POLITECNICO DI TORINO
Repository ISTITUZIONALE

On the Generation of SPICE-Compatible Nonlinear Behavioral Macromodels

Original
On the Generation of SPICE-Compatible Nonlinear Behavioral Macromodels / Carlucci, Antonio; Gosea, Ion Victor; Grivet-Talocia, Stefano. - In: IEEE TRANSACTIONS ON COMPONENTS, PACKAGING, AND MANUFACTURING TECHNOLOGY. - ISSN 2156-3950. - 15:9(2025), pp. 1857-1867. [10.1109/tcpmt.2025.3597771]

Availability:
This version is available at: 11583/3003124 since: 2025-09-18T06:35:43Z

Publisher:
IEEE

Published
DOI:10.1109/tcpmt.2025.3597771

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright
IEEE postprint/Author's Accepted Manuscript

©2025 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collecting works, for resale or lists, or reuse of any copyrighted component of this work in other works.

(Article begins on next page)

10 October 2025

On the Generation of SPICE-Compatible Nonlinear Behavioral Macromodels

Antonio Carlucci*, *Member, IEEE*, Ion Victor Gosea†, Stefano Grivet-Talocia*, *Fellow, IEEE*
*Dept. Electronics and Telecommunications, Politecnico di Torino, Italy
† Max Planck Institute for Dynamics of Complex Technical Systems, CSC group, Magdeburg, Germany
antonio.carlucci@polito.it, gosea@mpi-magdeburg.mpg.de, stefano.grivet@polito.it

***Abstract*—This paper presents a complete framework for the generation of behavioral macromodels of a wide class of nonlinear components, devices, and systems. The model structure and related identification algorithms are based on the Volterra series formulated in the frequency domain through multivariate Generalized Transfer Functions. A multivariate rational model is first estimated in pole-residue form from sampled responses and then converted to a bilinear state-space form. The main novel contribution of this work is a SPICE-compatible circuit synthesis, which enables the usage of nonlinear macromodels within circuit simulation environments as part of more complex system-level simulations. Examples are provided for a Low DropOut voltage regulator and for a system-level Power Distribution Network embedding integrated regulators. For such examples, the proposed SPICE equivalents offer speedup factors ranging from $12\times$ up to $650\times$ with negligible loss in accuracy.**

I. INTRODUCTION

Computer-Aided Design (CAD) of modern packaged semiconductor systems calls for repeated numerical simulations, aimed at reproducing and predicting both wanted and undesired parasitic interactions that may have an influence on the system behavior. Such numerical simulations must be performed at the system level, including both (full-wave) electromagnetic representations and detailed circuit- and sometimes transistor-level descriptions. A notable example is system-level Power Integrity (PI) verification of Power Delivery Networks (PDNs) equipped with banks of integrated voltage regulators [1], [2].

In order to reduce overall complexity, behavioral modeling or *macromodeling* techniques are extensively used. The most notable and ubiquitous example is macromodeling of Linear and Time-Invariant (LTI) structures through rational approximation of frequency responses, typically available in the form of tabulated Scattering parameters obtained by Maxwell equations solvers [3]. Such structures include all types of interconnects, transmission lines, connectors, vias, power planes, and grids, to name a few. The Vector Fitting (VF) algorithm for rational approximation [4] and its derivatives [5]–[9], as well as associated passivity enforcement schemes [10]–[14], are now available in most commercial CAD software products and represent established tools in Signal and Power Integrity modeling and simulation flows.

In addition to the availability of robust algorithms for model generation, a key enabling factor for the widespread use of such behavioral models is the ability to incorporate them as components in larger simulation decks, typically in a SPICE environment [15]–[17]. For the particular case of LTI models based on rational fitting and VF, direct circuit synthesis is straightforward: the rational function model is first realized as a set of Ordinary Differential Equations (ODEs) in state-space form, which are then synthesized as a behavioral SPICE netlist including controlled sources. Advanced circuit solvers also allow a direct simulation through recursive convolutions, which further speed up the solution by skipping the state-space synthesis step. See [3] for a complete overview.

We remark that rational functions in the frequency domain and the associated linear state-space (or descriptor) realizations are a universal representation for LTI systems. This generality was also one of the key factors leading to the success of linear macromodeling over the last few years. In this work, we concentrate on the more difficult problem of nonlinear macromodeling, for which the definition and use of a universal representation pose significant challenges.

Several examples exist of compact behavioral modeling approaches for nonlinear structures. Notable examples include the widespread IBIS framework [18], which provides an industry-validated solution to represent high-speed drivers and receivers for channel modeling and simulation. The success of IBIS models relies on the fact that the underlying transistor-level structures have a well-defined structure and topology, which is mimicked in the behavioral model. In this respect, this representation is not general and is valid only for the particular structures for which it is designed. This limitation affects several other more sophisticated approaches for transceiver modeling, such as [19]–[21].

In this work, we consider another representation of nonlinear analog circuit blocks, based on the Volterra series theory [22], [23]. This representation is general under a theoretical standpoint, but its practical application can be envisioned only for those systems characterized by a mild nonlinearity. Several application examples fall in this class, including various types of voltage regulators as well as system-level models of PDNs that embed voltage regulator stages.

This paper presents a nonlinear modeling framework for constructing compact, data-driven models of nonlinear components and circuits that can be synthesized as SPICE-compatible behavioral networks. The approach heavily builds on the results in [24], [25], where the methodological foundations are provided to build nonlinear models starting from frequency-domain data. The theory presented in [24], [25] shows that, by combining the Volterra series formalism with a multi-

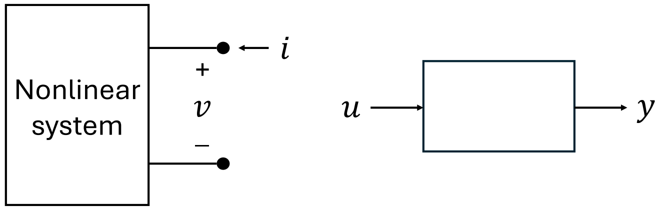

Fig. 1. Nonlinear one-port system and its system-theoretic representation.

dimensional Laplace transform and an iterative multivariate rational approximation scheme that suitably generalizes VF, compact nonlinear macromodels can be easily obtained with high accuracy and reliability.

Consider a nonlinear system (a general term to refer to a device, component, or circuit) whose interface with the external environment is defined by an electrical port with an associated port voltage $v(t)$ and current $i(t)$, as shown in Fig. 1(left). By choosing a particular variable (e.g. port voltage/current, impinging scattering wave, etc.) as the system input, denoted as $u(t)$, and a different variable as the output $y(t)$, this system can be viewed as a (nonlinear dynamic) mapping from u to y [see Fig. 1(right)]. The material in [24] shows how to obtain a nonlinear macromodel that approximates with good accuracy this nonlinear dynamic mapping. The use of macromodels is, however, limited to evaluating the output $y(t)$ when the input $u(t)$ is a known and fixed signal. As such, the macromodel cannot be used as an equivalent circuit block to be interconnected with other circuit blocks in a larger system-level modeling and simulation scenario. Therefore, applicability remains severely limited.

In this work, we fill this gap by completing the nonlinear macromodeling framework with a SPICE synthesis. Starting from the model representation in [24] and the associated state-space realization that is designed to be *bilinear*, i.e., including additional terms with products between input and states, we show how to construct a behavioral SPICE netlist that can be used as a black-box in circuit simulation environments and tools. Termination or interconnection of the resulting macromodel can be arbitrary, as required for the system analysis to be performed.

The structure of this paper is as follows. Section II reviews known background material to make this manuscript reasonably self-contained. In particular, bilinear macromodels and their Volterra series representation in the time and frequency domains are discussed, together with a summary of the multivariate rational approximation scheme presented in [24]. Also, a standard state-space synthesis approach via controlled sources is discussed in Sec. II-D, since it forms one of the building blocks of the proposed nonlinear synthesis. The latter is detailed in Sec. III, where the general topology, various alternatives for SPICE realization, and a complexity analysis are presented. Section IV presents validation and numerical results on a single voltage regulator and on a more complex voltage-regulated PDN structure.

II. BACKGROUND

Linear macromodeling based on Vector Fitting was recently extended to nonlinear systems through the methodology described in [24], [25], which is based on Volterra series theory. This section reviews the background material that allows the construction of data-driven behavioral macromodels of weakly nonlinear systems.

A. Volterra series representation of nonlinear systems

The cornerstone of the method developed in [24] is the theory of Volterra series [22], which allows the representation of the input-output behavior of a nonlinear system. Although effective in large-signal conditions where the nonlinear nature of the underlying system is evident, the notion of Volterra series is still *local* and centered around an *operating point* of the device. Let us assume that, for a given stationary value $\bar{u}$ of the input $u(t) = \bar{u}$, the constant steady-state system output is $\bar{y}$. Then, to study the system behavior around this operating point, we decompose the signals as follows

$$\begin{aligned} u(t) &= \bar{u} + \tilde{u}(t) \\ y(t) &= \bar{y} + \tilde{y}(t) \end{aligned} \tag{1}$$

With this notation, the Volterra series expresses the output $y(t)$ as an infinite series whose terms are multidimensional convolution integrals, defined as

$$\begin{aligned} &y(t) = \bar{y} + \\ &\sum_{m=1}^{\infty} \int_0^t \cdots \int_0^t h_m(\tau_1, \ldots, \tau_m) \tilde{u}(t-\tau_1) \cdots \tilde{u}(t-\tau_m) \mathrm{d}\boldsymbol{\tau} \end{aligned} \tag{2}$$

where $\bar{y}$ is the system response to a stationary input $u(t) = \bar{u}$. In (2), the shorthand $\mathrm{d}\boldsymbol{\tau} \triangleq \mathrm{d}\tau_1 \cdots \mathrm{d}\tau_m$ is used, and the multivariate functions $h_m(\tau_1, \ldots, \tau_m)$ appearing in the integrals are the *Volterra kernels*, which are symmetric functions of their arguments (see [22, Sec. 1.2] for more details).

Applying the multidimensional Laplace transform [26] to the generic kernel h_m of degree m leads to the definition of the Generalized Transfer Function (GTF)

$$\begin{aligned} &H_m(s_1, \ldots, s_m) = \\ &\quad \int_0^{\infty} \cdots \int_0^{\infty} h_m(\tau_1, \ldots, \tau_m) e^{-(s_1\tau_1 + \cdots + s_m\tau_m)} \mathrm{d}\boldsymbol{\tau}. \end{aligned} \tag{3}$$

Therefore, higher-degree nonlinear contributions in the input-output operator are represented as multivariate frequency responses with a higher number of independent frequency (Laplace) arguments. We refer the Reader to [23] for an in-depth discussion of the mathematical aspects of Volterra theory. It is to be noted that Volterra GTFs are also related to X-parameters [27], [28]. In fact, samples of the GTFs can be extracted from X-parameter data, as discussed in [29].

B. Multivariate rational approximation of Volterra kernels

The analysis of the nonlinear macromodeling method developed in this work begins with considering sampled values of the GTFs, which are a frequency-domain representation of the behavior of the underlying system. Samples of the generic degree-m GTF are collected at a set of K_m points $\boldsymbol{s}_m^{(k)} \triangleq (s_1^{(k)}, \ldots, s_m^{(k)})$ and they are denoted as $\breve{H}_m^{(k)} \triangleq H_m(s_1^{(k)}, \ldots, s_m^{(k)})$, with k being the sample index. These samples are collected for several degrees, starting from $m = 1$ up to a maximum degree M.

The modeling procedure described in [24] consists of optimizing poles and residues of a conveniently defined multivariate rational model $\hat{H}_m$,

$$\hat{H}_m(s_1, \ldots, s_m) = \sum_{n_1=1}^{\nu_1} \cdots \sum_{n_m=1}^{\nu_m} r_{n_1,\ldots,n_m} \varphi^{\text{sym}}_{n_1,\ldots,n_m}(s_1, \ldots, s_m) \quad (4)$$

where $r_{n_1,\ldots,n_m} \in \mathbb{C}$ are constant coefficients (termed *generalized residues*). The basis functions $\varphi^{\text{sym}}_{n_1,\ldots,n_m}(s_1, \ldots, s_m)$ are suitably-defined multivariate rational functions that depend on M sets of model poles $\mathcal{P}_m = \{p^{(m)}_{n_m}\}_{n_m=1}^{\nu_m} \subset \mathbb{C}$, with ν_m poles in each set. We first introduce the auxiliary functions $\varphi_{n_1,\ldots,n_m}(s_1, \ldots, s_m)$, which are defined recursively as [24]

$$\varphi_{n_1}(s_1) = \frac{1}{s_1 - p^{(1)}_{n_1}}, \quad (5a)$$

$$\varphi_{n_1,\ldots,n_m}(s_1, \ldots, s_m) = \frac{\varphi_{n_1,\ldots,n_{m-1}}(s_1, \ldots, s_{m-1})}{s_1 + \cdots + s_m - p^{(m)}_{n_m}} \quad (5b)$$

Using these auxiliary functions, the basis functions $\varphi^{\text{sym}}_{n_1,\ldots,n_m}(s_1, \ldots, s_m)$ appearing in (4) are the *symmetrizations* of the $\varphi_{n_1,\ldots,n_m}$'s,

$$\varphi^{\text{sym}}_{n_1,\ldots,n_m}(s_1, \ldots, s_m) \triangleq \frac{1}{m!} \sum_{\sigma \in \mathbb{P}_m} \varphi_{n_1,\ldots,n_m}(s_{\sigma(1)}, \ldots, s_{\sigma(m)}) \quad (6)$$

where the sum runs over the set $\mathbb{P}_m$ of all permutations σ of the integers $\{1, \ldots, m\}$. More precisely, φ^{sym} is obtained by summing the values of $\varphi_{n_1,\ldots,n_m}$ over all the $m!$ permutations of its m arguments.

For each degree $m = 1, \ldots, M$, the rational model $\hat{H}_m$ is fitted to the data $\breve{H}_m^{(k)}$ by optimizing its generalized residues and the model poles through an extension of the VF algorithm. In this optimization, the objective is to minimize the model-data error represented by the squared loss

$$J = \sum_{m=1}^{M} \sum_{k_m=1}^{K_m} \left| \breve{H}_m^{(k_m)} - \hat{H}_m\left(s_1^{(k_m)}, \ldots, s_m^{(k_m)}\right) \right|^2. \quad (7)$$

Leveraging the recursive structure of (5), the model poles can be found by an iterative process that identifies the set $\mathcal{P}_m$ in several steps starting from $m = 1$ to $m = M$. In fact, a consequence of (5) is that the GTF model $\hat{H}_m$ can be factorized as a product [24]

$$\hat{H}_m(s_1, \ldots, s_m) = \boldsymbol{F}_m(s_1 + \cdots + s_m) \boldsymbol{Q}_m(s_1, \ldots, s_m) \quad (8)$$

where the leading factor $\boldsymbol{F}_m(s_1 + \cdots + s_m)$ is univariate and is the only one that depends on the degree-m poles $\mathcal{P}_m$, whereas $\boldsymbol{Q}_m(s_1, \ldots, s_m)$ only depends on the (known) pole sets $\mathcal{P}_\mu$ of degree $\mu < m$. At each step, the elements of $\mathcal{P}_m$ are estimated by running the VF iteration to optimize the poles of $\boldsymbol{F}_m$, considering $\boldsymbol{Q}_m$ known from previous iteration steps.

C. Bilinear macromodels

Upon completion of the fitting phase, the frequency-domain rational model (4) can be equivalently recast as a bilinear state-space model (i.e., a time-domain representation),

$$\dot{\boldsymbol{x}} = \mathbf{A}\boldsymbol{x} + \mathbf{N}\boldsymbol{x}\tilde{u} + \boldsymbol{b}\tilde{u} \quad (9a)$$

$$\tilde{y} = \boldsymbol{c}^{\mathsf{T}}\boldsymbol{x} \quad (9b)$$

where the matrix $\mathbf{A}$ depends on model poles, and the vector $\boldsymbol{c}$ contains the generalized residues. Compared to standard state-space representations of LTI systems, (9) includes an additional nonlinear term associated with the matrix $\mathbf{N}$, with products of the input signal $\tilde{u}$ with all state variables. It is important to remark that $\mathbf{A} \in \mathbb{C}^{N \times N}$ is a block-diagonal matrix,

$$\mathbf{A} = \text{blkdiag}\{\mathbf{A}_1, \ldots, \mathbf{A}_m\} \quad (10)$$

where $\mathbf{A}_1 = \text{diag}\{p_n^{(1)}\}_{n=1}^{\nu_1}$ and the m-th diagonal block $\mathbf{A}_m \in \mathbb{C}^{\nu_1 \cdots \nu_m \times \nu_1 \cdots \nu_m}$ is

$$\mathbf{A}_m = \mathbf{A}'_m \otimes \mathbb{I}_{\nu_1 \cdots \nu_{m-1}}, \quad \mathbf{A}'_m \triangleq \text{diag}\{p_n^{(m)}\}_{n=1}^{\nu_m} \quad (11)$$

with $\otimes$ denoting the matrix Kronecker product. Moreover, $\mathbf{N}$ is a block-subdiagonal concatenation of the following submatrices

$$\mathbf{N}_m = \mathbf{1}_{\nu_m} \otimes \mathbb{I}_{\nu_1 \cdots \nu_{m-1}}, \quad m \geq 2. \quad (12)$$

with the blocks $\mathbf{N}_m$ placed on the subdiagonal of $\mathbf{N}$ [22], [24]. In case some poles are complex-valued, standard coordinate transformations [3] can be used to construct a real-valued realization.

Although it may be deemed simplistic, the class of bilinear systems (9) has proven to be adequate to approximate much larger classes of nonlinear systems [30], [31]. This result is remarkable, since the bilinear model structure encodes nonlinearity by simple state-input products. In the presence of stronger nonlinearities (e.g., higher degree polynomials), constructive ways exist to manipulate the corresponding state-space equations and transform them into a bilinear system by means of *lifting* techniques [32]–[35]. This procedure generally leads to an increase in the number of state variables. A widely applicable technique is *Carleman linearization*, discussed in detail in [22]. Appendix A contains a practical demonstration of this technique as applied to a PDN structure with integrated voltage regulators represented through time-averaged models. This derivation shows that a bilinear model structure is indeed adequate to represent the dynamics of voltage-regulated PDNs. The foregoing Sec. IV will further confirm this conclusion by demonstrating the proposed behavioral model synthesis on a practical regulated PDN example.

In [24], it was shown that systems of practical relevance can be modeled using this technique, and the resulting macromodel is typically solved in the time domain very efficiently to obtain a close approximation of the original system response. In that work, the macromodel was always solved via a custom implementation of the recursive convolution method in MATLAB. In this work, we start from the bilinear state-space macromodel (9) and synthesize it into an equivalent SPICE-compatible netlist that can be instantiated as a component in a generic

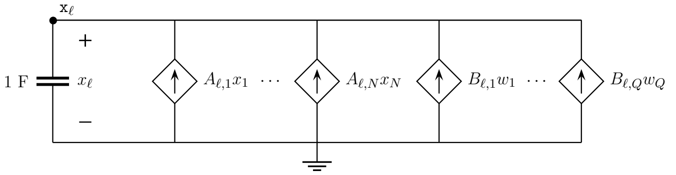

Fig. 2. Circuit corresponding to the state equation (14) describing the dynamic evolution of the ℓ-th state variable, here interpreted as the voltage across the capacitor (nodal voltage of node x_ℓ).

transient analysis, thus fully providing a compact behavioral representation of the original system.

For the sake of completeness, it should be remarked that alternative modeling strategies exist, such as those based on Wiener or Hammerstein structures. If polynomial nonlinearities are considered, it can be shown [30] that the Wiener structure can be equivalently represented as a particular instance of a bilinear system. Identification of particular Wiener models using Vector Fitting is addressed in [36]. Similarly, the Hammerstein model is seen to be less flexible than the model structure used here because its GTFs are of the type $H_m(s_1, \ldots, s_m) = f_m(s_1 + \cdots + s_m)$, i.e., are only functions of $s_1 + \cdots + s_m$. Scenarios with more general nonlinearities can be formulated through block-oriented models, whose identification is thoroughly discussed in [37].

D. Direct circuit synthesis of LTI macromodels

When using Vector Fitting [38] to build macromodels of linear components, the rational transfer function model obtained by rational fitting is usually represented in a state-space format,

$$\dot{\boldsymbol{x}} = \mathbf{A}\boldsymbol{x} + \mathbf{B}\boldsymbol{w} \tag{13a}$$

$$\boldsymbol{z} = \mathbf{C}\boldsymbol{x} + \mathbf{D}\boldsymbol{w} \tag{13b}$$

where $\boldsymbol{w}(t) \in \mathbb{R}^Q$ is the vector of input signals, $\boldsymbol{z}(t) \in \mathbb{R}^P$ contains the output signals, and $\mathbf{A} \in \mathbb{R}^{N \times N}$.

The state-space realization (13) refers to a general Multi-Input Multi-Output (MIMO) LTI system with Q inputs and P outputs, and it is a representation that applies to a large class of LTI systems. The state-space model (13), expressed in terms of differential equations, can be equivalently translated or synthesized into an electrical network. Although network synthesis can be a hard problem when only RLC(k) elements are allowed (see [39], [40]), in this context we are interested in a behavioral (i.e., non-physical) representation, in which case synthesizing (13) is straightforward by using controlled sources. A well-known solution is detailed in [3, Sec. 11.6.2] and reviewed in the following. This procedure is best explained by considering individual entries of the matrix coefficients in the state equation (13a), as follows

$$\dot{x}_\ell(t) = \sum_{j=1}^{N} A_{\ell,j} x_j(t) + \sum_{q=1}^{Q} B_{\ell,q} w_q(t), \qquad \ell = 1, \ldots, N \tag{14}$$

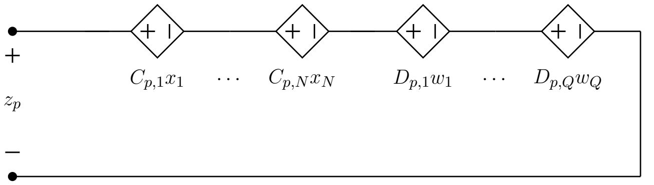

Fig. 3. Circuit corresponding to the output equation (15) in case the variable $z_p(t)$ is a voltage.

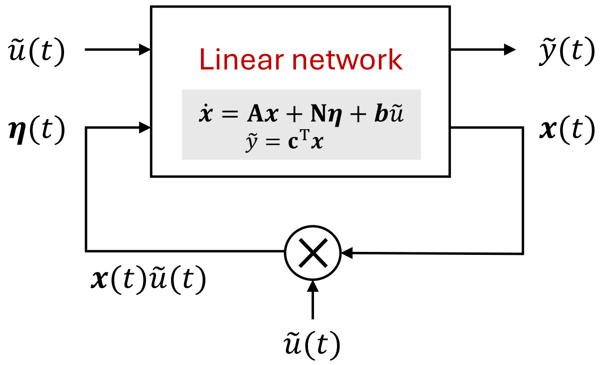
$\tilde{u}(t)$
Linear network
$\dot{x} = \mathbf{A}x + \mathbf{N}\boldsymbol{\eta} + \boldsymbol{b}\tilde{u}$
$\tilde{y} = \mathbf{c}^{\mathrm{T}}\boldsymbol{x}$
$\tilde{y}(t)$
$\boldsymbol{\eta}(t)$
$\boldsymbol{x}(t)$
$\boldsymbol{x}(t)\tilde{u}(t)$
$\tilde{u}(t)$

Fig. 4. Block-schematic representation of the bilinear system (16).

where the indices ℓ and j run over the individual state variables, that are the entries of $\boldsymbol{x}$. By interpreting $x_\ell(t)$ as a voltage, the differential equation (14) can be interpreted as Kirchoff's Current Law where the right-hand side is a sum of currents flowing through a capacitor, as shown in Fig. 2. We remark that, in case the $\mathbf{A}$ matrix is sparse, all the controlled sources with zero gain $A_{\ell,j} = 0$ can be omitted, resulting in a much simpler network that is also solved faster by the SPICE engine. Also note that a dual alternative is possible where $x_\ell(t)$ is interpreted as a current, in which case several series-connected controlled voltage sources would be used in place of the shunt current sources of Fig. 2. See [3, Sec. 11.6.2] for an in-depth discussion.

Regarding the output equation, consider a similar rewriting

$$z_p(t) = \sum_{j=1}^{N} C_{p,j} x_j(t) + \sum_{q=1}^{Q} D_{p,q} w_q(t), \qquad p = 1, \ldots, P \tag{15}$$

where the $C_{p,j}$'s are the entries of $\mathbf{C}$. In case the output variables $z_p(t)$ represent the port voltages of the system that is being modeled, these equations can be synthesized as in Fig. 3.

To summarize, the overall network that synthesizes (13) is made up of N subcircuits as in Fig. 2, each corresponding to a different state variable x_ℓ, coupled with the subcircuits in Fig. 3 that generate the outputs $\boldsymbol{z}(t)$.

III. CIRCUIT SYNTHESIS

This section describes the circuit realization of the bilinear behavioral models reviewed in Sec. II. The bilinear model (9) differs from a standard LTI system only for the presence of the bilinear term $\boldsymbol{x}\tilde{u}$. This allows synthesizing a circuit that implements the differential equation (9a) by mostly building on the existing technique for LTI systems reviewed in Sec. II-D.

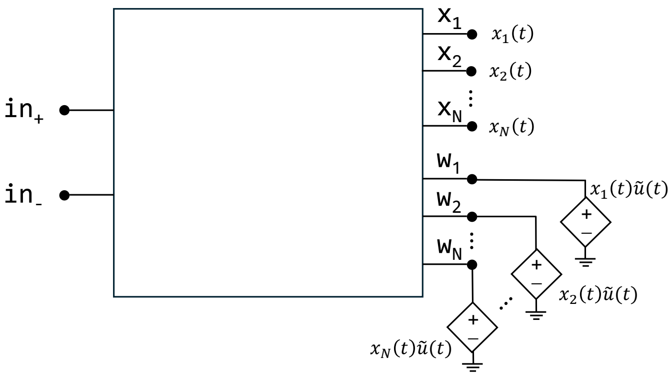

Fig. 5. Representation of a SPICE subcircuit implementing the linear network of Fig. 4.

In fact, the core idea used here is to view the system (9) as a linear system coupled with a nonlinear feedback law. To this aim, let us introduce the auxiliary variable $\boldsymbol{\eta} \triangleq \boldsymbol{x}\tilde{u}$, which allows rewriting (9a) in the equivalent format

$$\dot{\boldsymbol{x}} = \mathbf{A}\boldsymbol{x} + \mathbf{N}\boldsymbol{\eta} + \boldsymbol{b}\tilde{u} \tag{16a}$$

$$\boldsymbol{\eta} = \boldsymbol{x}\tilde{u} \tag{16b}$$

This simple rewriting enables viewing the bilinear macromodel as the combination of a linear system (16a), which has an additional fictitious input $\boldsymbol{\eta}$, coupled with a static feedback law (16b). This setting corresponds to the block schematic in Fig. 4.

We see that (16a) is a linear system with an augmented set of inputs (namely, $\boldsymbol{\eta}$ and $\tilde{u}$) compared to (9a), whose input is just $\tilde{u}$. This linear part (16a) can be synthesized into a SPICE netlist following Sec. II-D, considering $\boldsymbol{\eta}$ and $\tilde{u}$ as the inputs, and the entire state vector $\boldsymbol{x}$ as an output of this sub-block. In fact, the linear synthesis procedure reviewed in Sec. II-D can be readily applied to generate a SPICE netlist that implements such a MIMO system, because the linear system (16a) is a particular case of (13) with

$$\boldsymbol{w} \triangleq \begin{pmatrix} \boldsymbol{\eta} \\ \tilde{u} \end{pmatrix}, \quad \boldsymbol{z} \triangleq \boldsymbol{x} \tag{17}$$

Consider now the nonlinear relation (16b), corresponding to the nonlinear feedback block in Fig. 4. This equation can be implemented using behavioral sources that take individual entries $x_\ell(t)$ of $\boldsymbol{x}$ and multiply them by the input signal $\tilde{u}(t)$ at every time instant. The implementation is easily carried out by using behavioral controlled sources, as discussed in the following.

A. Implementation in SPICE

In synthesizing the linear network block in Fig. 4 by means of the direct synthesis approach reviewed in Sec. II-D, an arbitrary choice about whether to interpret the state variables $x_\ell(t)$ as currents or voltages should be made. For the sake of exposition, let us assume that $x_\ell(t)$, as well as the auxiliary inputs in the vector $\eta_\ell(t) = x_\ell(t)u(t)$, are interpreted as voltages. Applying the procedure of Sec. II-D leads to a SPICE subcircuit, shown in Fig. 5, whose interface includes a pair of terminals, namely `in+` and `in-`, defining a main electrical port associated to variables $\tilde{u}$ and $\tilde{y}$. In addition to that, Fig. 5 shows a set of N terminals $\mathtt{x}_\ell$ whose nodal voltages are $x_\ell(t)$ (upper-right side of Fig. 5), and a set of N terminals whose nodal voltages are the inputs $\eta_\ell(t)$, associated with the nodes $\mathtt{w}_\ell$ in the lower-right side of Fig. 5.

The products $x_\ell(t)\tilde{u}(t)$ are realized by means of controlled sources, as shown in the lower-right part of Fig. 5. Assuming that $\tilde{u}$ is the port voltage, there are at least two ways to carry out this implementation in HSPICE. For example, to implement $\eta_1(t) = x_1(t)\tilde{u}(t)$, a first alternative is to use literal expressions such as

```
E1 w1 0 vol='V(x1,0)*V(in+,in-)' ,
```

which is valid in HSPICE. In LTSpice, a behavioral (`B`) voltage (`V`) source would be used with the following syntax

```
BV1 w1 0 V='V(x1,0)*V(in+,in-)' .
```

An alternative implementation valid in HSPICE is through polynomial sources, defined as follows:

```
E1 w1 0 POLY(2) x1 0 in+ in- 0 0 0 1
```

Note that the network described above is a circuit realization of the model (9) that links $\tilde{u}$ and $\tilde{y}$. These are related to the original variables u and y through constant shifts $\bar{u}$, $\bar{y}$. It is easy to introduce independent constant sources around the terminals `in+`, `in-` in Fig. 5 to add the shifts $\bar{u}$ and $\bar{y}$.

B. Complexity Analysis

Bilinear macromodels obtained with the methodology in Sec. II-C can easily lead to realizations (9) where the number of state variables N is large. In fact, using the simplest realization procedure outlined in [24], $N = \nu_1 + \nu_1\nu_2 + \cdots + \nu_1 \cdots \nu_M$. This might suggest that the SPICE netlist describing the circuit in Fig. 5 is complex and inefficient to simulate. This is not the case because the bilinear realization is very sparse.

To synthesize the linear block in Fig. 5, the direct synthesis approach uses N capacitors. The number of controlled sources in the circuit depends on the number $|\mathbf{A}|$ of nonzero entries in the $\mathbf{A}$ matrix. Because of (10), this is

$$|\mathbf{A}| = \sum_{m=1}^{M} |\mathbf{A}_m| \, . \tag{18}$$

After $\mathbf{A}_1$ is turned into a real quasi-diagonal realization with at most 2×2 blocks on its main diagonal (as discussed in [3, Sec. 8.3]), the number of nonzero elements is given by

$$|\mathbf{A}_1| = \nu_1^{(r)} + 2\nu_1^{(c)} \tag{19}$$

where $\nu_1^{(r)}$ is the number of real poles and $\nu_1^{(c)}$ the number of complex poles in $\mathcal{P}_1$. Similarly for $\mathbf{A}'_m$,

$$|\mathbf{A}'_m| = \nu_m^{(r)} + 2\nu_m^{(c)}. \tag{20}$$

The relation (11) implies that

$$|\mathbf{A}_m| = \nu_1 \cdots \nu_{m-1} \, |\mathbf{A}'_m| = \nu_1 \cdots \nu_{m-1}(\nu_m^{(r)} + 2\nu_m^{(c)}) \tag{21}$$

The non-zero entries of the input-state matrices $\mathbf{N}$ and $\boldsymbol{b}$ also determine the number of controlled sources in the

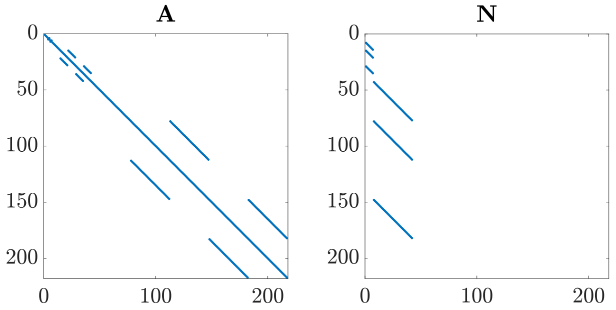

Fig. 6. Sparsity patterns of the $\mathbf{A}$ and $\mathbf{N}$ after bringing the realization in the real-valued quasi-diagonal form following [24, Remark 1].

subcircuits of the type in Fig. 2. In a real-valued realization, obtained by the change of coordinate discussed in [24, Remark 1], it can be seen that

$$|\boldsymbol{b}| = \nu_1^{(r)} + \frac{1}{2}\nu_1^{(c)} \tag{22}$$

and, based on the relation (12),

$$|\mathbf{N}_m| = \nu_1 \cdots \nu_{m-1}(\nu_m^{(r)} + \frac{1}{2}\nu_m^{(c)}) \tag{23}$$

The sparsity patterns of $\mathbf{A}$ and $\mathbf{N}$ in the real quasi-diagonal format are shown in Fig. 6, where each point in the square corresponds to a non-vanishing entry of the matrix $\mathbf{A}$ or $\mathbf{N}$. It is graphically apparent that most matrix entries are zero. The number of controlled sources required to synthesize the state equation (16) is

$$|\mathbf{A}| + |\mathbf{N}| + |\boldsymbol{b}| = \sum_{m=1}^{M} \nu_1 \cdots \nu_{m-1}(2\nu_m^{(r)} + \frac{5}{2}\nu_m^{(c)}), \tag{24}$$

which scales like $N = \sum_{m=1}^{M} \nu_1 \cdots \nu_m$.

In summary, the realization matrices are sparse because the number of non-zero entries scales linearly with the size N of the matrix. This is advantageous in practice because time-domain circuit simulation requires solving linear systems of equations, which is done more efficiently when using sparse matrices. Direct sparse methods (e.g. KLU [41], SuperLU [42]) typically have complexity between $O(N^{1.5})$ and $O(N^2)$ [43], [44] (although it can be worse for some problems due to fill-in), whereas iterative solvers show a per-iteration cost that scales linearly with N. Therefore, in terms of number of poles $\bar{\nu} \triangleq \max_m \nu_m$ and Volterra order M, a direct solver would take approximately $O(\bar{\nu}^{1.5M})$ operations, and an iterative solver $O(\bar{\nu}^M)$ operations per iteration.

IV. NUMERICAL RESULTS

This section reports numerical results that demonstrate the validity of the proposed circuit synthesis method. Section IV-A considers a nonlinear macromodel of a Low-DropOut (LDO) voltage regulator to show the correctness of the synthesis method described in Sec. III. The same macromodel is then simulated in combination with several capacitive loads, and the results are compared to the true system responses in Sec. IV-B. A second example concerning a system-level model of a Power Delivery Network with integrated regulators is presented in Sec. IV-C.

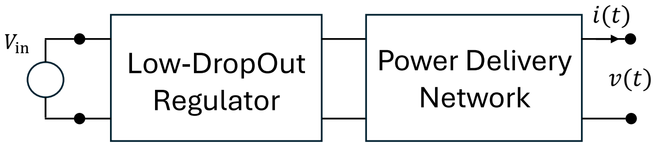

Fig. 7. Schematic representation of the example of Sec. IV-A.

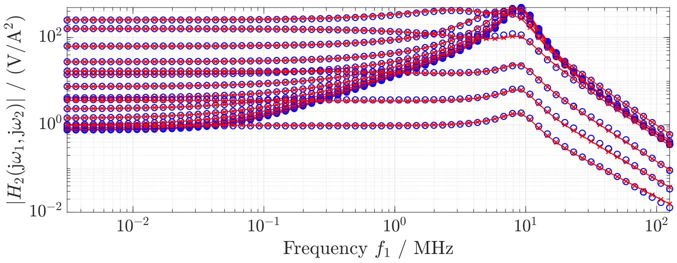

Fig. 8. Model-data comparison for the LDO example described in Sec. IV-A.

A. Validation: a Low-DropOut Voltage Regulator

As a first example, we consider a Power Delivery Network including an LDO voltage regulator combined with a passive network representing a Printed Circuit Board (PCB). A transistor-level view of the LDO circuit, originally proposed and described in [45], was instantiated in HSPICE, together with a passive model of the PCB as depicted in Fig. 7. The purpose of this system is to provide a regulated supply voltage to other analog blocks connected to its output (electrical port on the right side in Fig. 7).

In order to simulate the system behavior in combination with other (different) loads, we built a nonlinear macromodel of its output *impedance*, i.e. the relation between $i(t)$ and $v(t)$ in Fig. 7, using the algorithm presented in [24] and reviewed in Sec. II-C. Following [24], a characterization of the device was extracted using Harmonic Balance (HB) simulations to obtain samples of the GTFs according to the harmonic probing method [22], [46]. In particular, the large-signal behavior of the system was characterized around an operating point corresponding to a bias current $\bar{i} = 20\,\text{mA}$. For this constant input, the DC value of the output voltage is $\bar{v} \approx 2.710\,\text{V}$. HB analysis with up to three large-signal tones yielded samples of GTFs of degree up to three, in particular $K_1 = 55$, $K_2 = 9940$, $K_3 = 1200$ samples.

The GTF data was then fitted to a multivariate rational model with $\nu_1 = 8$, $\nu_2 = 8$, $\nu_3 = 5$ poles. The number of poles was manually increased until a satisfactory model-data error was reached. Figure 8 shows a frequency-domain

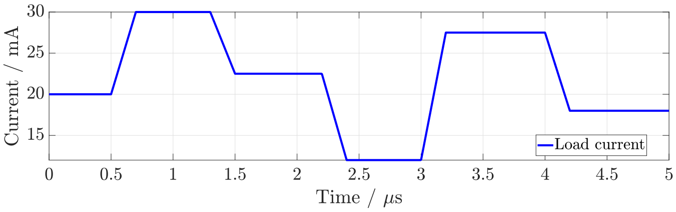

Fig. 9. Load current signal for the example in Sec. IV-A.

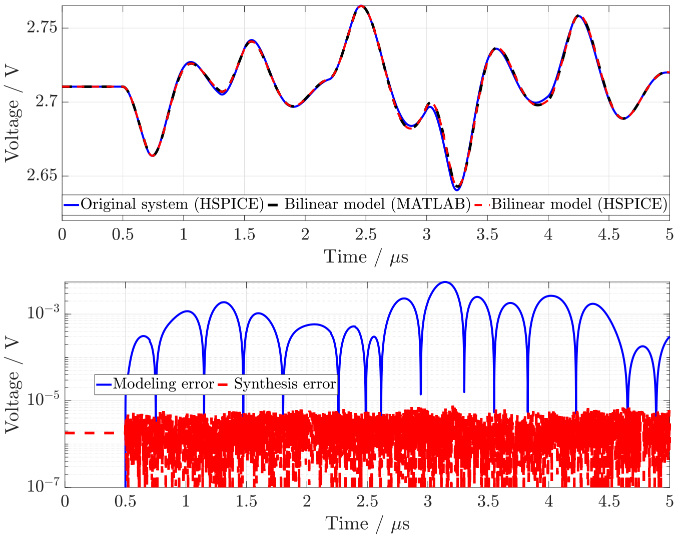

Fig. 10. Numerical validation of circuit synthesis (example in Sec. IV-A). Top: load voltage responses. Bottom: error between true response and synthesized bilinear macromodel (blue line), and between the MATLAB and HSPICE simulations of the same bilinear macromodel (red line).

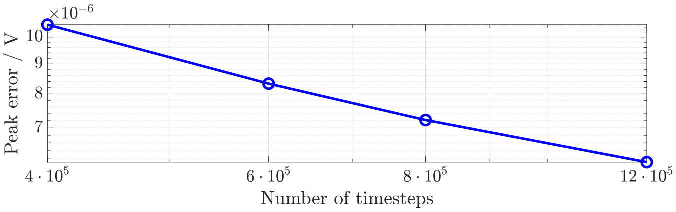

Fig. 11. Error between MATLAB and HSPICE responses of the nonlinear macromodel as the MATLAB timestep is decreased.

TABLE I

COMPONENT VALUES DEFINING THE CAPACITIVE LOAD OF SEC. IV-B IN FIVE DIFFERENT EXPERIMENTS.

Experiment #	C_l	R_s	G_p
1	1 nF	$1.83\,\Omega$	$18.3\,\mu$S
2	10 nF	$2.44\,\Omega$	$16.9\,\mu$S
3	50 nF	$1.00\,\Omega$	$14.9\,\mu$S
4	100 nF	$1.60\,\Omega$	$14.3\,\mu$S
5	200 nF	$1.29\,\Omega$	$13.0\,\mu$S

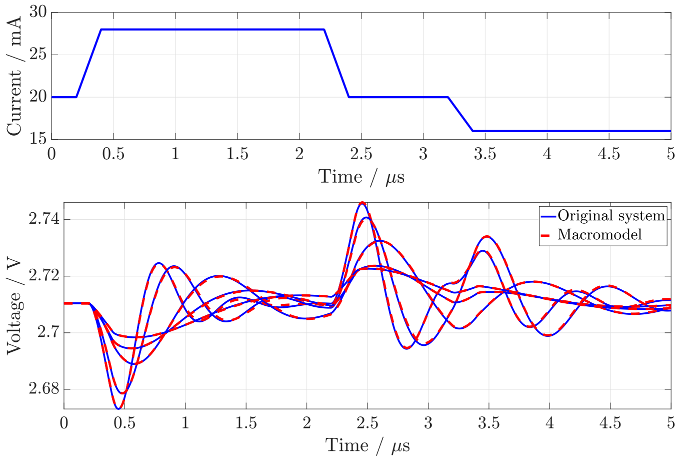

Fig. 12. Example in Sec. IV-B with varying load. Top: input current signal. Bottom: load voltage response computed in HSPICE.

comparison between the initial data and the reduced model for the GTF $H_2(\mathrm{j}\omega_1, \mathrm{j}\omega_2)$ of degree $m = 2$, which is displayed as a function of the first variable $\omega_1 = 2\pi f_1$ for several fixed values of ω_2.

As reported in [24], this model can be simulated in the time domain for any given current input $u(t)$ using an extension of the recursive convolution method, here implemented in MATLAB with a fixed timestep. The synthesis method proposed herein was also applied to turn the bilinear macromodel into an electrical network described by a SPICE-compatible netlist. Using the input current reported in Fig. 9, the macromodel response was found using both MATLAB and HSPICE (using the synthesized network), with the results reported in Fig. 10 (top). The synthesized netlist is an exactly equivalent representation of the same model, implying that any deviation between the response computed in MATLAB [black dashed line in Fig. 10(top)] and the synthesized network (red dashed line) is due to numerical mismatch stemming from the fact that different time-domain integration methods are used in these two cases. Moreover, HSPICE results are given with a finite precision of six significant digits. The error between MATLAB and HSPICE is reported by the red curve in Fig. 10(bottom). This *synthesis error* is at most $6.1 \cdot 10^{-6}$ V. If the number of timesteps used for MATLAB solution is increased from $40 \cdot 10^4$ to $120 \cdot 10^4$, this error steadily decreases as reported in Fig. 11. The observed convergence and the fact that the error is of the same order of magnitude as the accuracy of HSPICE output demonstrate that the synthesis method is correct, and the observed *synthesis error* is just a numerical integration error.

Figure 10 also compares the responses of the original system and of the synthesized macromodel, which differ because of the approximation entailed by the macromodeling method. Nonetheless, the macromodel provides an accurate response [see Fig. 10 (top)], with a peak error of 5 mV.

B. Parametric LDO Study with Load Variations

After validating the proposed synthesis method in Section IV-A, we now consider a realistic use case where the synthesized network described in Section IV-A is instantiated in a simulation environment connected to a load. The load consists of a capacitance C_l, with an Equivalent Series Resistance (ESR) R_s and a shunt conductance G_p. Five different combinations of these parameter values are considered, as reported in Table I. For each combination, the load voltage response to the input current signal reported in Fig. 12(top) is found by running HSPICE transient simulations using the original transistor-level system and the synthesized macromodel. Figure 12(bottom) compares the corresponding voltage responses for all five load combinations.

The macromodel produces a close approximation of the reference system behavior even when it is connected to these capacitive loads, and it can be solved in a fraction of the time. In fact, while the HSPICE simulation including the transistor-level reference circuit takes 5.36 s, the simulation that uses

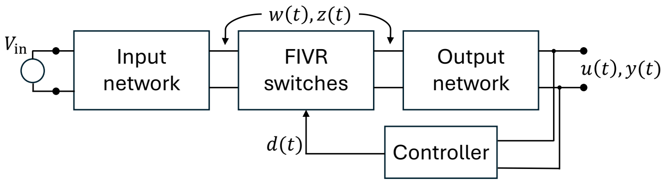

Fig. 13. Architecture of a Power Delivery Network with Integrated Regulators.

the synthesized macromodel only takes 0.45 s (for the same number of timesteps), leading to a speedup factor of nearly 12×.

C. A Power Delivery Network with Integrated Regulators

Modern multi-core microprocessors are powered by PDNs that include Integrated Voltage Regulators (IVRs) [1] to improve power efficiency and enable fine-grained supply voltage management. These consist of an array of switching regulators built inside the package, so as to have an independent regulator for each processor core. Power Integrity verification in these modern PDN architectures is carried out through system-level time-domain simulations. The latter involves a high computational complexity that can be tackled using compact reduced models [2].

A high-level view of the topology of the considered PDN architecture is depicted in Fig. 13. The input network block represents the main PCB and the package with its decoupling capacitors. The output network includes passive components of the output regulator filter and a model of the on-die power grid. Feedback control by means of compensators is used to regulate the load voltage. A complete description of this class of systems, including a precise mathematical formalization, is available in [2].

Similarly to the previous example, these PDNs provide supply voltage to other digital circuits that draw current from the output port (see Fig. 13). In this section, we describe a macromodel of the relation between the load current $i(t)$, chosen as the input signal $u(t) \triangleq i_L(t)$, and the load voltage (i.e., $y(t) \triangleq v(t)$). For simplicity, we focus on a system with a single output port, obtained by considering only a single core of an originally multi-core PDN. The reference model of the PDN is available as an HSPICE netlist (courtesy of Intel Corp.), where detailed circuit models of all blocks are instantiated together with a time-averaged model of the regulator switches.

The nonlinear macromodeling procedure of [24] was applied to build a quadratic model ($M = 2$) starting from $K_1 = 1000$, $K_2 = 20000$ GTF samples. This data was used to fit a multivariate rational model with $\nu_1 = \nu_2 = 9$ poles. The operating point selected to build this model is defined by a nominal supply voltage $V_{\text{in}} = 1.8\,\text{V}$ and a stationary load current $\bar{i}_L = 5\,\text{A}$. In these conditions, the regulated nominal output voltage is $\bar{y} = 0.696$ V. The nonlinear macromodel was synthesized as a SPICE-compatible netlist to enable transient analysis via HSPICE following the approach described herein.

Time-domain validation was carried out using the load current signal depicted in Fig. 14. In Fig. 15(top), the response of the reference netlist is compared with the nonlinear macromodel, both solved in HSPICE. The peak error observed in this simulation is only about 3.04 mV, over a voltage swing of more than 100 mV.

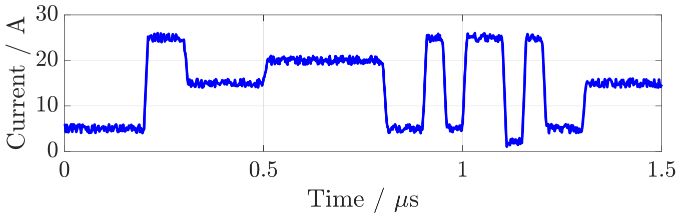

Fig. 14. Load current signal used to validate the example in Sec. IV-C.

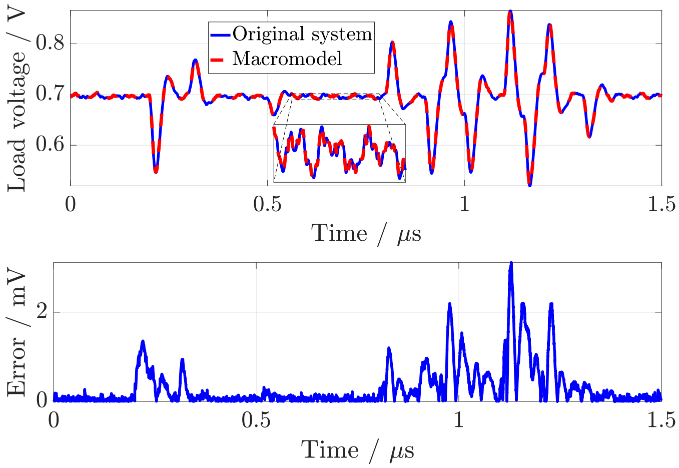

Fig. 15. Time-domain validation of the PDN example in Sec. IV-C, carried out via HSPICE. Top: load voltage response. Bottom: Error between the original system and the nonlinear macromodel.

For this realistic system, using a nonlinear macromodel proves to be much more efficient than the original simulation. In fact, the simulation runtime to solve the reference HSPICE netlist is 1592 s, while the macromodel only takes 2.46 s, corresponding to a speedup of nearly 650×.

V. CONCLUSIONS

This paper proposed a SPICE-compatible synthesis of nonlinear macromodels in bilinear state-space form. Such a model format is appropriate for the representation of weakly nonlinear systems, a class that includes several structures of practical interest, such as system-level power delivery networks equipped with integrated voltage regulators represented by time-averaged models. With the proposed synthesis, this paper completes a framework for data-driven nonlinear macromodeling. Overall, this framework can be interpreted as a general extension of the well-known Vector Fitting algorithm to the nonlinear setting.

The results in this work call for future extensions in three main directions. First, a generalization to multi-input, multi-output nonlinear structures is required to extend the scope and applicability. Second, an attempt to extend this to stronger types of nonlinearities, for which the formalism of Volterra series might lead to overly complex and non-scalable representations, is highly desired. Note that the theoretical tool of the Volterra series does not present fundamental limitations that prevent its application to strongly nonlinear systems. Also, the modeling procedure used here can, in principle, scale to a higher Volterra order M. However, the bottlenecks that have to be overcome with large M are the difficulty of practically sampling high-degree GTFs (M-dimensional domain) and the exponential scaling of model size with M. Finally, a thorough theoretical study on the stability properties of the proposed macromodels when interconnected with other subsystems is needed. In fact, whereas passivity of linear macromodels provides a sufficient condition for global stability, global stability conditions for nonlinear systems are not generally available, considering that individual nonlinear circuit blocks that are of interest for the proposed behavioral macromodeling flow may not be passive themselves. Although no stability issues were encountered with the documented examples, this point surely deserves more attention and is reserved for future investigations.

REFERENCES

[1] K. Radhakrishnan, M. Swaminathan, and B. K. Bhattacharyya, "Power delivery for high-performance microprocessors—challenges, solutions, and future trends," *IEEE Transactions on Components, Packaging and Manufacturing Technology*, vol. 11, no. 4, pp. 655–671, 2021.

[2] A. Carlucci, S. Grivet-Talocia, S. Kulasekaran, and K. Radhakrishnan, "Structured model order reduction of system-level power delivery networks," *IEEE Access*, vol. 12, pp. 18 198–18 214, 2024.

[3] S. Grivet-Talocia and B. Gustavsen, *Passive Macromodeling: Theory and Applications*. New York: John Wiley and Sons, 2016 (published online on Dec 7, 2015).

[4] B. Gustavsen and A. Semlyen, "Rational approximation of frequency domain responses by vector fitting," *Power Delivery, IEEE Transactions on*, vol. 14, no. 3, pp. 1052–1061, jul 1999.

[5] D. Deschrijver, M. Mrozowski, T. Dhaene, and D. De Zutter, "Macromodeling of multiport systems using a fast implementation of the vector fitting method," *Microwave and Wireless Components Letters, IEEE*, vol. 18, no. 6, pp. 383–385, june 2008.

[6] S. Ganeshan, N. K. Elumalai, R. Achar, and W. K. Lee, "Gvf: Gpu-based vector fitting for modeling of multiport tabulated data networks," *IEEE Transactions on Components, Packaging and Manufacturing Technology*, vol. 10, no. 8, pp. 1375–1387, 2020.

[7] D. Deschrijver, B. Haegeman, and T. Dhaene, "Orthonormal vector fitting: A robust macromodeling tool for rational approximation of frequency domain responses," *Advanced Packaging, IEEE Transactions on*, vol. 30, no. 2, pp. 216–225, may 2007.

[8] A. Chinea and S. Grivet-Talocia, "On the parallelization of vector fitting algorithms," *IEEE Transactions on Components, Packaging and Manufacturing Technology*, vol. 1, no. 11, pp. 1761–1773, November 2011.

[9] S. Grivet-Talocia and M. Bandinu, "Improving the convergence of vector fitting for equivalent circuit extraction from noisy frequency responses," *IEEE Trans. Electromagnetic Compatibility*, vol. 48, no. 1, pp. 104–120, February 2006.

[10] B. Gustavsen and A. Semlyen, "Enforcing passivity for admittance matrices approximated by rational functions," *Power Engineering Review, IEEE*, vol. 21, no. 2, p. 54, feb. 2001.

[11] C. P. Coelho, J. Phillips, and L. M. Silveira, "A convex programming approach for generating guaranteed passive approximations to tabulated frequency-data," *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol. 23, no. 2, pp. 293 – 301, feb. 2004.

[12] T. Dhaene, D. Deschrijver, and N. Stevens, "Efficient algorithm for passivity enforcement of S-parameter-based macromodels," *Microwave Theory and Techniques, IEEE Transactions on*, vol. 57, no. 2, pp. 415–420, Feb 2009.

[13] S. Grivet-Talocia and A. Ubolli, "A comparative study of passivity enforcement schemes for linear lumped macromodels," *IEEE Trans. Advanced Packaging*, vol. 31, no. 4, pp. 673–683, Nov 2008.

[14] S. Grivet-Talocia, "Passivity enforcement via perturbation of Hamitonian matrices," *IEEE Trans. Circuits and Systems I: Fundamental Theory and Applications*, vol. 51, no. 9, pp. 1755–1769, September 2004.

[15] G. Antonini, "SPICE equivalent circuits of frequency-domain responses," *Electromagnetic Compatibility, IEEE Transactions on*, vol. 45, no. 3, pp. 502–512, Aug 2003.

[16] Z. Qi, H. Yu, P. Liu, S.-D. Tan, and L. He, "Wideband passive multiport model order reduction and realization of rlcm circuits," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 25, no. 8, pp. 1496–1509, 2006.

[17] S. Grivet-Talocia, G. Signorini, S. B. Olivadese, C. Siviero, and P. Brenner, "Thermal noise compliant synthesis of linear lumped macromodels," *Components, Packaging and Manufacturing Technology, IEEE Transactions on*, vol. 5, no. 1, pp. 75–85, Jan 2015.

[18] "IBIS (I/O buffer information specification)," accessed on 20 June 2025. [Online]. Available: https://ibis.org

[19] I. S. Stievano, I. A. Maio, and F. G. Canavero, "MπLog, macromodeling via parametric identification of logic gates," *Advanced Packaging, IEEE Transactions on*, vol. 27, no. 1, pp. 15–23, 2004.

[20] G. Signorini, C. Siviero, S. Grivet-Talocia, and I. S. Stievano, "Macromodeling of i/o buffers via compressed tensor representations and rational approximations," *IEEE Transactions on Components, Packaging and Manufacturing Technology*, vol. 6, no. 10, pp. 1522–1534, 2016.

[21] T. Zhu, M. B. Steer, and P. D. Franzon, "Accurate and scalable io buffer macromodel based on surrogate modeling," *IEEE Transactions on Components, Packaging and Manufacturing Technology*, vol. 1, no. 8, pp. 1240–1249, 2011.

[22] W. Rugh, *Nonlinear System Theory: The Volterra/Wiener Approach*, ser. Johns Hopkins series in information sciences and systems. Johns Hopkins University Press, 1981.

[23] S. Boyd, L. Chua, and C. A. Desoer, "Analytical Foundations of Volterra Series," *IMA Journal of Mathematical Control and Information*, vol. 3, pp. 243–282, 1984.

[24] A. Carlucci, I. V. Gosea, and S. Grivet-Talocia, "Data-driven modeling of weakly nonlinear circuits via generalized transfer function approximation," *IEEE Access*, vol. 13, pp. 2746–2762, 2025.

[25] ——, "An extension of vector fitting to weakly nonlinear circuits," in *2025 IEEE 29th Workshop on Signal and Power Integrity (SPI)*, 2025, pp. 1–4.

[26] J. Debnath and R. Dahiya, "Theorems on multidimensional laplace transform for solution of boundary value problems," *Computers & Mathematics with Applications*, vol. 18, no. 12, pp. 1033–1056, 1989.

[27] D. Root, J. Verspecht, D. Sharrit, J. Wood, and A. Cognata, "Broad-band poly-harmonic distortion (phd) behavioral models from fast automated simulations and large-signal vectorial network measurements," *IEEE Transactions on Microwave Theory and Techniques*, vol. 53, no. 11, pp. 3656–3664, 2005.

[28] D. Root, J. Verspecht, J. Horn, and M. Marcu, *X-Parameters: Characterization, Modeling, and Design of Nonlinear RF and Microwave Components*, ser. The Cambridge RF and Microwave Engineering Series. Cambridge University Press, 2013.

[29] 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 Transactions on Components, Packaging and Manufacturing Technology*, vol. 7, no. 1, pp. 39–49, 2017.

[30] S. Boyd and L. Chua, "Fading memory and the problem of approximating nonlinear operators with Volterra series," *IEEE Transactions on Circuits and Systems*, vol. 32, no. 11, pp. 1150–1161, 1985.

[31] I. W. Sandberg, "Expansions for nonlinear systems," *Bell System Technical Journal*, vol. 61, no. 2, pp. 159–199, 1982. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/j.1538-7305.1982.tb00331.x

[32] T. Breiten and T. Damm, "Krylov subspace methods for model order reduction of bilinear control systems," *Systems & Control Letters*, vol. 59, no. 8, pp. 443–450, 2010.

[33] C. Gu, "QLMOR: a projection-based nonlinear model order reduction approach using quadratic-linear representation of nonlinear systems," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 30, no. 9, pp. 1307–1320, 2011.

[34] I. V. Gosea, "Exact and inexact lifting transformations of nonlinear dynamical systems: Transfer functions, equivalence, and complexity reduction," *Applied Sciences*, vol. 12, no. 5, 2022.

[35] 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*, ser. International Series of Numerical Mathematics. Birkhäuser, Cham, 2021, vol. 171, pp. 3–30.

[36] A. Carlucci and S. Grivet-Talocia, "Nonlinear macromodeling of voltage-regulated power delivery networks," in *2024 IEEE 33rd Con-*

ference on Electrical Performance of Electronic Packaging and Systems (EPEPS), 2024, pp. 1–3.

[37] M. Schoukens and K. Tiels, "Identification of block-oriented nonlinear systems starting from linear approximations: A survey," *Automatica*, vol. 85, pp. 272–292, 2017. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0005109817303990

[38] B. Gustavsen and A. Semlyen, "Rational approximation of frequency domain responses by vector fitting," *IEEE Transactions on Power Delivery*, vol. 14, no. 3, pp. 1052–1061, 1999.

[39] B. D. O. Anderson and S. Vongpanitlerd, *Network analysis and synthesis*. Prentice-Hall, 1973.

[40] V. Belevitch, *Classical network theory*. Holden-Day, 1968.

[41] T. A. Davis and E. Palamadai Natarajan, "Algorithm 907: Klu, a direct sparse solver for circuit simulation problems," *ACM Trans. Math. Softw.*, vol. 37, no. 3, Sep. 2010.

[42] X. S. Li, "An overview of superlu: Algorithms, implementation, and user interface," *ACM Trans. Math. Softw.*, vol. 31, no. 3, p. 302–325, Sep. 2005. [Online]. Available: https://doi.org/10.1145/1089014.1089017

[43] N. Rai. (2025) A technical survey of sparse linear solvers in electronic design automation. [Online]. Available: https://arxiv.org/abs/2504.11716

[44] T. A. Davis, *Direct Methods for Sparse Linear Systems*. Society for Industrial and Applied Mathematics, 2006.

[45] R. J. Milliken, J. Silva-Martinez, and E. Sanchez-Sinencio, "Full on-chip CMOS low-dropout voltage regulator," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 54, no. 9, pp. 1879–1890, 2007.

[46] S. Boyd, Y. Tang, and L. Chua, "Measuring Volterra kernels," *IEEE Transactions on Circuits and Systems*, vol. 30, no. 8, pp. 571–577, 1983.

[47] P. Goyal, M. I. Ahmad, and P. Benner, "Model reduction of quadratic-bilinear descriptor systems via Carleman bilinearization," in *2015 European Control Conference (ECC)*, 2015, pp. 1177–1182.

APPENDIX

A. Bilinear lifting of voltage-regulated PDN

In this additional section, we provide a practical demonstration of Carleman's technique to obtain a bilinear approximation of a more generic nonlinear system [22]. We consider the voltage-regulated PDN depicted in Fig. 13, for which a bilinear model was identified from frequency responses and validated in Sec. IV-C. The purpose of the following derivation is to demonstrate that the bilinear model structure, although very simple, is indeed appropriate to model regulated PDNs. The proof is conducted starting from first-principle equations (formulated following [2]) and showing that these can indeed be cast in a bilinear form.

The exact equations of the system can be derived by first introducing two vectors of auxiliary variables $\boldsymbol{w}$, $\boldsymbol{z}$ containing the voltages and currents at the interface of the middle block (*FIVR switches*), see Fig. 13. This block is controlled by the duty cycle signal $d(t)$ produced by the feedback compensator. If a low-frequency averaged model is adopted for the FIVR switches, the characteristic equation for this block can be compactly written as

$$\boldsymbol{w}(t) = \boldsymbol{\Delta}\boldsymbol{z}(t)d(t), \tag{25}$$

where matrix $\boldsymbol{\Delta}$ is constant.

The input network, the output network, and the compensator are linear systems including electromagnetic models of board and package, decoupling capacitors, and an error amplifier (e.g., type-III). As such, they can be collectively represented by a linear state-space system

$$\dot{\boldsymbol{x}} = \mathbf{A}\boldsymbol{x} + \mathbf{B}_w\boldsymbol{w} + \mathbf{B}_u u \tag{26a}$$

$$\boldsymbol{z} = \mathbf{C}_z\boldsymbol{x} + \mathbf{D}_{zw}\boldsymbol{w} + \mathbf{D}_{zu} u \tag{26b}$$

$$d = \mathbf{C}_d\boldsymbol{x} + \mathbf{D}_{dw}\boldsymbol{w} + \mathbf{D}_{du} u \tag{26c}$$

$$y = \mathbf{C}_y\boldsymbol{x} + \mathbf{D}_{yw}\boldsymbol{w} + \mathbf{D}_{yu} u \tag{26d}$$

Note that, as (25) is used to replace $\boldsymbol{w}$ in (26), the resulting system is nonlinear because of the products $d(t)\boldsymbol{z}(t)$. More precisely, a *quadratic* structure is revealed by introducing the following extended state vector

$$\boldsymbol{\xi} = \begin{pmatrix} \boldsymbol{x} \\ \boldsymbol{z} \\ d \end{pmatrix} \tag{27}$$

and recasting (26a)-(26c) in the form

$$\mathcal{E}\dot{\boldsymbol{\xi}} = \mathcal{A}\boldsymbol{\xi} + \mathcal{Q}\boldsymbol{\xi}\otimes\boldsymbol{\xi} + \mathcal{B}u \tag{28}$$

where $\mathcal{E} = \text{blkdiag}\{\mathbb{I}, 0, 0\}$ is a concatenation of diagonal blocks of sizes compatible with $\boldsymbol{x}$, $\boldsymbol{z}$, d, and

$$\mathcal{A} = \begin{pmatrix} \mathbf{A} & 0 & 0 \\ \mathbf{C}_z & -\mathbb{I} & 0 \\ \mathbf{C}_d & 0 & -\mathbb{I} \end{pmatrix}. \tag{29}$$

To apply Carleman's linearization to (28), let us start by defining $\boldsymbol{\xi}^{(2)} = \boldsymbol{\xi}\otimes\boldsymbol{\xi}$, so that

$$\dot{\boldsymbol{\xi}}^{(2)} = \dot{\boldsymbol{\xi}}\otimes\boldsymbol{\xi} + \boldsymbol{\xi}\otimes\dot{\boldsymbol{\xi}} \tag{30}$$

With this definition, (28) becomes

$$\mathcal{E}\dot{\boldsymbol{\xi}} = \mathcal{A}\boldsymbol{\xi} + \mathcal{Q}\boldsymbol{\xi}^{(2)} + \mathcal{B}u \tag{31}$$

Using the notation $\mathcal{E}^{(2)} \triangleq \mathcal{E}\otimes\mathcal{E}$, the dynamic evolution of the new variable $\boldsymbol{\xi}^{(2)}$ is governed by

$$\begin{aligned} &\mathcal{E}^{(2)}\dot{\boldsymbol{\xi}}^{(2)} = \mathcal{E}\dot{\boldsymbol{\xi}}\otimes\mathcal{E}\boldsymbol{\xi} + \mathcal{E}\boldsymbol{\xi}\otimes\mathcal{E}\dot{\boldsymbol{\xi}} = \\ &(\mathcal{A}\boldsymbol{\xi} + \mathcal{Q}\boldsymbol{\xi}^{(2)} + \mathcal{B}u)\otimes\mathcal{E}\boldsymbol{\xi} + \mathcal{E}\boldsymbol{\xi}\otimes(\mathcal{A}\boldsymbol{\xi} + \mathcal{Q}\boldsymbol{\xi}^{(2)} + \mathcal{B}u) = \\ &\underbrace{(\mathcal{A}\otimes\mathcal{E} + \mathcal{E}\otimes\mathcal{A})}_{\triangleq\mathcal{A}^{(2)}}\boldsymbol{\xi}^{(2)} + \underbrace{(\mathcal{B}\otimes\mathcal{E} + \mathcal{E}\otimes\mathcal{B})}_{\triangleq\mathcal{N}^{(2)}}\boldsymbol{\xi}u + \\ &\underbrace{(\mathcal{Q}\otimes\mathcal{E} + \mathcal{E}\otimes\mathcal{Q})\boldsymbol{\xi}^{(3)}}_{\rho^{(3)}} \end{aligned} \tag{32}$$

where the symbol $\boldsymbol{\xi}^{(3)} \triangleq \boldsymbol{\xi}\otimes\boldsymbol{\xi}\otimes\boldsymbol{\xi}$ has been introduced. Note that a bilinear term $\boldsymbol{\xi}u$ appears in (32), as well as a higher-order *remainder* $\rho^{(3)}$. By combining Equations (31) and (32), we obtain

$$\begin{aligned} &\begin{pmatrix} \mathcal{E} & \\ & \mathcal{E}^{(2)} \end{pmatrix} \begin{pmatrix} \dot{\boldsymbol{\xi}} \\ \dot{\boldsymbol{\xi}}^{(2)} \end{pmatrix} = \begin{pmatrix} \mathcal{A} & \mathcal{Q} \\ 0 & \mathcal{A}^{(2)} \end{pmatrix} \begin{pmatrix} \boldsymbol{\xi} \\ \boldsymbol{\xi}^{(2)} \end{pmatrix} + \\ &\begin{pmatrix} 0 & 0 \\ \mathcal{N}^{(2)} & 0 \end{pmatrix} \begin{pmatrix} \boldsymbol{\xi} \\ \boldsymbol{\xi}^{(2)} \end{pmatrix} u + \begin{pmatrix} 0 \\ \rho^{(3)} \end{pmatrix} \end{aligned} \tag{33}$$

Except for the remainder $\rho^{(3)}$, the result is a bilinear system in the augmented state vector $(\boldsymbol{\xi};\, \boldsymbol{\xi}^{(2)})$. Note that Carleman's linearization for a quadratic system in descriptor format was also addressed in [47] with the same result as (33).

The process can continue by further extending the state vector by adding $\boldsymbol{\xi}^{(3)}$, together with another equation describing the evolution of $\boldsymbol{\xi}^{(3)}$ that is obtained by taking its derivative $\dot{\boldsymbol{\xi}}^{(3)} = \dot{\boldsymbol{\xi}}\otimes\boldsymbol{\xi}\otimes\boldsymbol{\xi} + \ldots$ in the same manner as above. This will again yield a bilinear system with an additional higher-order remainder. It can be proven that stopping the procedure at step m leads to a bilinear model whose Volterra kernels match exactly those of the reference system up to degree m. More details can be found in [22].

Antonio Carlucci (Member, IEEE) received the B.Sc. and M.Sc. degrees in Electronic Engineering in 2019 and 2021. In 2025, he obtained the Ph.D. degree from Politecnico di Torino, Italy. His research focuses on reduced-order modeling of electronic systems. He received the 2023 and 2025 Best Student Paper Awards of the IEEE Workshop on Signal and Power Integrity, and the 2024 Best Student Paper Award of the 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, the M.Sc. degree in electrical engineering, and the Ph.D. in electrical engineering under the guidance of Prof. Athanasios C. Antoulas from Jacobs University Bremen, Germany, in 2011, 2013, and 2017, respectively. He has been affiliated since February 2017 with the Max Planck Institute for Dynamics of Complex Technical Systems in Magdeburg, Germany. He was part of the DRI group (until October 2022), and then afterward of the Computational Methods in Systems and Control Theory (CSC) group led by Prof. Peter Benner. His research interests include model reduction and reduced-order modeling, data-driven approaches, numerical linear algebra, physics-based surrogate modeling, and enabling digital twins for process engineering and (electro) chemical applications. He received the best poster award at the 11th Conference of Applied Mathematics and Scientific Computing, Brijuni, Croatia, 2022.

Stefano Grivet-Talocia (M'98–SM'07–F'18) 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, where He serves as Head of the Doctoral School. He co-founded the academic spinoff company IdemWorks (Turin, Italy) in 2007, serving as the President until its acquisition by CST in 2016. He has authored more than 250 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. Dr. Grivet-Talocia was a co-recipient of the 2007 Best Paper Award of the IEEE TRANSACTIONS ON ADVANCED PACKAGING. He received the IBM Shared University Research Award in 2007, 2008, and 2009 and an Intel Strategic Research Segment Grant in 2022, 2023, and 2024. He was an Associate Editor of the IEEE TRANSACTIONS ON ELECTROMAGNETIC COMPATIBILITY from 1999 to 2001 and He is currently serving as Associate Editor for the IEEE TRANSACTIONS ON COMPONENTS, PACKAGING AND MANUFACTURING TECHNOLOGY. 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.