots+s_{\sigma(m)})$ is not affected by the permutation σ.

As for the vector $\boldsymbol{Q}_m$ in (34), it contains the symmetrizations of the basis functions $\varphi_{n_1,\dots,n_{m-1}}$, as follows

$$\begin{aligned}\boldsymbol{Q}_m(s_1,\dots,s_m) = \frac{1}{m!}\sum_{\sigma\in\mathbb{P}_m}\Psi_{m-1}(s_{\sigma(1)}+\cdots+s_{\sigma(m-1)})\otimes \\ \cdots\otimes\Psi_1(s_{\sigma(1)})\end{aligned} \tag{35}$$

The key observation on (34) is that the vector $\boldsymbol{Q}_m$ is entirely determined by the pole sets up to degree $m-1$, and the only quantity depending on the degree-m pole set $\{p_n^{(m)}\}$ is $\boldsymbol{F}_m$.

Therefore, we can take a greedy approach towards estimating the GTF poles where, after the first $m-1$ sets are available and fixed, the m-th set is chosen to coincide with the poles of $\boldsymbol{F}_m$, which are found by rational fitting. In fact, we can address the problem of finding poles and residues of $\boldsymbol{F}_m$ to minimize the residual of the approximation

$$\boldsymbol{F}_m(s_1^{(k)}+\cdots+s_m^{(k)})\underbrace{\boldsymbol{Q}_m(s_1^{(k)},\ldots,s_m^{(k)})}_{\triangleq \boldsymbol{Q}_m^{(k)}} \approx \breve{H}_m^{(k)} \tag{36}$$

for $k=1,\ldots,K_m$ again through VF, precisely because the unknown is the univariate rational function $\boldsymbol{F}_m(s)$ evaluated at $s_1^{(k)}+\cdots+s_m^{(k)}$, and $\boldsymbol{Q}_m^{(k)}$ are just known vectors. The only aspect of (36) that prevents direct application of VF in its standard form is the presence of the factor $\boldsymbol{Q}_m^{(k)}$, absent in (30), which is the point of departure in the standard VF formulation. However, it is easy to modify VF to incorporate an arbitrary but known weighting factor. In fact, using the barycentric form of $\boldsymbol{F}_m(s)=\boldsymbol{b}(s)/a(s)$, where $\boldsymbol{b}(s)$ is a vector-valued numerator function, a linearized version of the loss function associated with (36) can still be obtained in the same manner as (33), and it reads

$$\tilde{J}_{\mathrm{m}}^{\mathrm{weight}}=\sum_{k=1}^{K_m}\left|\boldsymbol{b}\left(s_1^{(k)}+\cdots+s_m^{(k)}\right)\boldsymbol{Q}_m^{(k)} -a\left(s_1^{(k)}+\cdots+s_m^{(k)}\right)\breve{H}_m^{(k)}\right|^2 \tag{37}$$

Eq. (37) gives the cost function to be minimized during the pole relocation to find the m-th pole set, using the same update rule as in standard VF. This extension of VF has been developed by the authors for an originally different purpose in [61] and [62], where a more detailed account is available.

Note that this pole-finding procedure based on a sequence of weighted VF steps is enabled by the factorization (34), which is also leveraged in [57]. However, in [57], both poles and residues determining the degree-m subsystem are estimated through weighted VF, and the approximation result for degree m is influenced by the residues previously estimated at lower degrees. This can produce *error accumulation* going from $m=1$ to higher m. In this work, weighted VF is only used for estimating the basis poles used to define the φ^{sym}'s. Hence, the vectors $\boldsymbol{Q}_m$ are different and independent of the lower-degree generalized residues, which are estimated independently of each other (see Sec. IV-D).

To sum up, we report a pseudo-code of the GTF fitting method in Algorithm 1, which combines the pole-finding method just described with residue estimation from Sec. IV-D.

The flowchart in Fig. 3 represents the proposed algorithm along with the context where it is meant to be used. After a first step to obtain the data samples using one of the existing methods [36], [38], [39], [40], [41], the proposed fitting approach summarized in Algorithm 1 can be applied. Note that the electrical and geometric parameters of the device to be modeled belong to the original simulation model or physical device used for GTF extraction (first step in Fig. 3) and are not directly used in subsequent modeling steps because the proposed method is non-intrusive. Fig. 3 contains references to the well-known VF algorithm, whose algorithmic steps are described in [11], and to weighted VF which is presented in [61].

Algorithm 1 Approximation of GTFs

Require: M, K_m, ν_m and the data set $\mathcal{D}$ as defined in (7)
  Find poles $\{p_n^{(1)}\}_{n=1}^{\nu_1}$ using VF iteration on $\breve{H}_1(s_1^{(k)})$
  for $m=2,\ldots,M$ **do**
    Compute vectors $\boldsymbol{Q}_m^{(k)}$ for $k=1,\ldots,K_m$ as in (35)
    Find poles $\{p_n\}_{n=1}^{\nu_m}$ using weighted VF [61], see (36)
    Compute generalized residues $\boldsymbol{r}_m^*$, see Sec. IV-D
  end for

VI. ERROR ANALYSIS AND ORDER SELECTION

The series representation (1) allows to decompose the response of the *true* system $\breve{y}(t)$ as a sum of nonlinear contributions up to degree M, denoted as $\breve{y}_M(t)$, and the higher-degree terms $\breve{y}_{hd}(t)$ whose behavior is not taken into account in the modeling process. As for the model response as $y(t)$, it only contains nonlinear terms up to degree M by construction. A similar decomposition is useful in the error signal $e(t)=\breve{y}(t)-y(t)$ which, for a given input signal $u(t)$, reads

$$e(t)=\sum_{m=1}^{M}\int_0^t e_m(\tau_1,\ldots,\tau_m)u(t-\tau_1)\cdots u(t-\tau_m)\,\mathrm{d}\boldsymbol{\tau}+ +\sum_{m=M+1}^{\infty}\int_0^t \breve{h}_m(\tau_1,\ldots,\tau_m)u(t-\tau_1)\cdots u(t-\tau_m)\,\mathrm{d}\boldsymbol{\tau} \tag{38}$$

In (38), the second sum is the unmodeled nonlinearity $\breve{y}_{hd}$, that depends on the exact kernels $\breve{h}_m$ of the underlying system. On the other hand, $e_m=\breve{h}_m-h_m$ is the modeling error in the first M kernels, arising from the residual fitting error in (28).

The connection between the error norm $\|e(t)\|_\infty$ resulting from a given input energy $\|u(t)\|_2$ is made through the gain bound function [48]. Applying standard L_2 inequalities to (38) yields

$$|e(t)|\le\sum_{m=1}^{M}\|e_m\|_{\mathcal{H}_2}\|u\|_2^m+\sum_{m=M+1}^{\infty}\left\|\breve{h}_m\right\|_{\mathcal{H}_2}\|u\|_2^m \tag{39}$$

where the $\mathcal{H}_2$ system norm is as defined in [31]. The $\mathcal{H}_2$ system norm also admits a frequency-domain representation through Plancherel's theorem

$$\|e_m\|_{\mathcal{H}_2}^2=\frac{1}{(2\pi)^m}\int_{-\infty}^{+\infty}\left|\breve{H}_m-H_m\right|^2\mathrm{d}\omega_1\cdots\mathrm{d}\omega_m \tag{40}$$

where the functions H_m and $\breve{H}_m$ in the integral are evaluated at $(\mathrm{j}\omega_1,\ldots,\mathrm{j}\omega_m)$. We see that, as in the linear scenario, eq. (39) and (40) motivate using the cost function J_m in (28), as it can be seen as a discretized surrogate of the integral expression for $\|e_m\|_{\mathcal{H}_2}^2$ in (40), that explicitly appears in the bound (39).

The choice of the maximum degree of modeled nonlinearity M is mainly driven by considerations regarding how much the behavior of the system under analysis deviates from its small-signal first-order approximation for a given input amplitude. It might be determined using specific methods to find the largest significant degree such as [63], or be limited in practice by the effort required to sample GTFs in a high-dimensional space. On this matter, we emphasize that, although this curse of dimensionality is somewhat intrinsic to the Volterra series, to the extent that it has been recognized and addressed in several works [64], [65], the proposed approximation algorithm is compatible with efficient sparse sampling schemes as it does not require that the $\boldsymbol{s}_m^{(k)}$ lie on a regular grid, unlike the method [54].

Regardless of how M is chosen, the error due to the unmodeled nonlinearity in (39) represents a fixed residual error. On the other hand, the fitting errors e_m can be controlled by augmenting the pole sets (i.e. increasing the number of poles ν_m). As in standard VF, the parameter ν_m is given in advance in Algorithm 1. In case an a priori estimate of ν_m is not available, it is also possible to equip the VF iteration with automated order selection, as discussed in [66], by monitoring the error as the number of poles is increased until a stopping tolerance is reached. In the present context, it should be taken into account that, when the fitting error is low enough, the inaccuracy in system response will be dominated by $\breve{y}_{hd}(t)$, and there is no point in further increasing the number of poles.

VII. NUMERICAL EXAMPLES

This section reports three applications of the GTF fitting method to nonlinear circuits of practical relevance. In all cases, we illustrate the details of GTF fitting, including accuracy in the frequency domain, as well as the time-domain response. In each section, the link between the general symbols used until now – e.g. $u(t)$,$y(t)$ – and the electrical quantities (currents or voltages) pertaining to each example will be provided.

A. PRELIMINARY: GTF EXTRACTION

The preliminary step in the typical workflow of Fig. 3 is the extraction of GTF samples of the weakly nonlinear circuit under analysis. Measurement of Volterra transfer functions using a Vector Analyzer has been addressed in [41]. Extraction from measured or simulated X-parameters is discussed in [36]. In our experiments, the GTF samples were extracted through the classic harmonic probing method [38], [39]. In practice, we used harmonic balance (HB) analysis [67] performed via commercial solvers, which solves the circuit response to a quasi-periodic multi-tone input in the frequency domain. Starting from the Fourier coefficients of the response resulting from HB for several values of the input amplitude, the GTF can be extracted by solving an overdetermined linear system (as detailed in [39]). More advanced probing methods have been developed, for example, using Gaussian inputs [68], or modulated signals [40].

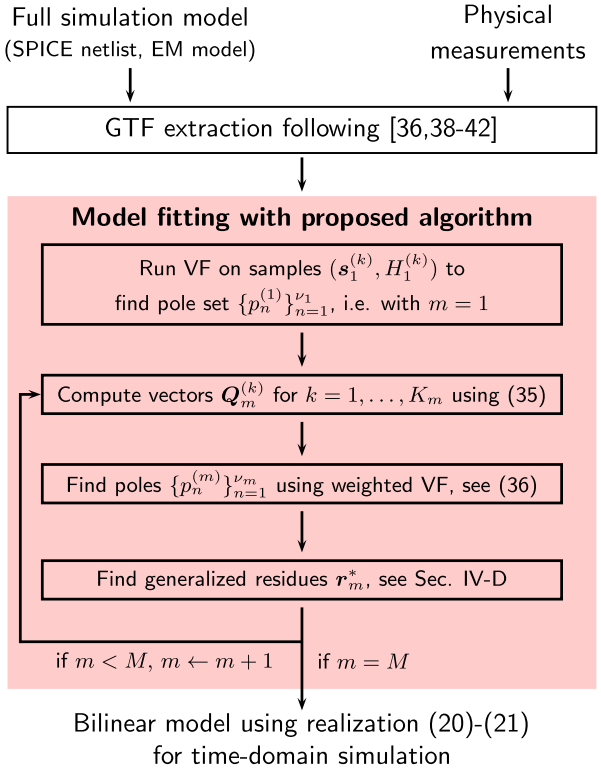


FIGURE 3. Typical workflow of applicability of the GTF fitting algorithm.

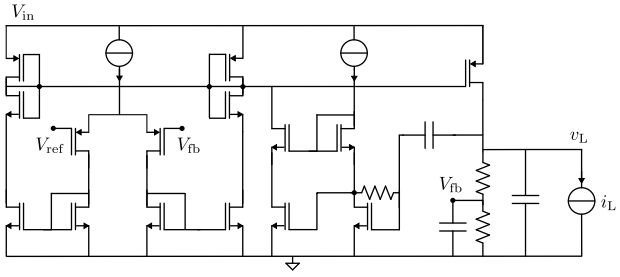


FIGURE 4. Circuit diagram of the LDO example of Sec. VII-B, originally presented in [69]. The load is an ideal current source with value $i_L(t)$, the load voltage is $v_L(t)$.

B. LOW DROPOUT REGULATOR

This first test case (proof of concept) is an LDO regulator, typically employed to provide a regulated supply voltage to other circuit blocks. In Fig. 4, we report the circuit diagram as originally presented in [69] (see original source for additional details such as component values and topology description). Here, we operate the LDO with $V_{\rm in} = 3$ V supply voltage and a load bias current $I_L = 20$ mA.

To build a model around this bias point ($u_0 = I_L$), the output voltage response to load current variations was measured in the frequency domain by HB analysis with up

to three tones. This is because our purpose is to model the transfer function from the load current i_L, which coincides with the input $u(t)$, to the output voltage v_L, i.e. the system output $y(t)$.

In single-tone experiments, the maximum frequency is set to 50 MHz and $K_1 = 60$ samples of $\check{H}_1^{(k)}(\mathrm{j}2\pi f_1)$ are collected. These represent samples of the small-signal output impedance around the prescribed bias point. In two-tone experiments, we measured the response at points (f_1, f_2) on a square grid. Each of f_1, f_2 is swept on 140 logarithmically-spaced points ($\approx$ 30 points/decade) in the interval $\in [500\,\mathrm{Hz}, 20\,\mathrm{MHz}]$. This results in $K_2 = 2 \times 140^2$ samples of the 2^{nd}-degree GTF $\check{H}_2^{(k)}(\mathrm{j}2\pi f_1, \pm\mathrm{j}2\pi f_2)$. Similarly, three-tones simulations were conducted up to 20 MHz to obtain a total of $K_3 = 4 \times 140^2$ samples of the type $\check{H}_3^{(k)}(\mathrm{j}2\pi f_1, \pm\mathrm{j}2\pi f_2, \pm\mathrm{j}2\pi f_3)$. These were used to construct a bilinear model of maximum degree $M = 3$ as described above with $\nu_1 = 6$, $\nu_2 = 7$, and $\nu_3 = 7$ poles. As typically done with the classical VF algorithm, the number of poles was increased until a satisfactory model-data fit was obtained.

In the frequency domain, a model-data comparison for GTFs of degree one and two is reported in Fig. 5, whereas a subset of the degree-3 responses (i.e. the H_3 GTF) is displayed in Fig. 6.

The voltage response is comprised of an affine term corresponding to the DC voltage at the bias point $V_{L,dc} = y_0 = 2.7107\,\mathrm{V}$, in addition to the bilinear model response to the shifted input $u(t) = i_L(t) - I_L$.

For time-domain validation, we first consider the case where $u(t)$ is a sequence of current steps between 0 to $\pm 10\,\mathrm{mA}$, with rise time $t_r = 0.2\,\mu\mathrm{s}$. In this case, the reference circuit is solved in the time domain using HSPICE in the interval $t \in [0, 6\mu\mathrm{s}]$ and compared with the bilinear model solved in MATLAB. The solution is discretized in 5×10^4 timesteps and found through recursive convolutions applied to the bilinear model (20).

A comparison of the modeled load voltage response with the reference solution is shown in Fig. 7(top). Starting from the degree-3 model, we also truncated it to lower degrees of nonlinearity ($M = 1, 2$) and computed the time-domain responses to evaluate the contribution of higher-order components. Figure 7 (top panel) confirms that the degree-3 model is more accurate than a simple linear (small-signal) approximation. More quantitatively, the instantaneous time-domain error with different model degrees is reported in the middle panel of Fig. 7. A summary of the errors in terms of peak and RMS (root-mean-square) values is also provided in the bottom panel.

To illustrate the different sources of errors discussed in Sec. VI, we built several models with different numbers of poles $\nu = \nu_1 = \nu_2 = \nu_3$. Performing the same simulation with the trapezoidal pulse input for each of these models, we obtain the RMS errors reported in Fig. 8. We see that the degree-2 model (green line) stagnates after $\nu = 4$, suggesting that the error due to the truncation of higher-degree components is dominant. In fact, the error

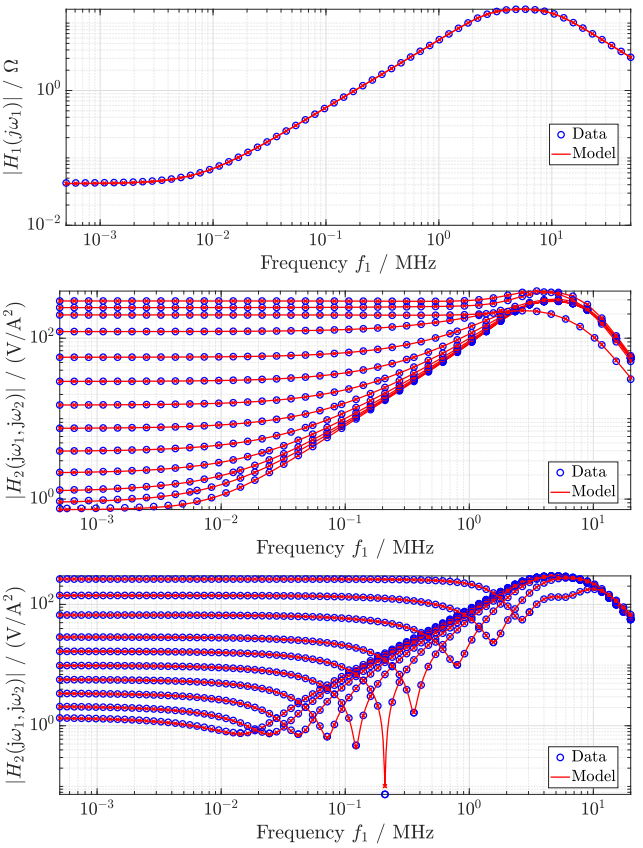


FIGURE 5. Data-model comparison of the frequency-domain GTFs of the LDO example in Sec. VII-B. Top panel: degree-1 GTF. Middle panel: degree-2 GTF as a function of f_1 for several values of f_2. Bottom panel: same as the middle panel, containing samples of H_2 in the second quadrant, i.e. with $f_1 > 0$, $f_2 < 0$.

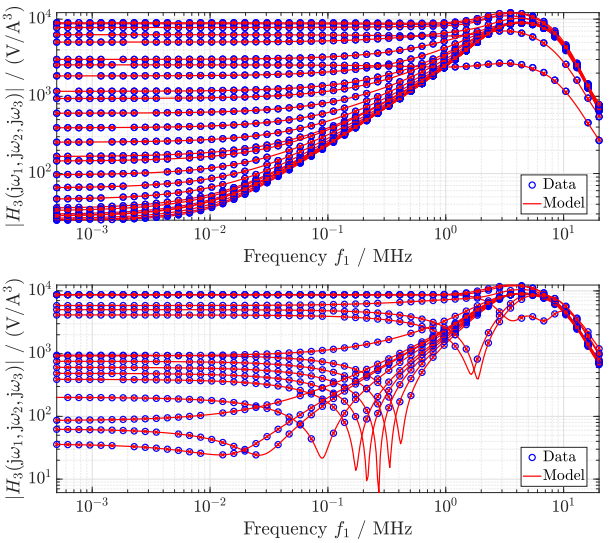


FIGURE 6. Data-model comparison of degree-3 GTFs regarding the LDO example in Sec. VII-B. For several combinations (f_2, f_3), $|H_3|$ is displayed as a function of f_1. The top panel shows curves corresponding to both $f_2, f_3 > 0$, while the bottom one shows slices where at least one among f_2, f_3 is negative.

becomes lower and continues to decrease with increasing ν if the model degree is increased to $M = 3$ (red line).

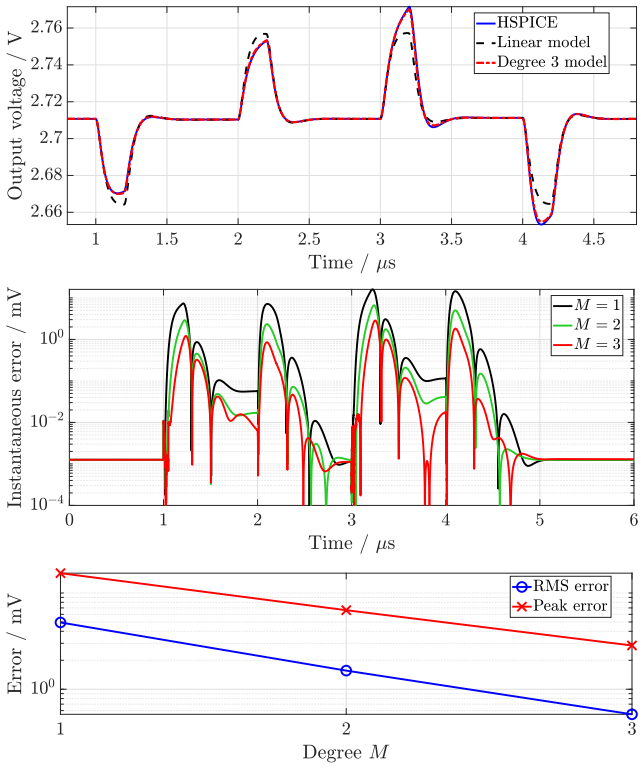


FIGURE 7. Time-domain validation of the LDO model in Sec. VII-B using a sequence of step signals. Top panel: transient response. Middle panel: instantaneous error with respect to time for several values of maximum degree M. Bottom panel: root-mean-square (RMS) error and peak error (over the entire simulation interval) versus model degree M.

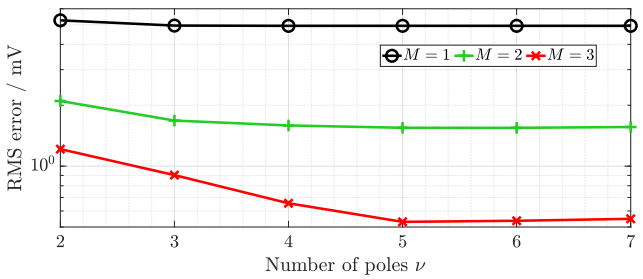


FIGURE 8. RMS error trend (corresponding to the transient simulation in Fig. 7) as the number of poles $\nu = \nu_1 = \nu_2 = \nu_3$ is increased, for different model degrees up to $M = 3$.

A second test was carried out with a different input signal $u(t)$, obtained by low-pass filtering a 10μs-long white noise signal through a single-pole filter with -3-dB frequency at 0.2 MHz. The noise power was adjusted to obtain a 10 mA peak amplitude after filtering. The model response is compared with the HSPICE reference solution in Fig. 9, where a subinterval of the simulation is reported so as to show the differences between the linear model response (degree $M = 1$) and the nonlinear model with $M = 3$. The dependence of peak and RMS errors (in the entire simulation interval) versus M are also reported, to show that a higher degree of modeled nonlinearity yields increased accuracy in the time domain.

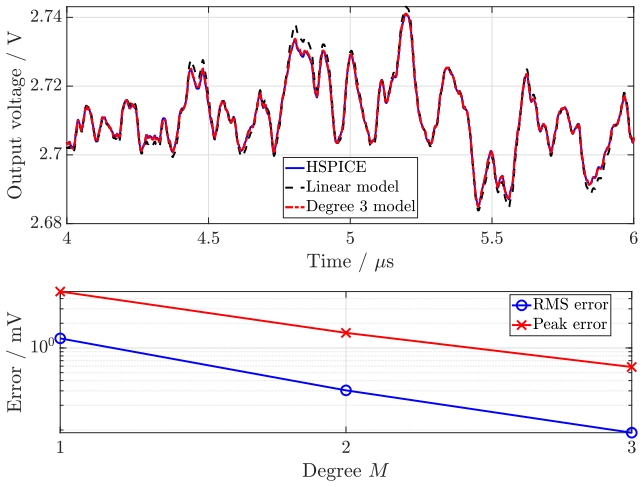


FIGURE 9. Time-domain validation of the LDO example from Sec. VII-B. Top panel: voltage responses in the sub-interval [4, 6] μs. Bottom panel: RMS and peak errors over time, as a function of the model degree M.

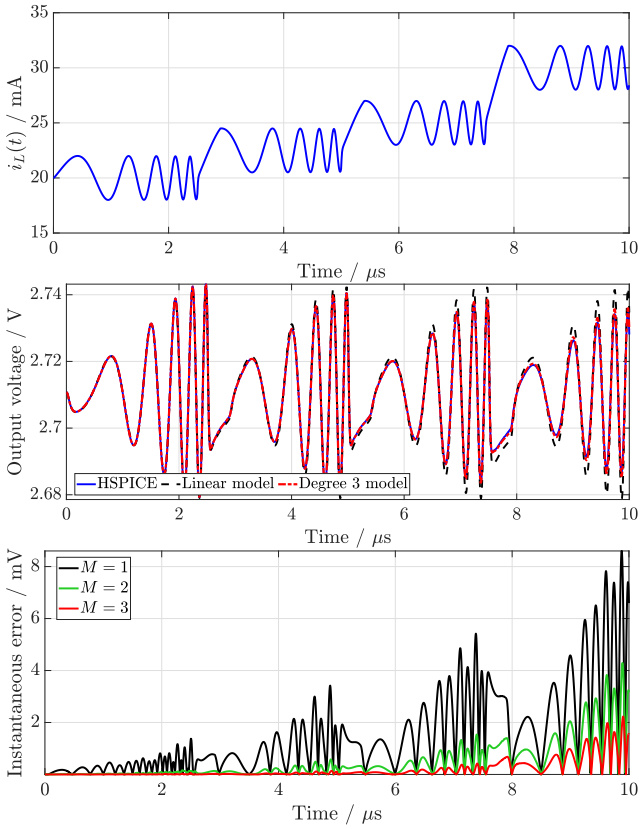


FIGURE 10. Analysis of varying load conditions for the LDO example in Sec. VII-B. Top: load current $i_L(t)$, i.e. input signal. Middle: transient response of HSPICE (reference) and proposed model (with $M = 1$ or 3). Bottom: instantaneous error (proposed vs. HSPICE) for several model degrees M.

As discussed in Sec. II-B, the Volterra series underpinning our results is an expansion around some operating point u_0, which provides higher-order nonlinear corrections to a simple small-signal linearized model (degree one). To analyze the

consequence of varying loading conditions that drive the system away from the expansion point (in this example, the bias current $I_L = 20\,\text{mA}$), we report the results of a transient simulation using the load current $i_L(t)$ shown in Fig. 10(top). It consists of a repeated chirp signal with frequency spanning the interval [0.5, 5] MHz oscillating around different bias points. As the operating point moves away from the initial value $I_L = u_0 = 20\,\text{mA}$ to 22.5, 25 and 30 mA, the error of the linear model (degree $M = 1$) increases visibly as shown in Fig. 10(bottom). Nonlinear models of degree $M = 2$ and $M = 3$ provide a lower error because they capture the nonlinear behavior that appears in large-signal conditions.

C. POST-LAYOUT LOW DROPOUT REGULATOR

This section describes a large-scale example, namely the low-quiescent-current LDO design originally proposed in [70] for low-power applications. We use the implementation described in [1], with a layout based on a 40 nm CMOS technology process in Cadence Virtuoso [44]. Post-layout parasitic extraction produces a network with more than 4×10^4 nodes, 2×10^4 capacitances, and 6×10^4 resistances. Consequently, transient analyses of the resulting network take up to several minutes, even using a commercial solver like Cadence SPECTRE.

The input voltage was set to $V_{DD} = 1\,\text{V}$ and the output voltage is regulated to a reference value $V_{ref} = 0.6\,\text{V}$. We built a nonlinear model of the output impedance, i.e. the load voltage response to a load current excitation, around a nominal operating point corresponding to a load current $I_L = 50\,\mu\text{A}$. As a starting point, GTF samples were collected up to degree $M = 3$ in the frequency range [1 kHz, 1 MHz]. To extract higher-order GTFs, we resorted to HB analyses with at most three tones and with amplitude in the range $[1, 3]\,\mu\text{A}$. These produced respectively $K_2 = 1331$ and $K_3 = 12799$ samples of $\breve{H}_2$ and $\breve{H}_3$. The proposed algorithm was run with $\nu_1 = 7$, $\nu_2 = 12$, $\nu_3 = 12$ poles to build the nonlinear rational model.

A frequency-domain comparison between the data and the degree-2 model is reported in Fig. 11. Each curve is a slice of the graph of the bivariate function $H_2(\text{j}2\pi f_1, \text{j}2\pi f_2)$ along f_1 and with fixed f_2. The top panel shows several curves corresponding to positive values of f_2, while the bottom one shows negative values. Fig. 12 shows similar results for one-dimensional slices of the graph of H_3.

In the time domain, we provide validation against the reference solution computed in SPECTRE using two different test signals. The first one is a 40-μs long chirp waveform whose instantaneous frequency $f(t)$ sweeps logarithmically from 10 kHz to 1 MHz. Its amplitude is increased linearly from $10\,\mu\text{A}$ to $20\mu\text{A}$ in the interval $[0, 10]\,\mu\text{s}$ and stays constant afterwards. Fig. 13 (top) reports the load voltage waveform computed by the reference SPECTRE, which is compared with the proposed model of degree three. The linear response, corresponding to the proposed model truncated at degree one, is also depicted to show the benefit of

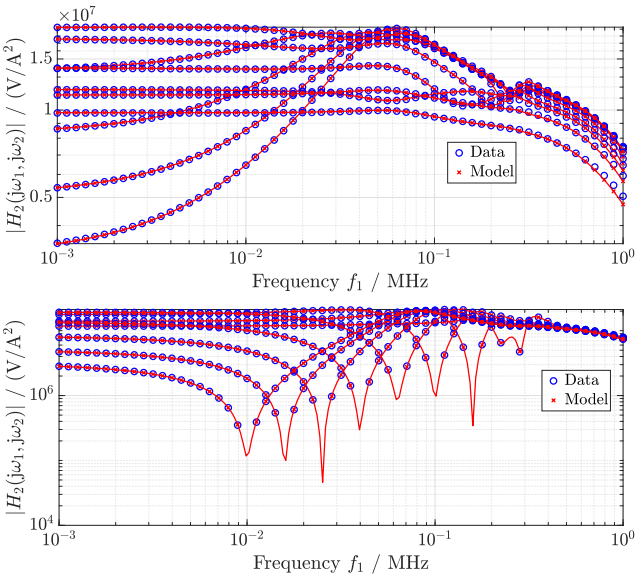


FIGURE 11. Data-model comparison of the degree-2 transfer function from the post-layout LDO example in Sec. VII-C. Top: slices corresponding to several fixed values of $f_2 > 0$. Bottom: slices corresponding to $f_2 < 0$.

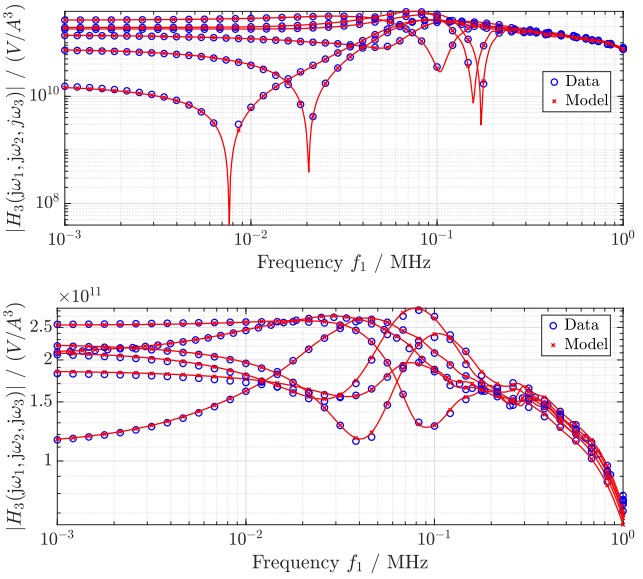


FIGURE 12. Data-model comparison of H_3 from the post-layout LDO example in Sec. VII-C.

adding the nonlinear correction. Fig. 13 (bottom) reports the instantaneous error for this transient simulation across all three degrees $M = 1, 2, 3$.

The second test signal consists of a trapezoidal pulse of duration $20\,\mu\text{s}$, amplitude $25\,\mu\text{A}$ and 10 ns rise/fall time. The load voltage response is reported in Fig. 14(top). Note that nonlinear models of increasing degree are closer to the reference solution computed in SPECTRE, as confirmed by the error trends (peak and RMS error) reported in the bottom panel. The overall simulation up to $40\,\mu\text{s}$ is solved in 4×10^3 timesteps in both SPECTRE and MATLAB. While

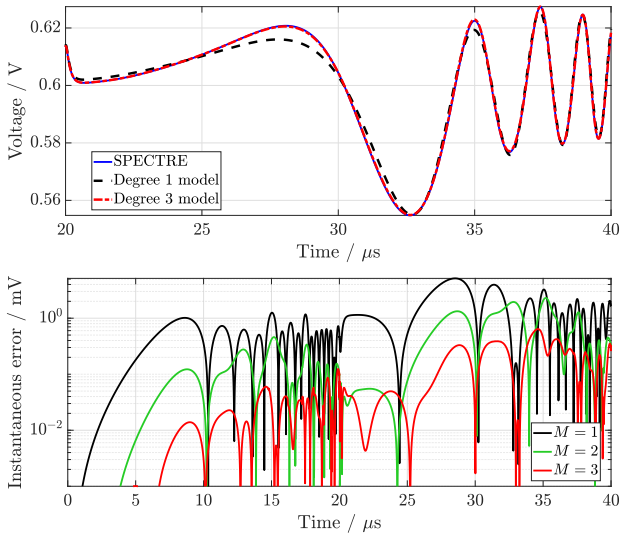


FIGURE 13. Time-domain analysis of the post-layout LDO of Sec. VII-C using a chirp signal input. Top: reference and modeled load voltage responses. Bottom: instantaneous error for each model degree M.

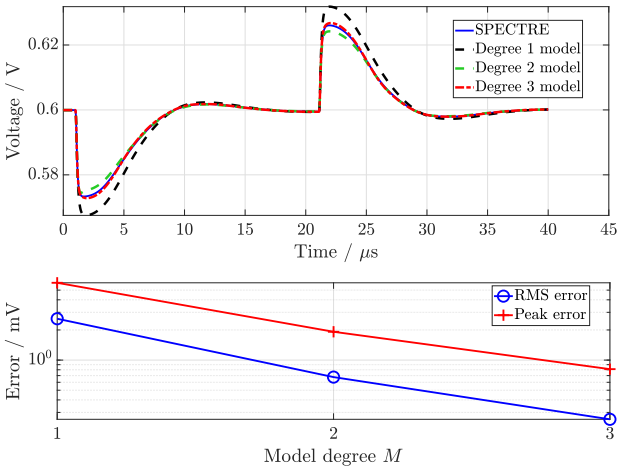


FIGURE 14. Step response of the post-layout LDO of Sec. VII-C. Top: reference and modeled load voltage responses. Bottom: peak and RMS error values as the model degree M is increased from 1 to 3.

the former takes 118 seconds on a 2.3-GHz Linux server, the proposed model of degree 3 is solved in under 0.1 s on a 1.7-GHz laptop. Hence, the compact model provides a computational speedup in excess of 1000×.

Owing to the large-scale nature of the original model, the preliminary GTF extraction task is computationally demanding. In our experiments with the basic harmonic probing method, it took just under 2 hours to extract GTF samples of degree two. For the third degree, the HB sweep required about 36 hours to complete. Compared to this data collection phase, the proposed rational fitting algorithm represents a negligible added computational cost as it runs in 1.3 minutes. It should be considered that data collection and recursive rational fitting constitute a one-time cost that is spent for model construction. Once the model is available, a major speedup in repeated transient analyses is granted.

D. LOW-NOISE AMPLIFIER

This section is concerned with a different example consisting in an LNA design, taken from the *X-parameters* examples available with Keysight ADS simulator [45]. The LNA is designed to operate at around 750 MHz, providing a 15 dB gain. A high-level view of this example is provided in Fig. 15, where the amplifier block hides a transistor-level model including distributed elements. Note that knowledge of the internal transistor models is not required to apply the proposed black-box algorithm, which is entirely independent of such internal details.

In this case, we extracted GTF samples up to $M = 3$ using HB simulations up to 3 GHz. The model was built using $K_1 = 116$, $K_2 = 3416$, $K_3 = 36121$ samples and $\nu_1 = 11$, $\nu_2 = 11$, $\nu_3 = 9$ poles. Using relative error weighting, the data was fitted with the results reported in Fig. 16. In the time domain, the model was compared to the simulation in

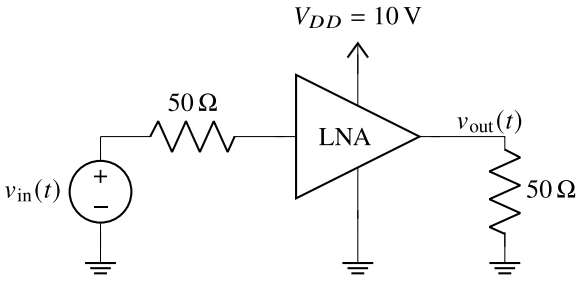


FIGURE 15. High-level view of the RF LNA example in Sec. VII-D.

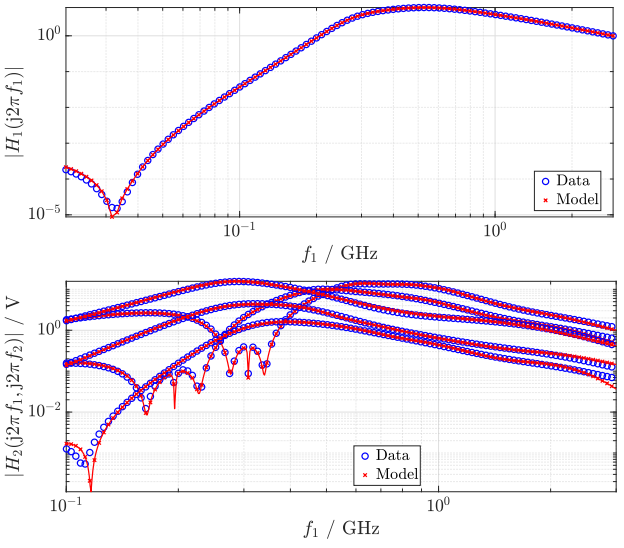


FIGURE 16. Data-model comparison of the first two GTFs from the example in Sec. VII-D. Top panel: degree-1 GTF, i.e. small-signal transfer function. Bottom panel: degree-2 GTF in magnitude versus the first variable f_1, for several values of f_2.

ADS. The input signal consists of a 750 MHz carrier with On-Off Keying modulation at 40 MHz. The carrier amplitude is 100 mV, meaning that the available power level is −16 dBm. The simulation length is 2 μs and the number of simulation

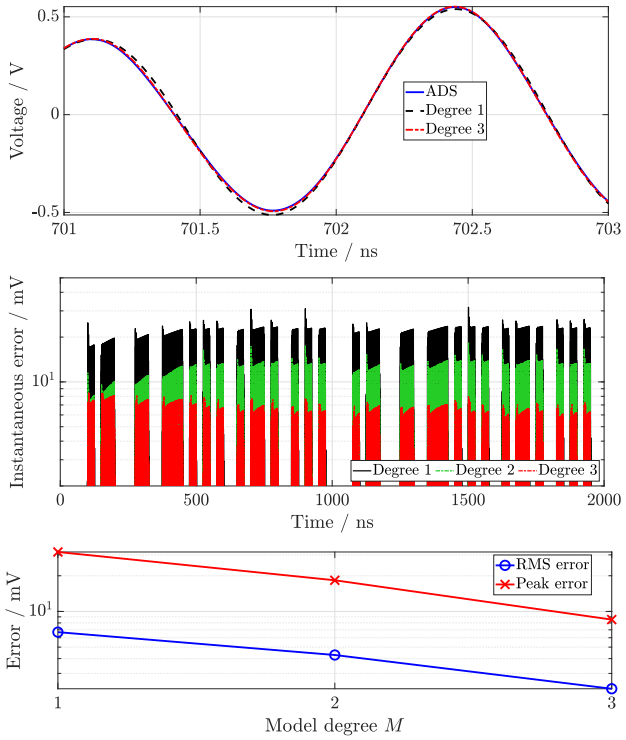


FIGURE 17. Time-domain validation of the example in Sec. VII-D using reference solution from ADS. Top: comparison between responses in a subinterval. Middle: error along the entire simulation, for all degrees up to three. Bottom: RMS and peak errors versus model degree M.

timesteps is $2 \cdot 10^5$ in both ADS and the proposed approach. The amplifier output waveforms (in a short subinterval of the overall simulation) and the error between the reference simulator and models of different degrees are reported in Fig. 17. Again, we observe a reduction in the time-domain error (both RMS and peak values) as a higher degree of nonlinearity is considered. This suggests that the inaccuracy of the degree-1 model is mainly due to a deviation from the small-signal regime at this power level, resulting in a weakly nonlinear behavior that is correctly represented by adding higher-order responses.

Although the original simulation model is relatively small-scale, a speedup is already observed because the described transient analysis is completed in 3.6 s using the black-box model, while Keysight ADS takes 20 s to solve the original system for the same number of timesteps. Regarding the modeling phase, the total time required for fitting (pole and residue identification) is six minutes. The most time-consuming part is the preliminary GTF extraction via harmonic balance. In our experiments, the HB sweep for extracting degree-2 samples $\breve{H}_2^{(k)}$ took 5.6 minutes, while the extraction $\breve{H}_3^{(k)}$ required 65 minutes using Keysight ADS on a 3.3-GHz computer.

VIII. CONCLUSION

This paper described a macromodeling algorithm based on the rational fitting of Volterra transfer functions to build behavioral models of weakly nonlinear circuits. While the modeling workflow remains entirely in the frequency domain, the method produces a bilinear state-space representation that can be simulated in the time domain. Moreover, the method developed herein only requires input/output observations of the steady-state responses of the device to be modeled, without knowledge of its governing equations. Using test cases of practical relevance, considerable speedup is observed compared to the original simulation model.

REFERENCES

[1] T. Bradde, S. Grivet-Talocia, P. Toledo, A. V. Proskurnikov, A. Zanco, G. C. Calafiore, and P. Crovetti, "Fast simulation of analog circuit blocks under nonstationary operating conditions," *IEEE Trans. Compon., Packag., Manuf. Technol.*, vol. 11, no. 9, pp. 1355–1368, Sep. 2021.

[2] P. Toledo, R. Rubino, F. Musolino, and P. Crovetti, "Re-thinking analog integrated circuits in digital terms: A new design concept for the IoT era," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 68, no. 3, pp. 816–822, Mar. 2021.

[3] A. Odabasioglu, M. Celik, and L. T. Pileggi, "PRIMA: Passive reduced-order interconnect macromodeling algorithm," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 17, no. 8, pp. 645–654, Aug. 1998.

[4] E. J. Grimme, "Krylov projection methods for model reduction," Ph.D. thesis, Graduate College, Univ. Illinois Urbana-Champaign, Champaign, IL, USA, 1997.

[5] B. N. Bond and L. Daniel, "A piecewise-linear moment-matching approach to parameterized model-order reduction for highly nonlinear systems," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 26, no. 12, pp. 2116–2129, Dec. 2007.

[6] C. Gu, "QLMOR: A projection-based nonlinear model order reduction approach using quadratic-linear representation of nonlinear systems," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 30, no. 9, pp. 1307–1320, Sep. 2011.

[7] L. Ljung, *System Identification: Theory for the User*. Upper Saddle River, NJ, USA: Prentice-Hall, 1999.

[8] B. Gustavsen and A. Semlyen, "Rational approximation of frequency domain responses by vector fitting," *IEEE Trans. Power Del.*, vol. 14, no. 3, pp. 1052–1061, Jul. 1999.

[9] B. Gustavsen, "A hybrid measurement approach for wideband characterization and modeling of power transformers," *IEEE Trans. Power Del.*, vol. 25, no. 3, pp. 1932–1939, Jul. 2010.

[10] E. Fevola, T. Bradde, P. Triverio, and S. Grivet-Talocia, "A vector fitting approach for the automated estimation of lumped boundary conditions of 1D circulation models," *Cardiovascular Eng. Technol.*, vol. 14, no. 4, pp. 505–525, Aug. 2023.

[11] S. Grivet-Talocia and B. Gustavsen, *Passive Macromodeling: Theory and Applications*. New York, NY, USA: Wiley, 2015.

[12] A. Ramirez, B. Gustavsen, and I. Ramirez, "Rational approximation of nonlinear systems via NL-VF," *IEEE Trans. Power Del.*, vol. 38, no. 3, pp. 1918–1926, Jun. 2023.

[13] A. Ramirez and B. Gustavsen, "Relaxed nonlinear vector fitting for calculation of rational approximation of systems defined by input/output responses in the frequency domain," *IEEE Trans. Power Del.*, vol. 39, no. 5, pp. 2965–2972, Oct. 2024.

[14] E.-W. Bai and F. Giri, "Introduction to block-oriented nonlinear systems," in *Block-Oriented Nonlinear System Identification*. London, U.K.: Springer, 2010, pp. 3–11.

[15] R. K. Pearson and M. Pottmann, "Gray-box identification of block-oriented nonlinear models," *J. Process Control*, vol. 10, no. 4, pp. 301–315, Aug. 2000.

[16] T. A. Johansen and B. Foss, "Constructing NARMAX models using ARMAX models," *Int. J. Control*, vol. 58, no. 5, pp. 1125–1153, Nov. 1993.

[17] W. Rugh, *Nonlinear System Theory: The Volterra/Wiener Approach* (Johns Hopkins Series in Information Sciences and Systems). Baltimore, MD, USA: The Johns Hopkins Univ. Press, 1981.

[18] C. Cheng, Z. Peng, W. Zhang, and G. Meng, "Volterra-series-based nonlinear system modeling and its engineering applications: A state-of-the-art review," *Mech. Syst. Signal Process.*, vol. 87, pp. 340–364, Mar. 2017.

[19] M. Schetzen, *The Volterra and Wiener Theories of Nonlinear Systems*. Melbourne, FL, USA: Krieger, 2006.

[20] Z. Naghibi, S. A. Sadrossadat, and S. Safari, "Time-domain modeling of nonlinear circuits using deep recurrent neural network technique," *AEU-Int. J. Electron. Commun.*, vol. 100, pp. 66–74, Feb. 2019.

[21] P. Bevanda, S. Sosnowski, and S. Hirche, "Koopman operator dynamical models: Learning, analysis and control," *Annu. Rev. Control*, vol. 52, pp. 197–212, Jan. 2021.

[22] T. Wang and T. Brazil, "Volterra-mapping-based behavioral modeling of nonlinear circuits and systems for high frequencies," *IEEE Trans. Microw. Theory Techn.*, vol. 51, no. 5, pp. 1433–1440, May 2003.

[23] A. Zhu and T. J. Brazil, "An overview of Volterra series based behavioral modeling of RF/microwave power amplifiers," in *Proc. IEEE Annu. Wireless Microw. Technol. Conf.*, Dec. 2006, pp. 1–5.

[24] P. Wambacq and W. Sansen, "Volterra series and their applications to analog integrated circuit design," in *Distortion Analysis of Analog Integrated Circuits*. Boston, MA, USA: Springer, 1998, pp. 59–115.

[25] P. Valet, D. Schwingshackl, U. Gaier, A. M. Tonello, and D. Giotta, "Inverse modelling techniques as dynamic digital pre-distortion for broadband systems for CS-DACs based on Volterra series expansion," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 71, no. 4, pp. 1506–1519, Apr. 2024.

[26] D. A. George, "Continuous nonlinear systems," Dept. Res. Lab. Electron., Massachusetts Inst. Technol., Cambridge, MA, USA, Tech. Rep. 355, 1959. [Online]. Available: http://hdl.handle.net/1721.1/4466

[27] L. Chua and C. Ng, "Frequency-domain analysis of nonlinear systems: Formulation of transfer functions," *IEE J. Electron. Circuits Syst.*, vol. 3, pp. 257–269, Nov. 1979.

[28] T. Breiten and T. Damm, "Krylov subspace methods for model order reduction of bilinear control systems," *Syst. Control Lett.*, vol. 59, no. 8, pp. 443–450, Aug. 2010.

[29] H. Liu, L. Daniel, and N. Wong, "Model reduction and simulation of nonlinear circuits via tensor decomposition," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 34, no. 7, pp. 1059–1069, Jul. 2015.

[30] P. Benner and T. Breiten, "Two-sided projection methods for nonlinear model order reduction," *SIAM J. Scientific Comput.*, vol. 37, no. 2, pp. B239–B260, Jan. 2015.

[31] G. Flagg and S. Gugercin, "Multipoint Volterra series interpolation and $\mathcal{H}_2$ optimal model reduction of bilinear systems," *SIAM J. Matrix Anal. Appl.*, vol. 36, no. 2, pp. 549–579, Jan. 2015.

[32] P. Benner and T. Breiten, "Interpolation-based $\mathcal{H}_2$-model reduction of bilinear control systems," *SIAM J. Matrix Anal. Appl.*, vol. 33, no. 3, pp. 859–885, 2012.

[33] P. Benner, P. Goyal, and S. Gugercin, "$\mathcal{H}_2$-Quasi-Optimal model order reduction for quadratic-bilinear control systems," *SIAM J. Matrix Anal. Appl.*, vol. 39, no. 2, pp. 983–1032, Jan. 2018.

[34] A. C. Antoulas, I. V. Gosea, and A. C. Ionita, "Model reduction of bilinear systems in the Loewner framework," *SIAM J. Sci. Comput.*, vol. 38, no. 5, pp. B889–B916, Jan. 2016.

[35] I. V. Gosea and A. C. Antoulas, "Data-driven model order reduction of quadratic-bilinear systems," *Numer. Linear Algebra Appl.*, vol. 25, no. 6, Dec. 2018, Art. no. e2200.

[36] X. Y. Z. Xiong, L. J. Jiang, J. E. Schutt-Ainé, and W. C. Chew, "Volterra series-based time-domain macromodeling of nonlinear circuits," *IEEE Trans. Compon., Packag., Manuf. Technol.*, vol. 7, no. 1, pp. 39–49, Jan. 2017.

[37] H. Liu, X. Y. Z. Xiong, K. Batselier, L. Jiang, L. Daniel, and N. Wong, "STAVES: Speedy tensor-aided volterra-based electronic simulator," in *Proc. IEEE/ACM Int. Conf. Comput.-Aided Design (ICCAD)*, Nov. 2015, pp. 583–588.

[38] E. Bedrosian and S. O. Rice, "The output properties of Volterra systems (nonlinear systems with memory) driven by harmonic and Gaussian inputs," *Proc. IEEE*, vol. 59, no. 12, pp. 1688–1707, 1971.

[39] S. Boyd, Y. Tang, and L. Chua, "Measuring Volterra kernels," *IEEE Trans. Circuits Syst.*, vol. CS-30, no. 8, pp. 571–577, Aug. 1983.

[40] T. Wang and T. J. Brazil, "The estimation of Volterra transfer functions with applications to RF power amplifier behavior evaluation for CDMA digital communication," in *IEEE MTT-S Int. Microw. Symp. Dig.*, vol. 1, Jun. 2000, pp. 425–428.

[41] S. Mukherjee, "Vector measurement of nonlinear transfer function," *IEEE Trans. Instrum. Meas.*, vol. 44, no. 4, pp. 892–897, Aug. 1995.

[42] B. Zhang and S. A. Billings, "Volterra series truncation and kernel estimation of nonlinear systems in the frequency domain," *Mech. Syst. Signal Process.*, vol. 84, pp. 39–57, Feb. 2017.

[43] H. J. Sussmann, "Semigroup representations, bilinear approximation of input–output maps, and generalized inputs," in *Mathematical Systems Theory*, G. Marchesini and S. K. Mitter, Eds., Berlin, Germany: Springer, 1976, pp. 172–191.

[44] Cadence Design Syst. (2023). *Cadence Virtuoso*. [Online]. Available: https://www.cadence.com/en_US/home/tools/custom-ic-analog-rf-design/virtuoso-platform.html

[45] Keysight Technol. (2023). *Keysight Advanced Design System (ADS)*. [Online]. Available: https://www.keysight.com/us/en/products/software/pathwave-design-software/pathwave-advanced-design-system.html

[46] T. G. Kolda, "Multilinear operators for higher-order decompositions," Dept. Comput. Sci. Math. Sci. Res. Dept., Sandia Nat. Laboratories, Albuquerque, NM, USA, Tech. Rep. SAND2006-2081, 2006.

[47] S. Boyd and L. Chua, "Fading memory and the problem of approximating nonlinear operators with Volterra series," *IEEE Trans. Circuits Syst.*, vol. CS-32, no. 11, pp. 1150–1161, Nov. 1985.

[48] S. Boyd, L. Chua, and C. A. Desoer, "Analytical foundations of Volterra series," *IMA J. Math. Control Inf.*, vol. 3, pp. 243–282, Jan. 1984.

[49] D. S. Karachalios, I. V. Gosea, L. Gkimisis, and A. C. Antoulas, "Data-driven quadratic modeling in the Loewner framework from input–output time-domain measurements," 2022, *arXiv:2211.10635*.

[50] D. S. Karachalios, I. V. Gosea, and A. C. Antoulas, "A framework for fitting quadratic-bilinear systems with applications to models of electrical circuits," *IFAC-PapersOnLine*, vol. 55, pp. 7–12, Jan. 2022.

[51] Z. Q. Lang, S. A. Billings, R. Yue, and J. Li, "Output frequency response function of nonlinear Volterra systems," *Automatica*, vol. 43, no. 5, pp. 805–816, May 2007.

[52] T. Carleman, "Application de la théorie des équations intégrales linéaires aux systémes d' équations différentielles non linéaires," *Acta Mathematica*, vol. 59, no. 1, pp. 63–87, 1932.

[53] Y. Nakatsukasa, O. Sète, and L. N. Trefethen, "The AAA algorithm for rational approximation," *SIAM J. Sci. Comput.*, vol. 40, no. 3, pp. A1494–A1522, Jan. 2018.

[54] I. V. Gosea and S. Gugercin, "Data-driven modeling of linear dynamical systems with quadratic output in the AAA framework," *J. Sci. Comput.*, vol. 91, no. 1, p. 16, Feb. 2022.

[55] D. S. Karachalios, I. V. Gosea, and A. C. Antoulas, "On bilinear time-domain identification and reduction in the loewner framework," in *Model Reduction of Complex Dynamical Systems*. Cham, Switzerland: Springer, 2021, pp. 3–30.

[56] D. S. Karachalios, "Data-driven system reduction and identification from input–output time-domain data with the Loewner framework," Ph.D. Dissertation, Fakultät für Mathematik, Otto-von-Guericke-University, Madgeburg, Germany, 2023. [Online]. Available: https://pure.mpg.de/rest/items/item_3520850_4/component/file_3571675/content

[57] A. Carlucci, I. V. Gosea, and S. Grivet-Talocia, "Approximation of generalized frequency response functions via vector fitting," in *Proc. Thematic Einstein Semester Math. Optim. Mach. Learn.* Berlin, Germany: De Gruyter, 2025.

[58] A. E. Frazho, "A shift operator approach to bilinear system theory," *SIAM J. Control Optim.*, vol. 18, no. 6, pp. 640–658, Nov. 1980.

[59] A. Isidori and A. Ruberti, "Realization theory of bilinear systems," in *Geometric Methods in System Theory*, D. Q. Mayne and R. W. Brockett, Eds., Dordrecht, The Netherlands: Springer, 1973, pp. 83–130.

[60] T. Nguyen, "Recursive convolution and discrete time domain simulation of lossy coupled transmission lines," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 13, no. 10, pp. 1301–1305, Oct. 1994.

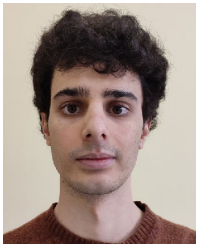
[61] A. Carlucci, T. Bradde, and S. Grivet-Talocia, "Addressing load sensitivity of rational macromodels," *IEEE Trans. Compon., Packag., Manuf. Technol.*, vol. 13, no. 10, pp. 1591–1602, Oct. 2023.

[62] A. Carlucci, T. Bradde, and S. Grivet-Talocia, "Improving robustness to termination conditions in passivity enforcement of rational macromodels," *IEEE Trans. Compon., Packag., Manuf. Technol.*, early access, May 30, 2024, doi: 10.1109/TCPMT.2024.3407526.

[63] L. O. Chua and Y. Liao, "Measuring Volterra kernels III: How to estimate the highest significant order," *Int. J. Circuit Theory Appl.*, vol. 19, no. 2, pp. 189–209, Mar. 1991.

[64] A. D. Libera, R. Carli, and G. Pillonetto, "Kernel-based methods for Volterra series identification," *Automatica*, vol. 129, Jul. 2021, Art. no. 109686.

[65] K. Batselier, Z. Chen, H. Liu, and N. Wong, "A tensor-based Volterra series black-box nonlinear system identification and simulation framework," in *Proc. IEEE/ACM Int. Conf. Comput.-Aided Design (ICCAD)*, Nov. 2016, pp. 1–7.
[66] S. Grivet-Talocia and M. Bandinu, "Improving the convergence of vector fitting for equivalent circuit extraction from noisy frequency responses," *IEEE Trans. Electromagn. Compat.*, vol. 48, no. 1, pp. 104–120, Feb. 2006.
[67] R. J. Gilmore and M. B. Steer, "Nonlinear circuit analysis using the method of harmonic balance—A review of the art. Part I. Introductory concepts," *Int. J. Microw. Millim.-Wave Comput.-Aided Eng.*, vol. 1, no. 1, pp. 22–37, Jan. 1991.
[68] A. J. Redfern and G. T. Zhou, "Performance analysis of Volterra kernel estimators with Gaussian inputs," in *Proc. IEEE Signal Process. Workshop Higher-Order Statist.*, Jul. 1997, pp. 162–165.
[69] R. J. Milliken, J. Silva-Martinez, and E. Sanchez-Sinencio, "Full on-chip CMOS low-dropout voltage regulator," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 54, no. 9, pp. 1879–1890, Sep. 2007.
[70] T. Y. Man, P. K. T. Mok, and M. Chan, "A high slew-rate push–pull output amplifier for low-quiescent current low-dropout regulators with transient-response improvement," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 54, no. 9, pp. 755–759, Sep. 2007.



ANTONIO CARLUCCI (Member, IEEE) received the B.Sc. and M.Sc. degrees in electronic engineering from the Politecnico di Torino, Turin, Italy, in 2019 and 2021, respectively, where he is currently pursuing the Ph.D. degree with the EMC Group. His research interest includes large-scale simulation of electronic systems using macromodeling methods. He received the Best Student Paper Award of SPI 2023 (27th IEEE Workshop on Signal and Power Integrity) and EPEPS 2024 (33rd IEEE Conference on Electrical Performance of Electronic Packaging and Systems).

ION VICTOR GOSEA received the B.Sc. degree in electrical engineering and computer science and the M.Sc. and Ph.D. degrees in electrical engineering from Jacobs University Bremen, Germany, in 2011, 2013, and 2017, respectively, under the guidance of Prof. Athanasios C. Antoulas. Since February 2017, he has been affiliated with the Max Planck Institute for Dynamics of Complex Technical Systems, Magdeburg, Germany, first with the DRI Group (until October 2022), and afterward with the CSC Group led by Prof. Peter Benner, until the present day. His research interests include reduced-order modeling, data-driven complexity reduction methods, scientific computing, numerical analysis, system and control theoretical model reduction techniques, and numerical linear algebra.

STEFANO GRIVET-TALOCIA (Fellow, IEEE) received the Laurea and Ph.D. degrees in electronic engineering from the Politecnico di Torino, Turin, Italy, in 1994 and 1998, respectively. From 1994 to 1996, he was with the NASA/Goddard Space Flight Center, Greenbelt, MD, USA. He is currently a Full Professor of electrical engineering with the Politecnico di Torino. He co-founded the academic spinoff company IdemWorks, Turin, Italy, in 2007, and was the President until its acquisition by CST, in 2016. He has authored more than 200 journal and conference papers. His current research interests include passive macromodeling of lumped and distributed interconnect structures, model-order reduction, modeling and simulation of fields, circuits, and their interaction, wavelets, time-frequency transforms, and their applications. He was a co-recipient of the 2007 Best Paper Award of IEEE TRANSACTIONS ON ADVANCED PACKAGING. He received the IBM Shared University Research Award in 2007, 2008, and 2009, and the Intel Strategic Research Segment Grant in 2022, 2023, and 2024. He was the General Chair of the 20th and 21st IEEE Workshops on Signal and Power Integrity (SPI2016 and SPI2017), the Co-Chair of SPI2023, and the Program Co-Chair of SPI2024. He was an Associate Editor of IEEE TRANSACTIONS ON ELECTROMAGNETIC COMPATIBILITY from 1999 to 2001. He is also serving as an Associate Editor for IEEE TRANSACTIONS ON COMPONENTS, PACKAGING AND MANUFACTURING TECHNOLOGY.