A Compressed Multivariate Macromodeling Framework for Fast Transient Verification of System-Level Power Delivery Networks

A Compressed Multivariate Macromodeling Framework for Fast Transient Verification of System-Level Power Delivery Networks

Antonio Carlucci, *Graduate Student Member, IEEE*, Tommaso Bradde, *Member, IEEE*, Stefano Grivet-Talocia, *Fellow, IEEE*, Scott Mongrain, Sid Kulasekaran, and Kaladhar Radhakrishnan, *Senior Member, IEEE*

***Abstract*— This article discusses a reduced-order modeling and simulation approach for fast transient power integrity verification at full system level. The reference structure is a complete power distribution network (PDN) from platform voltage regulator module (VRM) to multiple cores, including board, package, decoupling capacitors, and per-core fully integrated voltage regulators (FIVRs). All blocks are characterized and known through high-fidelity models derived from first-principle solvers (full-wave electromagnetic and circuit-level extractions). The complexity of such detailed characterization grows very large and becomes intractable, especially for power integrity verification of massive multicore platforms subjected to real workload scenarios. We approach this problem by exploiting a multistage macromodeling and compression process, leading to a compact representation of the system dynamics in terms of a linearized state-space structure with multiple feedback loops from the FIVR controllers. The PDN macromodel is obtained through a data-driven approach starting from reference small-signal frequency responses, obtaining a sparse and structured representation specifically designed to match the behavior of the reference system. The resulting compact model is then solved in time-domain very efficiently. Results on Mobile and enterprise Server benchmarks demonstrate a speedup in runtime up to 50× with respect to HSPICE, with negligible loss of accuracy.**

***Index Terms*— Fully-integrated voltage regulator (FIVR), macromodeling, multi-core architecture, power distribution network (PDN), power integrity, singular value decomposition, transient analysis, vector fitting.**

I. Introduction

As microprocessors power levels continue to rise, power delivery architects are increasingly relying on integrated voltage regulators (IVRs) to provide fine grain regulation at the chip level without increasing the complexity of the power delivery solution on the platform [1]. It is not uncommon for datacenter microprocessors to have over a hundred cores each with a dedicated IVR programed to deliver the minimum required voltage to support the frequency of operation for the core. One consequence of such a power delivery architecture is the coupling in power delivery noise from one core to another due to the shared input network. This in turn drives the need to develop a simulation framework that can run large-scale transient simulations for a number of different workloads to ensure the integrity of the power supply seen by the transistors. Early simulation techniques for analyzing power delivery noise have either ignored the coupling from one core to another or used a brute-force SPICE-based approach which is feasible only for small-scale low-core count microprocessors. This approach does not have the potential to scale to more complex systems with a large core count, as demanded by the state-of-the-art datacenter CPUs.

This article addresses system level power integrity verification for multicore microprocessors with Fully IVRs (FIVRs) [2] providing per-core voltage domain granularity. The input power supply to the FIVRs is generated from a single voltage regulator on the motherboard. Our objective is to simulate an entire power delivery network structured as a cascade of multiple stages between the main supply and the compute domains inside the microprocessor. In particular, we address the transient solution of the complete power delivery, including voltage regulation effects provided by FIVR as well as the coupling from one core to another. The large-scale nature of this simulation problem, both in terms of expected dynamic order and number of ports/signals to be evaluated, combined with the nonlinear FIVR circuitry and the associated feedback regulation loops, make this problem particularly challenging. A novel reduced order modeling approach is used to achieve significant speed up over the SPICE-based approach with minimal impact to the accuracy of the results.

II. Notation and Problem Statement

We consider the general system topology depicted in Fig. 1. At the motherboard level, the on-board voltage regulator module (VRM) is connected to the printed circuit board (PCB) power planes providing power up to the power pins

Manuscript received 10 February 2023; revised 29 May 2023; accepted 20 June 2023. Date of publication 5 July 2023; date of current version 20 October 2023. The work of Antonio Carlucci, Tommaso Bradde, and Stefano Grivet-Talocia was supported by Intel Corporation under the 2022 Intel SRS Grant titled as "API-S: Accelerated system-level transient Power Integrity Solvers." Recommended for publication by Associate Editor Z. Peng upon evaluation of reviewers' comments. *(Corresponding author: Antonio Carlucci.)*

Antonio Carlucci, Tommaso Bradde, and Stefano Grivet-Talocia are with the Department of Electronics and Telecommunications, Politecnico di Torino, 10129 Turin, Italy (e-mail: antonio.carlucci@polito.it; tommaso.bradde@polito.it; stefano.grivet@polito.it).

Scott Mongrain, Sid Kulasekaran, and Kaladhar Radhakrishnan are with Intel Corporation, Chandler, AZ 85226 USA (e-mail: scott.mongrain@intel.com; siddharth.kulasekaran@intel.com; kaladhar.radhakrishnan@intel.com).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCPMT.2023.3292449.

Digital Object Identifier 10.1109/TCPMT.2023.3292449

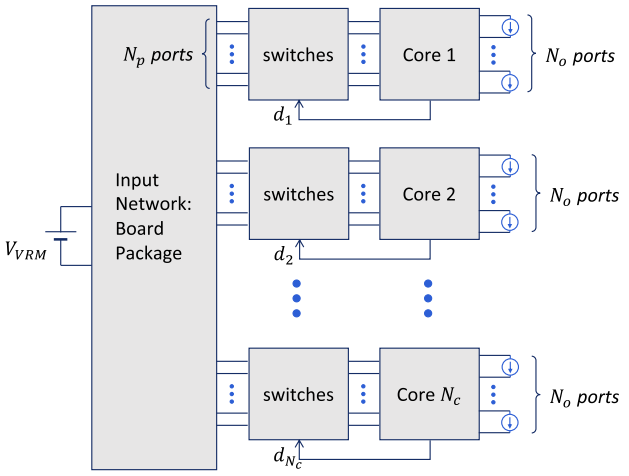


Fig. 1. Schematic of the multicore power distribution system under investigation.

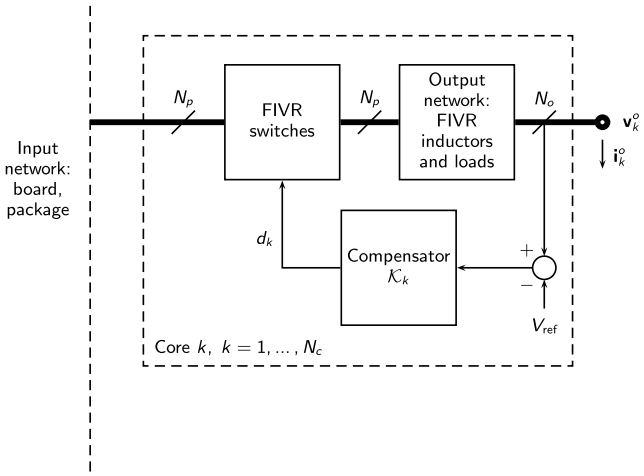


Fig. 2. Structure of the power distribution system for each of the N_c cores whose voltage is regulated by N_p-phase FIVRs.

of the microprocessor package. Both package and board are known through a high-accuracy electromagnetic characterization, in terms of multiport sampled scattering responses. We also assume that a full set of decoupling capacitors has already been optimized at an earlier stage, in order to meet the required target impedance specifications [3], [4]. Adopting a conventional RLC model (possibly through multiple parallel RLC branches) for each capacitor, we embed all such models by terminating the corresponding ports, obtaining a port-reduced subsystem that is interfaced on one end with the system VRM and on the other hand with the FIVR switches that provide the interface with the individual cores. This subsystem can be collectively represented by a large-scale distributed linear and time-invariant (LTI) multiport described by a transfer matrix $\mathbf{H}_b(s)$ and denoted as *input network*. As depicted in Fig. 1, this subsystem provides a global coupling path between all cores. In fact, one of the objectives of this investigation is to assess the contribution of such couplings in a real workload scenario where individual cores are excited by specific transient loading conditions.

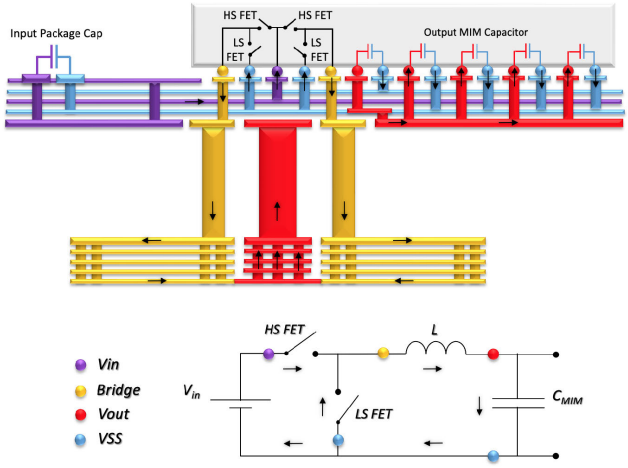


Fig. 3. Detailed view of a buck FIVR implementation: inductors are placed in the package, while switching circuitry with MIM capacitors are integrated ON-chip.

TABLE I
NUMBER OF CORES (N_c), FIVR PHASES (N_p), OUTPUTS PER CORE (N_o), AND TOTAL OUTPUTS $P = N_c N_o$ FOR MOBILE AND SERVER EXAMPLES

	N_c	N_p	N_o	P
Mobile	4	4	36	144
Server	8	3	57	456
Server	12	3	57	684
Server	16	3	57	912

Fig. 2 provides a more detailed view of the power distribution network (PDN) for the kth core. Inside the chip, a second voltage regulation stage is implemented through FIVRs, consisting of multiphase switching power supplies (e.g., buck converters). Voltage regulation is achieved by sensing the output voltage, comparing its instantaneous value to a reference voltage V_{ref}, and feeding the corresponding error signal through a dedicated per-core controller or *compensator* $\mathcal{K}_k$. The output of this controller is a duty-cycle signal d_k which drives the FIVR switching banks. Fig. 3 provides additional details of the FIVR structure as implemented on hardware. All the switching circuitry including power transistors, switching control circuits, and the output decoupling for these FIVRs are fabricated on-die, whereas the inductors are placed in the package. The FIVR output is a filtered and regulated voltage that is distributed through the die power rails to reach logic devices in their respective power domains. The blocks denoted as *output network* in Fig. 2 represent the PDN of each core through a circuit model, including integrated MIM capacitors which provide the output decoupling, plus a detailed electromagnetic model of the integrated inductors that complete the topology of the FIVRs. Also this output network can be represented as an LTI system with a transfer matrix $\mathbf{H}_c(s)$.

The interface signals between all blocks are defined by:

1) N_c: number of (identical) cores;
2) N_p: number of phases of each FIVR; and
3) N_o: number of output ports (per core).

These parameters are listed in Table I for the two benchmark examples that we will investigate in this work namely a

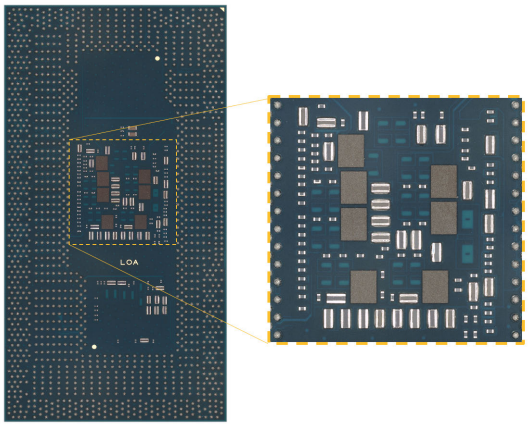


Fig. 4. Picture of a representative Intel Core microprocessor.

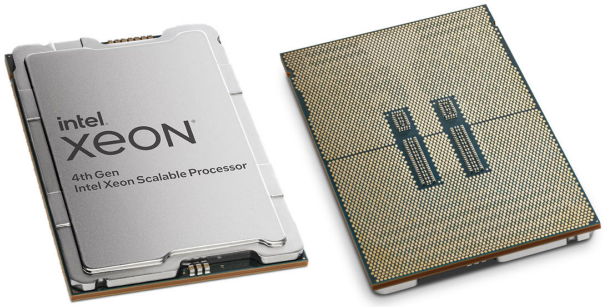


Fig. 5. Picture of a representative Intel Xeon microprocessor.

mobile system equipped with an Intel[1] Core[2] microprocessor (Fig. 4), and an enterprise server based on an Intel Xeon[1] microprocessor (Fig. 5), which we consider in three different configurations with $N_c = 8$, 12, and 16 cores. Therefore, the input network $\mathbf{H}_b(s)$ has $N_cN_p + 1$ ports, each core k is represented by a transfer function $\mathbf{H}_{c,k}(s)$ with N_p ports interfaced to the switches and N_o output ports where the transient voltage is to be computed, so that the overall output network $\mathbf{H}_c(s)$ has a total of $N_c(N_p + N_o)$ ports. The time-varying duty cycles of all cores are collected in the vector $\boldsymbol{d}(t) \in [0, 1]^{N_c}$. The main objective of this work is to compute efficiently the transient voltages $v^o_{k,n}(t)$ at all $n = 1, \ldots, N_o$ ports of each core $k = 1, \ldots, N_c$, excited by predefined current load signals $i^o_{k,n}(t)$ acting concurrently.

A. Preprocessing

The initial phase in our problem setup involves a preliminary macromodeling step applied to both input and output networks, based on the available electromagnetic solver data. After port termination with decoupling capacitor models, the resulting frequency responses are processed by a rational macromodeling engine based on vector fitting (VF) with passivity enforcement [5], [6], so that both $\mathbf{H}_b(s)$ and $\mathbf{H}_{c,k}(s)$ are available as a set of linear state-space equations and the

[1]Registered trademark.
[2]Trademarked.

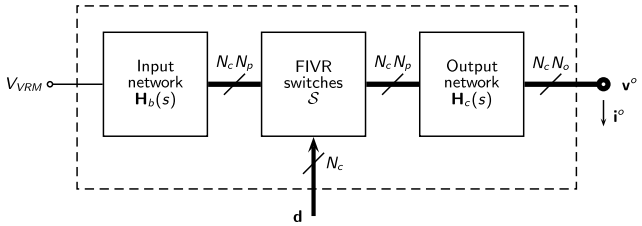


Fig. 6. Illustration of the open-loop PDN structure after removing the controller feedback loop of each FIVR.

associated synthesized SPICE realizations. Such macromodels enable a direct (reference) SPICE simulation of the complete system, once complemented with circuit models of the switches and the compensators. This will provide the solution that we will use as reference, both in terms of accuracy and runtime. Starting from this representation of the input and output networks as a set of linear state-space equations or, equivalently, their synthesized circuit realizations, frequency-domain samples of any network function associated with the full PDN system (e.g., the input impedance at the PDN load ports) can be obtained by means of AC analyses performed in a commercial circuit solver (SPICE) or by direct computation in MATLAB. In the following, these responses will be referred to as reference *data*.

III. Outlook

The proposed approach is based on a reduced-order representation of the open-loop dynamics of the PDN. If we remove the per-core feedback loops with the corresponding controllers, we obtain the structure depicted in Fig. 6, where the complete set of output networks of all FIVR switches and all cores are collected in the two macro-blocks $\mathcal{S}$ and $\mathbf{H}_c(s)$, respectively. The global structure enclosed in the dashed block represents a large-scale nonlinear time-varying dynamical system, where the large scale nature is induced by the large number of ports/signals and by the broad frequency bands over which both input and output network models are needed, and the nonlinear time-varying nature is induced only by the FIVR switches $\mathcal{S}$. The inputs to this structure are the (constant) VRM voltage source, all N_cN_o load currents on the core side, and all N_c duty-cycle signals.

One of the key aspects of proposed formulation is the adopted representation for the switches, here represented through averaged models. For each core k and phase j, the corresponding set of FIVR switches is represented by an ideal transformer with turn ratio $1{:}d_k(t)$, where $d_k(t)$ is the duty-cycle signal resulting from the compensator $\mathcal{K}_k$ of core k. This assumption has its own limitations but is known to be accurate when the buck converters operate in continuous conduction mode (CCM). Even with this simplifying assumption, the global open-loop system still remains nonlinear (the transformers couple input-output voltages and currents through a multiplication by d_k). In real operation, each $d_k = d_k(t)$ provides a time-varying nonlinearity. For this reason, first-principle circuit simulation (e.g., via HSPICE) is particularly time-consuming.

A general form of the open-loop PDN equations, e.g., as obtained by a standard modified nodal analysis (MNA), can be stated as

$$\begin{aligned} \dot{\boldsymbol{x}} &= \mathcal{F}(\boldsymbol{x}, \boldsymbol{i}^o, \boldsymbol{d}, V_{\text{VRM}}) \\ \boldsymbol{v}^o &= \mathcal{G}(\boldsymbol{x}, \boldsymbol{i}^o, \boldsymbol{d}, V_{\text{VRM}}) \end{aligned} \tag{1}$$

where $\boldsymbol{x}$ collects all required state variables. We propose an approximate and simplified representation of the dynamics as

$$\dot{\boldsymbol{x}} = \mathbf{A}\boldsymbol{x} + \mathbf{B}_1(\boldsymbol{d})\boldsymbol{i}^o + \mathbf{B}_2(\boldsymbol{i}^o)\boldsymbol{d} \tag{2}$$

with the contribution of different inputs separated and expressed in a linearized form, where:

1) the open-loop poles (eigenvalues of $\mathbf{A}$) are assumed constant and independent on $\boldsymbol{d}$;
2) the matrix $\mathbf{B}_1$ mapping the core current inputs is (nonlinearly) parameterized by the duty-cycle signals $\boldsymbol{d}$; we will see that this parameterization is essential for a correct representation of the voltage regulation dynamics induced by the feedback operation at runtime;
3) the matrix $\mathbf{B}_2$ is parameterized by the loading currents $\boldsymbol{i}^o$; we will, however, see that this dependence is very weak and can be ignored, so that $\mathbf{B}_2$ can be assumed as constant; and
4) the input V_{VRM} is embedded as a fixed value in the other matrix coefficients, since constant.

The above structure is in fact the result of several investigations and tests that were performed on the full system for both benchmark examples considered in this work. Part of these tests will be documented in Sections IV–VII in support of the derivations.

1) In Section IV, we consider all duty-cycle signals as "frozen," and we construct a reduced-order macromodel of the PDN as observed from the output ports. This will lead to a representation of the output impedance matrix $\mathbf{Z}(s; \boldsymbol{d})$ as a rational function including an explicit dependence on the operating point induced by the duty-cycle configuration. We will see that a common pole (non-parameterized) set to represent $\mathbf{Z}(s; \boldsymbol{d})$ is adequate, supporting a constant state matrix $\mathbf{A}$ in (2). Conversely, residue matrices need to be parameterized by $\boldsymbol{d}$, leading to the input map $\mathbf{B}_1(\boldsymbol{d})$ in (2) through a simple realization process.
2) In Section V, we consider the dynamics induced by the time-varying duty-cycle signals. Such dynamics will be characterized through a small-signal (linearized) approach, obtaining the second input contribution $\mathbf{B}_2$ in (2).
3) In Section VI, we will reintroduce the feedback loops for each core. The internal stability of the parameterized macromodel under closed-loop operation will be analyzed based on the assumed model topology. Finally, time discretization will be introduced to enable transient analysis.
4) Section VII will present numerical results, validations, and will discuss efficiency, speedup, and scalability.

IV. Open-Loop Dynamics With Locked Voltage Regulation

In this section, we focus our attention on the dependence of the output voltages on the core loading currents

1) after disconnecting the controllers and opening the feedback loops;
2) by "freezing" the duty-cycle signals $\boldsymbol{d}(t) = \boldsymbol{d}$, which are thus considered as a set of fixed (constant) parameters.

The entire PDN structure as depicted in Fig. 6 becomes a large-scale LTI system, which can be fully characterized by the output impedance matrix $\mathbf{Z}(s, \boldsymbol{d})$ relating the output voltages to the core excitation currents through $\boldsymbol{V}^o(s) = \mathbf{Z}(s, \boldsymbol{d})\boldsymbol{I}^o(s)$. This impedance depends on the particular configuration of the duty-cycle parameters $\boldsymbol{d}$, which are indeed intended to modulate the core voltages. Samples of this parameterized transfer function can be obtained, as anticipated in Section II-A, through AC analyses in a commercial circuit solver where the whole system is described using SPICE circuit realizations resulting from the preprocessing phase and duty-cycle parameters are fixed.

Given the above structure, we apply a second layer of model order reduction through a second rational fitting stage, with the objective of further reducing overall model complexity by exploiting the interactions between input and output networks. In particular, it is expected that the resistive/capacitive and lowpass behavior of the output network models provides a filtering and smoothing effect, thus enabling a compact low-order representation of the output impedance as observed from the output ports. Therefore, we consider a general model structure

$$\mathbf{Z}(s, \boldsymbol{d}) = \sum_{\nu=1}^{\bar{\nu}} \frac{\mathbf{R}_\nu(\boldsymbol{d})}{s - p_\nu} \tag{3}$$

where common poles p_ν are used to represent all impedance entries, and where the associate residues $\mathbf{R}_\nu$ are parameterized through low-order polynomials. Justification for this structure and detailed considerations on the evaluation of poles and parameterized residues follow.

A. Macromodel Structure

Let us denote with $P = N_c N_o$ the total number of output ports, so that the output impedance matrix $\mathbf{Z} \in \mathbb{C}^{P\times P}$. As typical in rational macromodeling, a set of frequency response samples are computed through small-signal AC sweeps as

$$\check{Z}_{ij}(j\omega_\ell; \boldsymbol{d}_\mu), \quad ij = 1, \ldots, P, \ \ell = 1, \ldots, L, \ \mu = 1, \ldots, M \tag{4}$$

where L is the total number of frequency samples, and each response is characterized by a given configuration of duty-cycle parameters $\boldsymbol{d}_\mu = (d_1^{(\mu)}, \ldots, d_{N_c}^{(\mu)})^{\mathsf{T}}$. Fig. 7 depicts a representative set of responses for the Mobile example, computed for a full ($N_c = 4$)-dimensional sweep over the duty-cycle parameter space. The responses can be grouped in two main classes, namely intra-core responses (largest magnitude, further split in Fig. 7 into diagonal 'A' and non-diagonal 'B') and inter-core couplings (smaller magnitude, 'C'). Within each

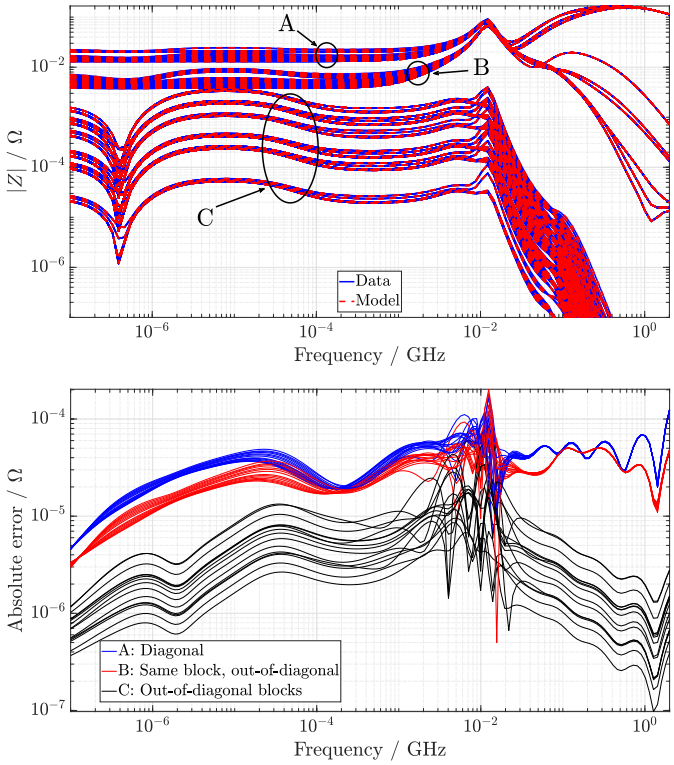


Fig. 7. Top panel shows a representative set of output impedance responses for the Mobile example, evaluated for different combinations of the duty-cycle parameters d_k (solid blue lines). Corresponding responses of a common-pole macromodel (3) are depicted with red dashed lines. Responses are classified in three categories: *A* (*diagonal*) refers to driving-point (self) impedances within each core; *B* (*same block, out-of-diagonal*) refers to impedance matrix entries representing cross-coupling between different load ports within the same core; *C* (*out-of-diagonal*) refers to coupling between load ports of different cores. The bottom panel shows the absolute error for selected representative transfer matrix entries for each category and for several combinations of the duty-cycle parameters d_k.

class, all responses look "very similar" and are characterized by resonant/antiresonant peaks that are located at the same frequencies.

Although the generation of (parameterized) rational macromodels can be considered as a fundamentally solved problem under a theoretical standpoint [7], [8], including stability and passivity enforcement [9], [10], [11], [12], [13], [14], the practical application to the problem under investigation poses some critical challenges, mainly due to the possibly large number of cores N_c. An effective algorithm must be scalable to at least one hundred cores, with larger figures expected for next generation microprocessors. The total number of output ports P can easily reach several thousands or tens of thousands. In turn, the number of responses P^2 is expected to reach millions or more, and each of these responses potentially depends on the duty-cycle d_k of each kth core. Even storage of the entire dataset that is usually required to fit a parameterized macromodel becomes unfeasible, in addition to the overhead required to evaluate the response data to be fit and the runtime to perform the rational fit. Fortunately, the specific features of the PDN under investigation allow several drastic simplifications, discussed below.

1) Common Poles: The first important assumption is on the suitability of a common-pole rational approximation (3) of all impedance responses, for all possible duty-cycle combinations. This hypothesis has been verified by constructing such a common-pole model of a large set of representative output impedance responses (for various duty-cycle combinations $\boldsymbol{d}_\mu$) and checking that the accuracy is satisfactory. A comparison between model and data for the Mobile example is provided in Fig. 7, where no visual difference can be appreciated between model and data at this scale. Similar results apply for the Server example in the various tested configurations (more details will be provided in Section VII). These results support and confirm the suitability of a common-pole structure.

2) Block-Structured Residue Parameterization: The second key ingredient enabling the proposed approach is the suitability of a block-structured and sparse dependence of the impedance matrix on the duty-cycle parameters. Let us partition the $P \times P$ impedance matrix into blocks as

$$\mathbf{Z}(s,\boldsymbol{d}) = \begin{bmatrix} \mathbf{Z}_{1,1}(s,\boldsymbol{d}) & \mathbf{Z}_{1,2}(s,\boldsymbol{d}) & \cdots & \mathbf{Z}_{1,N_c}(s,\boldsymbol{d}) \\ \mathbf{Z}_{2,1}(s,\boldsymbol{d}) & \mathbf{Z}_{2,2}(s,\boldsymbol{d}) & \cdots & \mathbf{Z}_{2,N_c}(s,\boldsymbol{d}) \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{Z}_{N_c,1}(s,\boldsymbol{d}) & \mathbf{Z}_{3,1}(s,\boldsymbol{d}) & \cdots & \mathbf{Z}_{N_c,N_c}(s,\boldsymbol{d}) \end{bmatrix}$$

where the individual blocks $\mathbf{Z}_{k,k'}(s,\boldsymbol{d}) \in \mathbb{C}^{N_o \times N_o}$. After an extensive verification campaign on both Mobile and Server benchmarks, it was concluded that the dependence of block (k,k') on duty-cycle signals $\{d_q,\ q \neq k,k'\}$ is negligible. Therefore, all matrix elements of each block (k,k') can be parameterized only by two independent duty-cycle components d_k and $d_{k'}$. Moreover, elements of diagonal blocks (k,k) can be expressed as univariate functions of the corresponding d_k only.

This structure is readily understood looking at Fig. 1. The diagonal blocks (k,k) provide the voltages of one core resulting from loading the same core k. Such voltages depend predominantly on the FIVR that drives the same core k. All other FIVRs not directly connected to this core are expected by design to have a minimal influence. The off-diagonal blocks (k,k') provide instead the cross-coupling effect that current switching on one core k' induces on an different core k. It is therefore expected that

1) the FIVR connected to the switching core k' adapts its duty cycle, thus offering a different loading condition on the input network. The cross-coupling through the input network becomes visible also to core k;
2) the FIVR connected to the "victim" core k captures and regulates the induced voltage fluctuations on code k, which are then visible at all output ports.

In summary, although the complete output impedance matrix $\mathbf{Z}(s,\boldsymbol{d})$ depends on all N_c independent parameters, each individual block depends at most on two parameters.

As a verification of this fact, a numerical experiment has been carried out in which two models for the same Mobile example are compared. In the first model, no assumption was made on the parametric dependence and all entries are assumed to depend on all duty-cycle parameters. In the second, the matrix entries depend on at most two parameters according

TABLE II
VALIDATION ERRORS FOR DIFFERENT PARAMETERIZED MODELS OF THE MOBILE EXAMPLE

Parameterization scheme	RMS error
Full	$3.64 \cdot 10^{-5}$
Sparse	$4.90 \cdot 10^{-5}$
Sparse and compressed	$5.58 \cdot 10^{-4}$

to the block structure of $\mathbf{Z}(s, \boldsymbol{d})$ as discussed above. Columns *Full* and *Sparse* in Table II report the worst case root-mean square (RMS) errors among all responses of these two models with respect to the corresponding frequency data, computed over a validation set of duty-cycle combinations $\boldsymbol{d}_\mu$ (i.e., not used for training the models). The accuracy of these two models is practically identical, thereby supporting and validating the proposed sparse parameterization scheme.

3) Exploiting Redundancy: The third enabling factor for proposed data-driven algorithm is the extreme redundancy of the full set of impedance responses (4). Looking at Fig. 7, we notice that all responses belong to one of very few categories (diagonal, off-diagonal in the same block, off-diagonal blocks). Within each category, all responses are practically identical except for minimal variations. This is exactly the situation in which the complete set of responses can be represented by a reduced set of "basis" functions. This fact is common to all those situations in which a very large set of ports is spread throughout a system sharing the same set of resonances. Each port excites and collects the superposition of the same modes. Therefore, all responses are characterized by the same resonances (hence a further justification of the common-pole approximation), and responses associated with ports that are geometrically/electrically close look very similar.

We exploit this redundancy by applying the *compressed macromodeling* framework originally presented in [15] and further elaborated/extended in [16]. In the present setting, this framework becomes a procedure by which we can represent the impedance matrix samples of a large dataset in a compressed form by identifying a low-dimensional subspace that is sufficient to describe the frequency dependence of all transfer matrix entries in (3) for virtually any value of $\boldsymbol{d}$ in the parametric domain.

For a precise statement of this procedure, consider the parametric dataset of output impedance matrices evaluated at a finite set of frequencies and parameter values (4) as arranged in a four-way tensor $\check{\mathcal{Z}}$, defined as $\check{\mathcal{Z}}_{i,j,\ell,\mu} = \check{Z}_{i,j}(j\omega_\ell, \boldsymbol{d}_\mu)$. Consider its mode-3 matricization $\check{\mathcal{Z}}_{(\ell)}$ (i.e., matricization with respect to the frequency index ℓ), which gives a matrix whose columns contain all frequency samples $\{\check{Z}_{i,j}(j\omega_\ell, \boldsymbol{d}_\mu),\ \ell = 1, \ldots, L\}$ corresponding to any fixed value of i, j and μ available in the dataset. The matrix $\check{\mathcal{Z}}_{(\ell)} \in \mathbb{C}^{L \times (P^2 M)}$ has L rows and $P^2 M$ columns. Although the number of columns is very large, the properties of this dataset ensure that there is a low-dimensional subspace spanned by a set of basis vectors that can be linearly combined to approximate all columns of $\check{\mathcal{Z}}_{(\ell)}$ up to an arbitrary and tuneable precision. We further split real and imaginary parts of this matrix and define the real-valued data matrix

$$\bar{\mathcal{Z}} = \begin{pmatrix} \mathrm{Re}\{\check{\mathcal{Z}}_{(\ell)}\} \\ \mathrm{Im}\{\check{\mathcal{Z}}_{(\ell)}\} \end{pmatrix}.$$

We now look for an orthonormal basis of ρ vectors $\{\check{\boldsymbol{w}}_r \in \mathbb{R}^{2L},\ r = 1, \ldots, \rho\}$ to approximate the column space of $\bar{\mathcal{Z}}$ with minimal error. The optimal choice for $\{\check{\boldsymbol{w}}_r\}$ consists in the first ρ principal components of $\bar{\mathcal{Z}}$ as given by its singular-value decomposition. Let us define the matrices

$$\check{\mathbf{W}} = \begin{pmatrix} \check{\boldsymbol{w}}_1 & \cdots & \check{\boldsymbol{w}}_\rho \end{pmatrix}, \quad \check{W} = \begin{pmatrix} \mathbb{I}_L & j\mathbb{I}_L \end{pmatrix} \bar{\mathbf{W}} \tag{5}$$

so that the projection of the data matrix on the low-dimensional subspace $\mathcal{W} = \mathrm{span}\{\boldsymbol{w}_1, \ldots, \boldsymbol{w}_\rho\}$ is given by $\bar{\mathbf{W}}^T \bar{\mathcal{Z}}$. The reconstructed matrix is $\bar{\mathbf{W}}\bar{\mathbf{W}}^T \bar{\mathcal{Z}} \approx \bar{\mathcal{Z}}$ and

$$\check{\mathcal{Z}}_{(\ell)} \approx \check{\mathbf{W}}\bar{\mathbf{W}}^T \bar{\mathcal{Z}} = \check{\mathbf{W}}\check{\mathbf{V}} \tag{6}$$

where we set $\bar{\mathbf{W}}^T \bar{\mathcal{Z}} \triangleq \check{\mathbf{V}}$.

In a practical implementation, the matrix $\bar{\mathcal{Z}}$ might be too large for a direct singular value decomposition (SVD). Nonetheless, it is still possible to randomly select a subset of its columns that is sufficiently larger than the target rank ρ and compute the principal components of this submatrix. This process is a simplified form of the so-called *randomized SVD* methods [17], which are proven to be adequate to derive an appropriate low-dimensional subspace $\mathcal{W}$ for which (6) holds with sufficient accuracy.

As proven in [15], the approximation error due to the low-rank approximation (6) is related to the first neglected singular value. Therefore, an accuracy-complexity tradeoff can be exploited by selecting a target accuracy for singular-value truncation and automatically deriving the required number of principal components ρ. For the Mobile example, only $\rho = 20$ components are required to reproduce the complete impedance responses with an approximation error $\varepsilon = 8.7 \times 10^{-5}$, corresponding to a reduction factor as much as $\rho/P^2 \approx 0.1\%$.

B. Macromodeling Principal Component Vectors

In (6), we can view the columns of $\check{\mathbf{W}}$ as being the frequency-domain samples of some unknown transfer functions (*principal components* or *basis vectors*), and $\check{\mathbf{V}}$ as the (real-valued) coefficients by which they are combined to recover the transfer function samples arranged along the columns of $\check{\mathcal{Z}}_{(\ell)}$. Hence, we can immediately model the frequency dependence of the entire impedance data (4) by building a model

$$\mathbf{W}(s) = \begin{pmatrix} \boldsymbol{w}_1(s) & \cdots & \boldsymbol{w}_\rho(s) \end{pmatrix}$$

of the basis functions only. This can be obtained by rational fitting where the following condition is enforced:

$$\boldsymbol{w}_r(j\omega_\ell) \approx \check{\mathbf{W}}_{\ell,r} \quad r = 1, \ldots, \rho,\ \ell = 1, \ldots, L. \tag{7}$$

In particular, in this work, we resort to VF to obtain a stable pole-residue model of order $\bar{\nu}$ for $\mathbf{W}(s)$

$$\mathbf{W}(s) = \sum_{\nu=1}^{\bar{\nu}} \frac{\boldsymbol{\Phi}_\nu}{s - p_\nu} \tag{8}$$

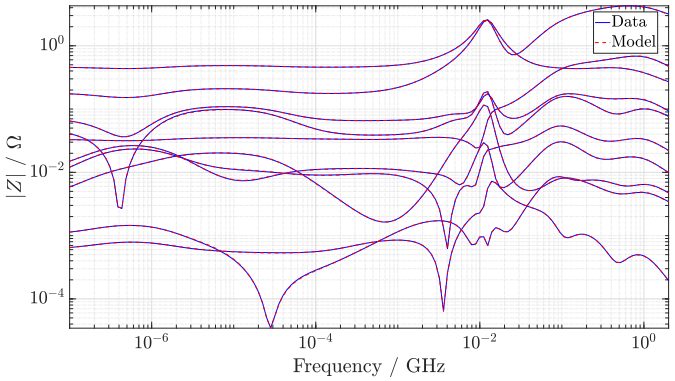


Fig. 8. Common-pole macromodel responses (red dashed lines) compared to the first ten principal component (basis functions) data (solid blue lines) for the Mobile example.

where p_ν and $\mathbf{\Phi}_\nu$ are the νth pole and residue, respectively. Since the basis function data inherit the frequency-dependence of the underlying large-scale impedance response data, also for the model $\mathbf{W}(s)$ a common-pole structure is appropriate. Fig. 8 compares the basis function model to the corresponding data for the Mobile example. Since these basis functions can differ significantly in magnitude, using relative error (inverse magnitude) weighting for rational fitting is recommended [5]. This is the adopted weighting scheme in all numerical examples.

C. Compressed Parameterization

Through the procedure derived above, we are able to turn the impedance response samples (4) computed for any particular and *fixed* duty-cycle configuration $\boldsymbol{d}_\mu$ into a compressed representation. This is done by first assembling the tensor $\check{\mathcal{Z}}(\boldsymbol{d}_\mu)$ with $[\check{\mathcal{Z}}(\boldsymbol{d}_\mu)]_{i,j,\ell} = \check{Z}_{i,j}(j\omega_\ell, \boldsymbol{d}_\mu)$. Splitting then the real and imaginary parts of its mode-3 matricization gives the data matrix $\bar{\mathcal{Z}}(\boldsymbol{d}_\mu)$. The coefficients of its compressed representation are obtained by projection

$$\check{\mathbf{V}}_\mu = \bar{\mathbf{W}}^T \bar{\mathcal{Z}}(\boldsymbol{d}_\mu) \tag{9}$$

where the projection matrix $\bar{\mathbf{W}}^T$ is fixed and known and where $\check{\mathbf{V}}_\mu \in \mathbb{R}^{\rho \times P^2}$. The data reconstruction thus reads

$$\check{\mathcal{Z}}(\boldsymbol{d}_\mu)_{(\ell)} \approx \check{\mathbf{W}}\check{\mathbf{V}}_\mu. \tag{10}$$

In the above derivations, we have explicitly introduced the argument $(\boldsymbol{d}_\mu)$ in all matrices that depend on this parameter. We now exploit this dependence to construct a continuously parameterized model for any arbitrary value of $\boldsymbol{d}$.

Let us compute (9) and (10) for a set of duty-cycle combinations $\{\boldsymbol{d}_\mu,\ \mu = 1, \ldots, M\}$. This data is used to train (fit) a multivariate polynomial model

$$\mathbf{V}(\boldsymbol{d}) = \sum_{\boldsymbol{\alpha} \in \mathcal{A}} \mathbf{V}_{\boldsymbol{\alpha}} d_1^{\alpha_1}, \ldots, d_{N_c}^{\alpha_{N_c}} \tag{11}$$

where $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_{N_c})$ is an N_c-tuple of integers and $\mathcal{A}$ is the set $\mathcal{A} = \{(\alpha_1, \ldots, \alpha_{N_c}),\ 0 \le \alpha_i \le \theta\}$, where θ is the polynomial order on each individual parameter. The coefficients $\mathbf{V}_{\boldsymbol{\alpha}}$ of this approximation are computed in the least-squares sense

$$\mathbf{V}(\boldsymbol{d}_\mu) \approx \check{\mathbf{V}}_\mu, \quad \mu = 1, \ldots, M. \tag{12}$$

In order to ensure a sufficiently overdetermined least squares problem for numerical robustness, we ensure that M is larger (e.g., ten times) than the total number of polynomial coefficients for each matrix entry (see below). The choice of multivariate polynomials is motivated by several previous works on parameterized macromodeling, where it has been extensively shown that polynomials are suitable for parameterization of the transfer function numerator and denominator coefficients, see [14]. In the present case, the parameter dependence of $\mathbf{V}(\boldsymbol{d})$ in (11) is equivalent to parameterization of the transfer function residues, as shown explicitly in the following derivations.

The multivariate polynomial (11) provides a parameterization of the duty-cycle dependence of the impedance responses in tensorized form

$$\check{\mathcal{Z}}(\boldsymbol{d})_{(\ell)} \approx \check{\mathbf{W}}\mathbf{V}(\boldsymbol{d}) \quad \forall \boldsymbol{d} \in [0, 1]^{N_c}. \tag{13}$$

Replacing now principal component data $\check{\mathbf{W}}$ with the corresponding rational approximation (8) leads to a parameterized macromodel of the output impedance

$$\mathbf{Z}(s, \boldsymbol{d}) = \text{mat}\{\mathbf{W}(s)\mathbf{V}(\boldsymbol{d})\} = \sum_{\nu=1}^{\bar{\nu}} \frac{\text{mat}\{\mathbf{\Phi}_\nu \mathbf{V}(\boldsymbol{d})\}}{s - p_\nu} \tag{14}$$

where the mat$\{\cdot\}$ operator reshapes the argument into a $P \times P$ matrix. Note that this expression is compatible with (3), where the residue parameterization is induced in compressed form through $\mathbf{R}_\nu(\boldsymbol{d}) = \mathbf{\Phi}_\nu \mathbf{V}(\boldsymbol{d})$.

The polynomial interpolation as presented in (11) is general but would not be scalable to large N_c due to the huge number of monomial terms involved in the expansion. However, we have already observed and confirmed through validation that each block (k, k') of $\mathbf{Z}(s, \boldsymbol{d})$ depends only on the two duty-cycle components d_k, $d_{k'}$. With the adopted notation, any given column m of the polynomial coefficient matrices $\mathbf{V}_{\boldsymbol{\alpha}}$ corresponds with a single impedance matrix element (i, j) through

$$i = 1 + \text{mod}(m - 1, P), \quad j = \lceil m/P \rceil$$

where mod is the reminder of integer division and $\lceil \cdot \rceil$ rounds its argument to the nearest larger integer. In turn, element (i, j) maps to matrix block (k, k') via

$$k = 1 + \lfloor (i - 1)/N_o \rfloor, \quad k' = 1 + \lfloor (j - 1)/N_o \rfloor$$

where $\lfloor \cdot \rfloor$ rounds its argument to the nearest smaller integer. Collectively, these expressions can be summarized by a map $\{k, k'\} = \mathcal{F}(m)$ that identifies the two parameters d_k, $d_{k'}$ on which each column m depends on. The latter column m can be assumed to be vanishing for all $\boldsymbol{\alpha}$ except for those where $\alpha_i = 0\ \forall i \notin \{k, k'\} = \mathcal{F}(m)$. In particular, $\mathbf{V}_{\boldsymbol{\alpha}} = 0$ if $\boldsymbol{\alpha}$ contains three or more non-zero elements. Hence, the special pattern of parametric dependence translates into the sparsity of the coefficients $\mathbf{V}_{\boldsymbol{\alpha}}$ which makes this parameterization feasible even with large N_c. In this work, we adopt polynomials of maximum degree $\theta = 2$, which implies that each matrix element is associated with at most $(1+\theta)^2 = 3^2 = 9$ nonvanishing coefficients (instead of the 3^{N_c} coefficients of a non-sparse polynomial). Therefore, the total

number of parameter samples used in the least squares fit (12) is $M \sim 90$, which is moderate even in the high-dimensional case.

D. Complete Modeling Flow

Now that all basic ingredients supporting the proposed compressed, sparse and parameterized output impedance model representation are available, we summarize all required modeling steps, together with practical considerations for handling complexity and ensuring scalability.

1) *Data Collection:* The first step is to collect frequency responses (4) finalized to building a common-pole macromodel. Few responses are required thanks to their redundancy, therefore we perform a random selection of impedance matrix elements (i, j), whose frequency samples are computed for a set of duty-cycle combinations $\{\boldsymbol{d}_\mu,\ \mu = 1, \ldots, M\}$. In order to ensure scalability to high core counts N_c, we adopt a Sobol sequence [18] in the unit hypercube $[0, 1]^{N_c}$, with M elements.
2) *Construct Basis Functions:* The above data are used to construct the data matrix $\check{\mathbf{W}}$ collecting the SVD-generated principal components, see (5).
3) *Rational Fit:* A common-pole model $\mathbf{W}(s)$ of the basis functions is constructed using a standard VF process, leading to (8).
4) *Full Data Collection:* For each $\mu = 1, \ldots, M$ evaluate the complete set of impedance responses $\check{Z}_{i,j}(j\omega_\ell; \boldsymbol{d}_\mu)$ for $i, j = 1, \ldots, P$. Given the expected very large dataset, the projection (9) of each computed impedance matrix onto the reduced basis vectors is performed on the fly, and only the associated coefficients $\check{\mathbf{V}}_\mu$ are stored. This stage of data compression is essential to ensure scalability of proposed approach to systems with large port counts. We also collect the impedance response data for an additional batch of parameter samples $\boldsymbol{d}_\mu$ with $\mu = M + 1, \ldots, M$, to be used for self-validation of the macromodel parameterization.
5) *Parameterization:* The sparse parameterized macromodel in compressed form (14) is finally computed by solving the polynomial approximation (12). This step is not memory-intensive since each block can be processed independently.

The above flow applied to the Mobile example leads to the results depicted in Fig. 9, which compares the compressed and sparse parameterized model responses to the corresponding responses over the validation set (not used for training). No visual difference is appreciated from this plot. A detail on the approximation errors is provided in Fig. 10, which reports the maximum RMS error for each output impedance matrix entry over the validation set. The errors of all matrix elements are well below the reasonable engineering accuracy required for the present application.

E. Time-Domain Representation and Validation

As a final step, the sparse compressed parameterized macromodel (14) is realized in state-space form to enable

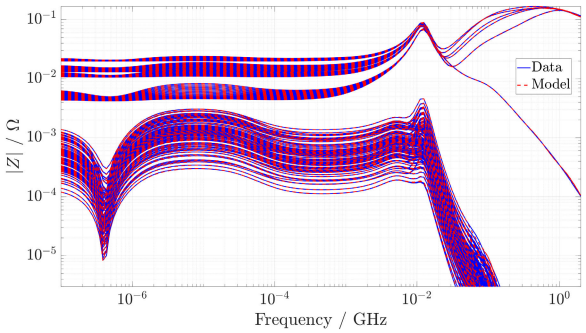


Fig. 9. Data-model comparison for the Mobile example. The model construction exploits the structural assumption that each output impedance matrix entry $[\mathbf{Z}_{k,k'}]_{i,j}$ depends only on the two duty-cycle parameters associated with the cores k, k'. Furthermore, impedance matrix entries are in a compressed representation (14).

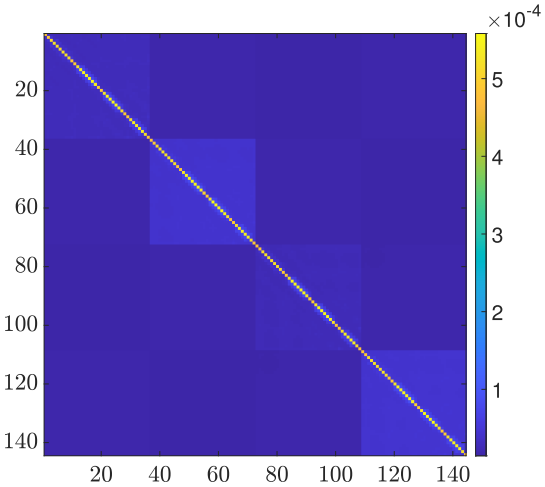


Fig. 10. Maximum RMS error for each transfer matrix entry of a model of the Mobile example employing compression and sparse duty-cycle parameterization. For each parameter value in a validation set, the RMS error has been computed for all entries in the $\mathbf{Z}$ matrix and the largest error is reported in this figure.

transient analysis. The following structure is obtained:

$$\begin{cases} \dot{\boldsymbol{x}}_o = \mathbf{A}_o \boldsymbol{x}_o + \mathbf{B}_o(\boldsymbol{d})\boldsymbol{i}^o \\ \boldsymbol{v}^o = \mathbf{C}_o \boldsymbol{x}_o \end{cases} \tag{15}$$

where the dynamic matrix $\mathbf{A}_o$ is block-diagonal with repeated poles p_ν, matrix $\mathbf{C}_o$ maps states to output voltages through stacked identity matrices, and where matrix $\mathbf{B}_o(\boldsymbol{d})$ realizes the sparsely parameterized residues $\mathbf{R}_\nu(\boldsymbol{d}) = \boldsymbol{\Phi}_\nu \mathbf{V}(\boldsymbol{d})$. This realization is standard, the reader is referred to [5] for details. Note that the realization process involves no approximation so that, for any combination of *constant* duty-cycle values $\boldsymbol{d} \in [0, 1]^{N_c}$, the transfer function associated with (15) matches exactly the rational output impedance form (14). Provided the standard stability enforcement techniques are used during the VF modeling stage, the poles p_ν have strictly negative real part and system (15) is asymptotically stable irrespective of the value of $\boldsymbol{d}$.

In order to further validate this state-space model, we report in Fig. 11 the transient results obtained by running the Mobile state-space model with a fixed configuration of duty-cycle parameters while switching output currents. The results

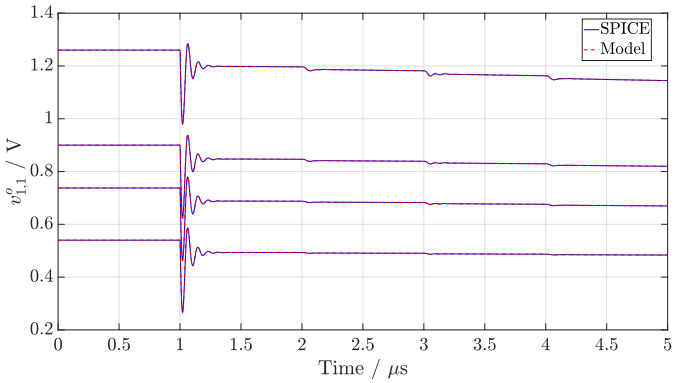


Fig. 11. Four transient simulations of the Mobile parameterized state-space macromodel (15). In each simulation, all cores are driven with the same duty-cycle value, set to 0.3, 0.41, 0.5, and 0.7, respectively. Output currents switch from 0 to 10 A per core, synchronously within each core and uniformly distributed over its ports. Different cores are excited sequentially at $t = 1, 2, 3, 4$ μs. Reference HSPICE results (solid blue lines) and corresponding macromodel responses (red dashed lines).

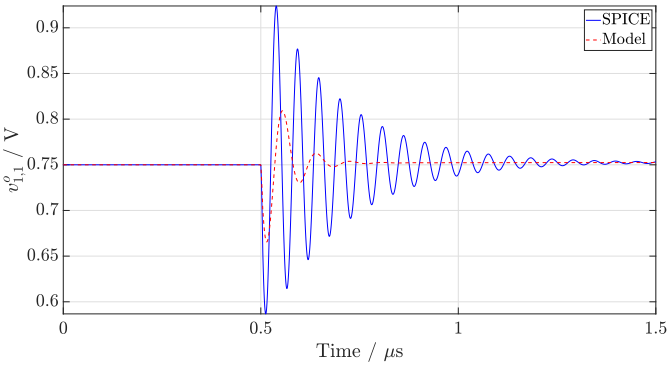


Fig. 12. Transient simulation of the Mobile parameterized state-space macromodel (15) after closing voltage regulation feedback loops. Output voltage of one port of core 1 resulting from 10 A total current switching in 5 ns on the same core. Reference HSPICE results (solid blue lines) and corresponding macromodel responses (red dashed lines). This results confirms that (15) is still not adequate to model closed-loop dynamics.

from the parameterized state-space model are compared to a reference HSPICE simulation. As expected, we observe in time-domain the same level of accuracy that was attained in the frequency domain.

V. Modeling Duty-Cycle Induced Dynamics

Although the macromodeling flow described in Section IV provides the output impedance for any value of the parameter $\boldsymbol{d}$ in its domain, this model is still unsuitable when $\boldsymbol{d}$ is a time-varying signal. In fact, reconnecting the feedback loops as in Fig. 2 makes $\boldsymbol{d} = \boldsymbol{d}(t)$ a time-varying signal resulting from the voltage regulation induced by each core controller. In turn, this signal triggers some dynamics through its (nonlinear) interaction with the voltage and current signals from the FIVR switches. A missing or improper representation of such dynamics leads inevitably to wrong results. This is confirmed by Fig. 12, which compares the result of the state-space macromodel (15) after closing the controller feedback to the corresponding HSPICE reference. The macromodel results are not correct due to an incomplete model. It was already anticipated in (2) that a basic linearization of the nonlinear and time-varying dynamics induced by the FIVR switches would require an extra term in a state-space representation, which considers $\boldsymbol{d}$ as an input and not as a parameter. This term is introduced next.

In order to obtain an adequate model, we now regard $\boldsymbol{d}$ as an input and we study its effect on the output voltage. The simplest representation is through linearized (small-signal) dynamics, therefore we assume the following decomposition:

$$\boldsymbol{d}(t) = \bar{\boldsymbol{d}} + \tilde{\boldsymbol{d}}(t) \tag{16}$$

where $\bar{\boldsymbol{d}}$ is a constant "bias" (reference) level and $\tilde{\boldsymbol{d}}(t)$ is a small-signal component. Dynamics induced by duty-cycle variations are considered through a small-signal model whose inputs are $\tilde{\boldsymbol{d}}$ and the corresponding outputs are $\tilde{\boldsymbol{v}}^o_{\rm ss}$. The bias value of the load voltage corresponding to $\bar{\boldsymbol{d}}$ is denoted as $\bar{\boldsymbol{v}}^o_{\rm ss}$. The small-signal model linking the small signal $\tilde{\boldsymbol{d}}(t)$ to the corresponding output voltage deviation $\tilde{\boldsymbol{v}}^o_{\rm ss}$ admits the following representation:

$$\begin{cases} \dot{\boldsymbol{x}}_{\rm ss} = \mathbf{A}_{\rm ss}\boldsymbol{x}_{\rm ss} + \mathbf{B}_{\rm ss}\tilde{\boldsymbol{d}} \\ \tilde{\boldsymbol{v}}^o_{\rm ss} = \mathbf{C}_{\rm ss}\boldsymbol{x}_{\rm ss} \end{cases} \tag{17}$$

and the corresponding large-signal load voltage $\boldsymbol{v}^o_{\rm ss}(t)$ is the superposition of the bias and the small-signal components

$$\boldsymbol{v}^o_{\rm ss}(t) = \bar{\boldsymbol{v}}^o_{\rm ss} + \tilde{\boldsymbol{v}}^o_{\rm ss}(t). \tag{18}$$

A. Small-Signal Model Identification

The model in (17) represents the small-signal response of the load voltage to a small-signal excitation on the duty-cycle input $\tilde{\boldsymbol{d}}(t)$. The transfer function describing this effect has N_c inputs and P outputs and can be obtained in practice through an AC analysis, e.g., using a commercial circuit solver such as SPICE, as outlined below. In the following, the linearized model (17) is centered at the operating point corresponding to $\bar{V}_{\rm VRM} = 1.8$ V, $\bar{\boldsymbol{i}}^o = 0$ A and $\bar{d}_k = 0.41$ $\forall k$. Thus, the simulation includes a 1.8 V DC voltage source $V_{\rm VRM}$ connected as in Fig. 1 and N_c DC voltage sources setting the bias value of the nodes d_k. As for the small signal, N_c AC excitation sources are series connected to the $\bar{d}_k$ bias sources and activated one at the time while running N_c independent small-signal AC sweeps. The kth run gives the frequency-domain samples $\boldsymbol{v}^o_{\rm ss}\big|_{d_k}(j\omega_\ell)$, i.e., the small-signal effect of the individual d_k on $\boldsymbol{v}^o_{\rm ss}$. Collecting these in a matrix gives

$$\breve{\mathbf{H}}(j\omega_\ell) = \left(\boldsymbol{v}^o_{\rm ss}\big|_{d_1}(j\omega_\ell) \cdots \boldsymbol{v}^o_{\rm ss}\big|_{d_{N_c}}(j\omega_\ell)\right).$$

The model in (17) is easily obtained by rational fitting $\breve{\mathbf{H}}(j\omega_\ell)$. The standard VF algorithm provides a stable model $\mathbf{H}(s)$ of any desired order to match the small-signal data. Finally, a state-space realization of $\mathbf{H}(s)$ gives the matrices $\mathbf{A}_{\rm ss}$, $\mathbf{B}_{\rm ss}$, and $\mathbf{C}_{\rm ss}$ in (17).

It is important to remark that, although in principle a small-signal model is valid only in a small neighborhood of the selected operating point, the actual extent of this neighborhood may be quite large when dealing with systems that are only slightly nonlinear. This is the case of the regulated PDNs here analyzed, for which the voltage response to the duty-cycle

variations is reproduced with acceptable accuracy even for large deviations from the linearization point. This statement is supported by experimental evidence, see the time-domain simulations of Section VII. Furthermore, we tested and validated this assumption by checking that choosing different operating points to define the small-signal operation had practically no influence on the results.

B. Complete State-Space Macromodel

The combination of (17) and (15) gives the final model representation in state-space form

$$\begin{cases} \dot{\boldsymbol{x}} = \mathbf{A}\boldsymbol{x} + \mathbf{B}_1(\boldsymbol{d})\boldsymbol{i}^o + \mathbf{B}_2\tilde{\boldsymbol{d}} \\ \boldsymbol{v}^o = \mathbf{C}\boldsymbol{x} + \bar{\boldsymbol{v}}^o_{\rm ss} \end{cases}, \quad \boldsymbol{x} = \begin{pmatrix} \boldsymbol{x}_o \\ \boldsymbol{x}_{\rm ss} \end{pmatrix} \tag{19}$$

with

$$\mathbf{A} = \text{blkdiag}(\mathbf{A}_o, \mathbf{A}_{\rm ss}), \qquad \mathbf{B}_1(\boldsymbol{d}) = \begin{pmatrix} \mathbf{B}_o(\boldsymbol{d}) \\ \mathbf{0} \end{pmatrix},$$
$$\mathbf{B}_2(\boldsymbol{d}) = \begin{pmatrix} \mathbf{0} \\ \mathbf{B}_{\rm ss} \end{pmatrix}, \qquad \mathbf{C} = \begin{pmatrix} \mathbf{C}_o & \mathbf{C}_{\rm ss} \end{pmatrix}. \tag{20}$$

This first equation in (19) combines state equations in (15) and (17) respectively, whereas the output equation gives the load voltage $\boldsymbol{v}^o$ as the superposition of the effects induced by load currents (15) and duty-cycle inputs (17) and (18). The complete model (19) is compatible with the expected structure based on a linearization assumption (2), with the only difference being a constant $\mathbf{B}_2$ since based on the linearization at single operating point $\bar{\boldsymbol{i}}_o$. Our numerical experiments on the investigated benchmarks led to the conclusion that an additional parameterization of this term with the load current is indeed not necessary, since a sufficient accuracy is attained with this simpler model representation.

VI. Closed-Loop PDN Macromodel

Under standard operating conditions, the time-domain profile of the duty-cycle $\boldsymbol{d}(t)$ is determined by the compensator blocks, according to the nonlinear state-space

$$\begin{cases} \dot{\boldsymbol{w}}(t) = \mathbf{A}_{\mathcal{K}}\,\boldsymbol{w}(t) + \mathbf{B}_{\mathcal{K}}\,\boldsymbol{e}(t) \\ \boldsymbol{y}(t) = \mathbf{C}_{\mathcal{K}}\,\boldsymbol{w}(t) + \mathbf{D}_{\mathcal{K}}\,\boldsymbol{e}(t) \\ \boldsymbol{d}(t) = g(\boldsymbol{y}(t)) \end{cases} \tag{21}$$

where the matrices $\mathbf{A}_{\mathcal{K}}, \mathbf{B}_{\mathcal{K}}, \mathbf{C}_{\mathcal{K}}, \mathbf{D}_{\mathcal{K}}$ represent the dynamics of all compensators and the elements of $\boldsymbol{e}(t) = \mathbf{N}\boldsymbol{v}^o(t) - \boldsymbol{V}_{\rm ref} \in \mathbb{R}^{N_c}$ are their input error signals. These errors are computed by comparing the reference voltages $\boldsymbol{V}_{\rm ref}$ with a subset of the output voltages, obtained by multiplying $\boldsymbol{v}^o(t)$, with a suitable selector matrix $\mathbf{N}$. Each entry of the duty-cycle vector, $d_k(t)$, is determined based on the value of the corresponding element of $\boldsymbol{y}(t)$, according to the saturation relation

$$d_k(t) = g(y_k(t)) = \begin{cases} 1, & \text{for } y_k \geq 1 \\ y_k, & \text{for } 0 \leq y_k \leq 1 \\ 0, & \text{for } y_k \leq 0. \end{cases} \tag{22}$$

To predict the behavior of the closed-loop PDN, the above equations are to be coupled with those of the parameterized macromodel. To do so, we induce the time dependency on the parameterized matrices in (19), by updating in real time their value according to the output of (21). The overall evolution of the closed-loop PDN environment is then explained by the following system:

$$\begin{cases} \dot{\boldsymbol{x}}(t) = \mathbf{A}\boldsymbol{x} + \mathbf{B}_1(\boldsymbol{d})\boldsymbol{i}^o + \mathbf{B}_2\tilde{\boldsymbol{d}} \\ \boldsymbol{v}^o(t) = \mathbf{C}\boldsymbol{x} + \bar{\boldsymbol{v}}^o_{\rm ss} \\ \dot{\boldsymbol{w}}(t) = \mathbf{A}_{\mathcal{K}}\,\boldsymbol{w}(t) + \mathbf{B}_{\mathcal{K}}\,\boldsymbol{e}(t) \\ \boldsymbol{y}(t) = \mathbf{C}_{\mathcal{K}}\,\boldsymbol{w}(t) + \mathbf{D}_{\mathcal{K}}\,\boldsymbol{e}(t) \\ \boldsymbol{d}(t) = g(\boldsymbol{y}(t)). \end{cases} \tag{23}$$

The above system of equations can be simulated in the time domain by means of the discretization strategy explained in [19], based on a first-order implicit Euler approximation.

Fig. 13(b) and (c) compares the results of the proposed macromodel (23) with closed feedback loops to the reference HSPICE simulation, for a configuration of the excitation (load) currents on the four cores depicted in Fig. 13(a). In particular, Fig. 13(b) reports the output voltage on the first port of first core, demonstrating both the proposed macromodel accuracy as well as the effectiveness of the voltage regulation through FIVRs. Fig. 13(c) reports a closeup on core cross-coupling by showing the voltage fluctuation induced on the last port of the last core by current switching on the other three cores. The transient error is well under control, as depicted in Fig. 13(d), where the difference between proposed method and HSPICE results for all $P = 144$ voltage signals is reported. The attained accuracy level is adequate for the application at hand. Should a more aggressive accuracy be required, a higher-degree parameterization of the sparse macromodel residue or a parameterization of the $\mathbf{B}_2$ matrix in (23) with the load currents may be considered.

A. Boundedness and Stability

Some remarks about the stability of (23) are in order. When matrices $\mathbf{A}_0, \mathbf{A}_{\rm ss}$, are Hurwitz, the condition $0 \leq \boldsymbol{d}(t) \leq 1$ implies that the state vectors $\boldsymbol{x}_0, \boldsymbol{x}_{\rm ss}$ remain bounded under any operating condition. Consequently, the output voltages $\boldsymbol{v}^o(t)$ are also bounded, and so are the errors $\boldsymbol{e}(t)$. Since it is also assumed that $\mathbf{A}_{\mathcal{K}}$ is Hurwitz, we conclude that all the system quantities remain bounded irrespective of the particular excitation $\boldsymbol{i}^0$.

VII. Numerical Results

The effectiveness of the proposed approach was tested through extensive numerical experiments. This section summarizes the results in terms of accuracy and runtime. We recall that the two test examples used in this article include a 4-core mobile platform and an enterprise server microprocessor in three different configurations with 8, 12, and 16 cores, see Table I for details. Detailed results on the Mobile example were already documented in Sections IV–VI, so this section will mostly concentrate on the server testcase.

Application of the proposed macromodeling framework to construct the reduced-order state-space models of the server PDN was carried out using the parameters reported

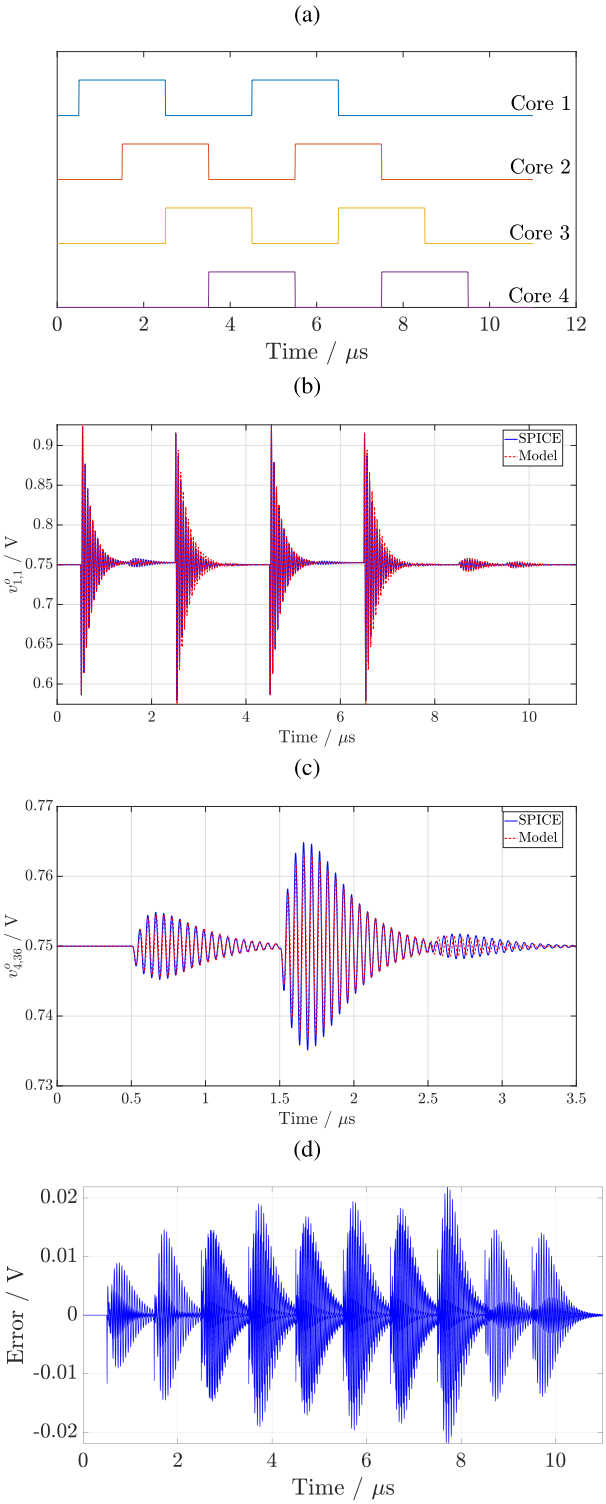


Fig. 13. Transient simulation of the Mobile parameterized state-space macromodel (23) in closed-loop configuration. The load currents exciting each core are shown in (a): the pulse amplitude is 10/36 A and the same waveform is replicated for all 36 ports of each core. The rise and fall times are 5 ns. Output voltages are depicted in (b) and (c), the latter providing a closeup on core cross-coupling (voltage on core 4 due to switching on the other cores). The error between the proposed macromodel simulation and the HSPICE reference is plotted in (d) for all output voltages for all cores (total 144 superimposed error curves).

in Table III, chosen so as to achieve a predefined accuracy around 10^{-4} Ω on the parameterized $\mathbf{Z}$ transfer function. The table reports the number of macromodel poles $\bar{\nu}$, the number

TABLE III
VALIDATION ERRORS AND SETTINGS FOR DIFFERENT PARAMETERIZED MODELS OF THE SERVER EXAMPLE

Example	N_c	$\bar{\nu}$	ρ	θ	RMS error
Server	8	12	10	1	$5.5 \cdot 10^{-4}$
Server	12	12	10	1	$5.5 \cdot 10^{-4}$
Server	16	12	10	1	$5.8 \cdot 10^{-4}$
Server	16	18	15	2	$3.4 \cdot 10^{-4}$

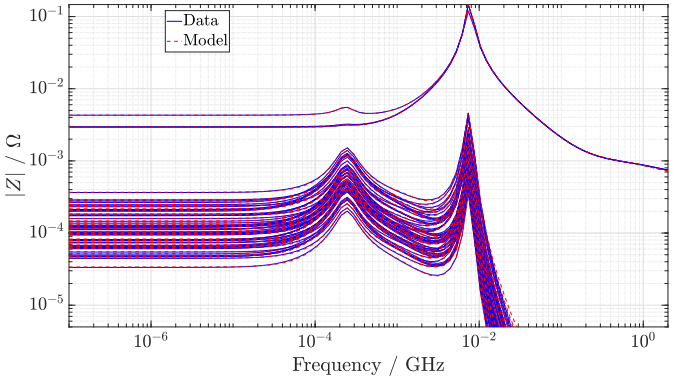


Fig. 14. Representative subset of all parametric model responses of the Server example with 16 cores evaluated on a test set of $\boldsymbol{d}$ points from a Sobol sequence.

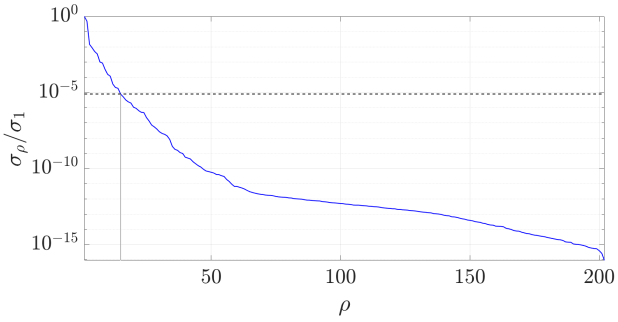


Fig. 15. Normalized singular values of the data matrix $\bar{\mathcal{Z}}$ used to find the basis functions in the Server example. A small number ρ of basis vectors is enough to represent the other entries with a small error (10^{-5}).

of reduced basis functions ρ used in the compression process, and the polynomial degree θ used for duty-cycle parameterization. Note that for the 16-core case two different settings were used to demonstrate that accuracy can be improved by considering higher-order models based on a larger number of basis functions and possibly with higher polynomial degree approximations. The RMS errors reported in the last column of Table III are defined with respect to reference frequency responses obtained by HSPICE.

Fig. 14 shows the parametric model responses ($N_c = 16$, $\bar{\nu} = 18$, $\rho = 15$, $\theta = 2$) compared to HSPICE for a test set of parameter values from a Sobol sequence that are not included in the training set. The good accuracy suggests that multivariate polynomials of low degree are indeed appropriate to reproduce the dependence of the model residues on $\boldsymbol{d}$ because of the good model-data match on the test set. Regarding compression, we find that the Server benchmark confirms the results obtained in the Mobile benchmark because few basis vectors ($\rho = 15$) are sufficient to represent the

TABLE IV
RUNTIME (SECONDS) OF THE MAIN MODELING STEPS

Example	N_c	AC analysis	Fitting (Sec. V-A)	Interpolation (Sec. IV-C)	Compression (Sec. IV-A3)	Basis Fitting (Sec. IV-B)
Mobile	4	48	5	0.9	0.6	1.5
Server	8	1700	63	8.3	32	0.34
Server	12	2300	280	32	64	0.39
Server	16	3100	350	140	130	1.1

TABLE V
TIME-DOMAIN ACCURACY AND RUNTIME FOR DIFFERENT PARAMETERIZED MODELS

Example	Cores N_c	Poles $\bar{\nu}$	Max error	Runtime Proposed	Runtime SPICE	Speedup
Mobile	4	24	20 mV	85 s	4315 s	51×
Server	8	12	17 mV	31 s	1444 s	46×
Server	12	12	20 mV	79 s	1411 s	18×
Server	16	12	24 mV	181 s	1494 s	8×

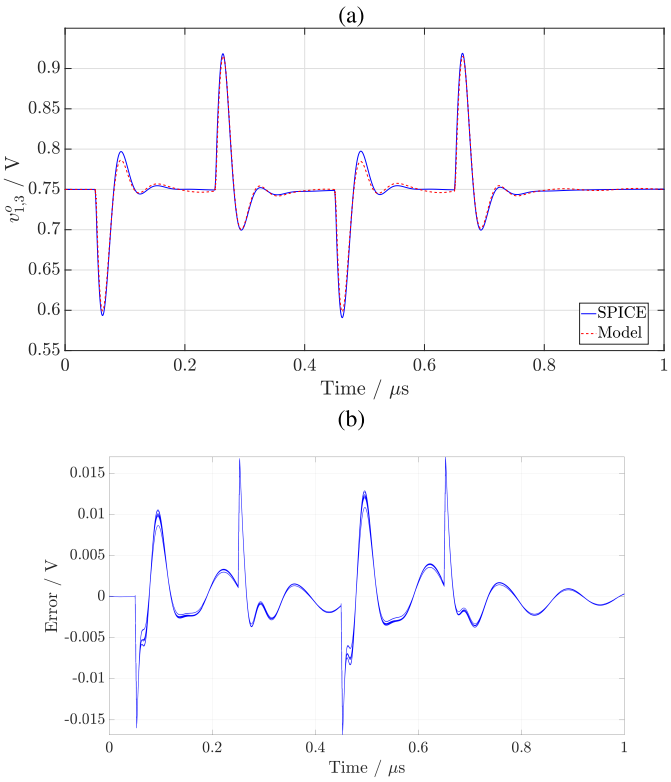


Fig. 16. Transient simulation of the 8-core Server example. (a) Step response at a selected port of the first core. (b) Voltage error on all ports of all cores (total $8 \times 57 = 456$ error curves).

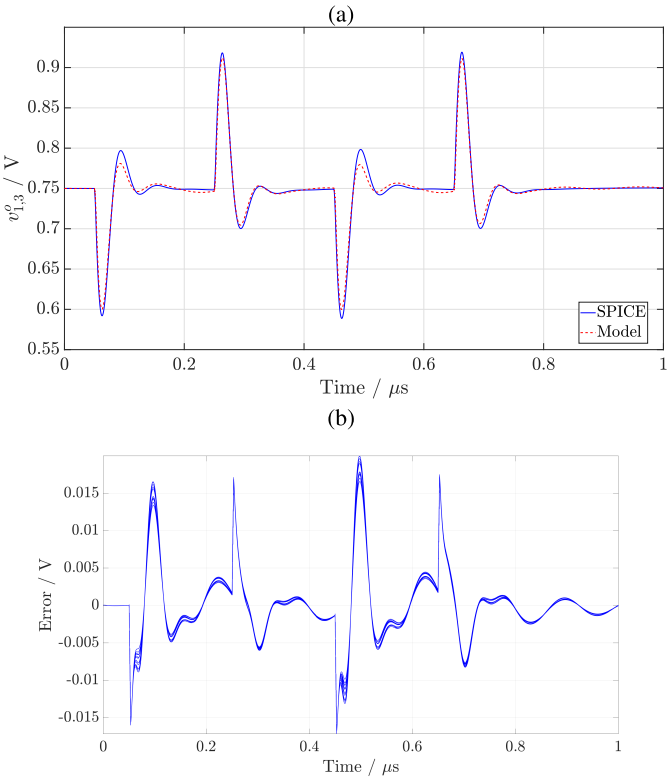


Fig. 17. As in Fig. 16, but for the 12-core Server example. (a) Step response. (b) Voltage errors (all ports, all cores).

entire dataset. The singular values of the data matrix before compression are reported in Fig. 15, showing that $\rho = 15$ vectors are enough for a relative accuracy of about 10^{-5}. In this case, the compression ratio is even better and is equal to $\rho/P^2 \approx 0.03\%$.

Table IV reports the runtime required for the various steps in proposed macromodeling flow, for all testcases. The table confirms that the most time-consuming operation is the collection of the reference responses to be used for training the model. The actual cost of all macromodeling steps is quite moderate and definitely affordable for present application.

Table V reports a summary on time-domain accuracy and runtime in transient closed-loop simulations based on the proposed macromodel-based solver, compared to HSPICE as a reference. Results for both Mobile and Server examples in its three configurations are reported. We remark that the HSPICE runtime is reported excluding the initial operating point analysis, whereas results from proposed solver were obtained using a prototypal implementation in MATLAB, without any parallelization. A detailed analysis of the transient simulation settings and results for the Server example follows.

We use the Server example to investigate the scalability of the proposed approach by repeating all numerical tests with $N_c = 8$, 12, and 16 core configurations. In all cases, we consider a current pulse excitation amplitude 20/57 A on each individual port, so that the total current swing per core equals 20 A with a 3-ns rise time. In this case, we excite all cores simultaneously with two 0.2-μs-long sequential current pulses with rising edges at $t = 0.05$ and 0.5 μs. The resulting load voltages predicted by our model compared with HSPICE are reported in the top panels (a) of Figs. 16–18. The corresponding errors are depicted in the bottom panels (b) and summarized in Table V.

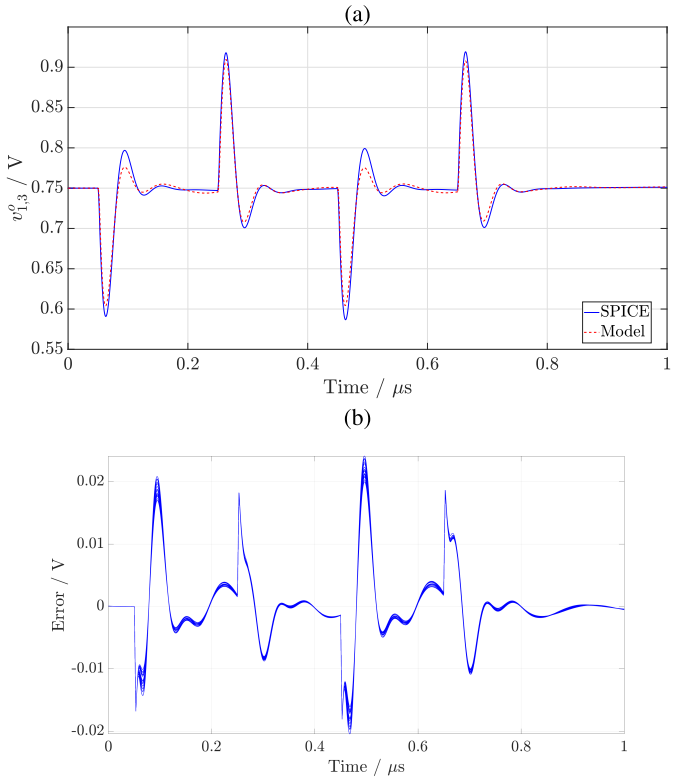


Fig. 18. As in Fig. 16, but for the 16-core Server example. (a) Step response. (b) Voltage errors (all ports, all cores).

These three examples show that the proposed approach, thanks to the combination of compression and rational fitting, gives macromodels of sufficient frequency-domain accuracy using very few poles ($\bar{\nu} = 12$) independently of N_c, as evidenced in Table III. Similarly, the transient errors have similar peak values for all three server configurations (Table V) and their time-domain evolution shows that the maximum error occurs on the rising and falling edges of the load current pulses. This maximum error is practically invariant on the number of cores that are modeled concurrently.

Regarding efficiency, although benchmarks with larger N_c are more computationally intensive and require longer runtime, there is still a significant improvement over HSPICE. For all investigated examples, the proposed (non-optimized) solver leads to a significant speedup, which ranges from 8× to over 50× depending on the testcase.

VIII. Conclusion

This article presented a macromodel-based transient solver for full-system power integrity verification. The system representation includes: an input coupling network representing an electromagnetic model of the PDN of board and package, including optimized decoupling capacitors; a battery of per-core FIVRs, whose switching circuitry is represented through averaged models; an output network with per-core models of FIVR inductors and MIM capacitance, together with a behavioral chip load models.

The proposed approach is based on a hierarchical sparse model order reduction process applied to the output impedance dynamics as seen from the load currents, which is dynamically parameterized by the duty-cycle signals provided by voltage sensing and control circuitry. The resulting model is cast into a state-space model with feedback-induced time-varying linear parameter varying structure, whose simulation in time-domain is straightforward.

Our prototypal implementation was applied to two Intel-based systems, namely a 4-core mobile platform and an 8–16 core enterprise server model. In both cases, the results demonstrated excellent accuracy with respect to reference HSPICE simulations, with a speedup in runtime ranging from 8× to about 50×. We conclude that the proposed approach has good potential, following a code optimization/parallelization and deployment process, to reach dramatic speedup factors in full-system power integrity verification under real workloads, even for massive multicore platforms for high-performance computing (HPC) and artificial intelligence (AI).

References

[1] K. Radhakrishnan, M. Swaminathan, and B. K. Bhattacharyya, "Power delivery for high-performance microprocessors—Challenges, solutions, and future trends," *IEEE Trans. Compon., Packag., Manuf. Technol.*, vol. 11, no. 4, pp. 655–671, Apr. 2021.

[2] E. A. Burton et al., "FIVR—Fully integrated voltage regulators on 4th generation Intel CoreT SoCs," in *Proc. IEEE Appl. Power Electron. Conf. Expo.*, Mar. 2014, pp. 432–439.

[3] M. Swaminathan and A. E. Engin, *Power Integrity Modeling and Design for Semiconductors and Systems*. Upper Saddle River, NJ, USA: Prentice-Hall, 2007.

[4] I. Erdin and R. Achar, "MCB-DPO: Multiport constrained barrier method-based decoupling capacitor placement optimization on irregularly shaped planes," *IEEE Trans. Compon., Packag., Manuf. Technol.*, vol. 12, no. 4, pp. 665–675, Apr. 2022.

[5] S. Grivet-Talocia and B. Gustavsen, *Passive Macromodeling: Theory and Applications*. Hoboken, NJ, USA: Wiley, 2015.

[6] *IdEM, Dassault Systèmes*. Accessed: Jul. 11, 2023. [Online]. Available: http://www.3ds.com/products-services/simulia/products/idem/

[7] P. Triverio, S. Grivet-Talocia, and M. S. Nakhla, "A parameterized macromodeling strategy with uniform stability test," *IEEE Trans. Adv. Packag.*, vol. 32, no. 1, pp. 205–215, Feb. 2009.

[8] Y. Q. Xiao, S. Grivet-Talocia, P. Manfredi, and R. Khazaka, "A novel framework for parametric Loewner matrix interpolation," *IEEE Trans. Compon., Packag., Manuf. Technol.*, vol. 9, no. 12, pp. 2404–2417, Dec. 2019.

[9] F. Ferranti, L. Knockaert, T. Dhaene, and G. Antonini, "Passivity-preserving parametric macromodeling for highly dynamic tabulated data based on Lur'e equations," *IEEE Trans. Microw. Theory Techn.*, vol. 58, no. 12, pp. 3688–3696, Dec. 2010.

[10] F. Ferranti, L. Knockaert, and T. Dhaene, "Passivity-preserving parametric macromodeling by means of scaled and shifted state-space systems," *IEEE Trans. Microw. Theory Techn.*, vol. 59, no. 10, pp. 2394–2403, Oct. 2011.

[11] E. R. Samuel, L. Knockaert, F. Ferranti, and T. Dhaene, "Guaranteed passive parameterized macromodeling by using Sylvester state-space realizations," *IEEE Trans. Microw. Theory Techn.*, vol. 61, no. 4, pp. 1444–1454, Apr. 2013.

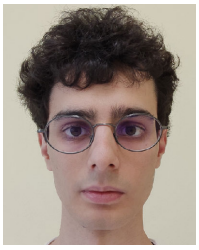
[12] A. Zanco, S. Grivet-Talocia, T. Bradde, and M. De Stefano, "Enforcing passivity of parameterized LTI macromodels via Hamiltonian-driven multivariate adaptive sampling," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 39, no. 1, pp. 225–238, Jan. 2020.

[13] A. Zanco and S. Grivet-Talocia, "Toward fully automated high-dimensional parameterized macromodeling," *IEEE Trans. Compon., Packag., Manuf. Technol.*, vol. 11, no. 9, pp. 1402–1416, Sep. 2021.

[14] T. Bradde, S. Grivet-Talocia, A. Zanco, and G. C. Calafiore, "Data-driven extraction of uniformly stable and passive parameterized macromodels," *IEEE Access*, vol. 10, pp. 15786–15804, 2022.

[15] S. B. Olivadese and S. Grivet-Talocia, "Compressed passive macromodeling," *IEEE Trans. Compon., Packag., Manuf. Technol.*, vol. 2, no. 8, pp. 1378–1388, Aug. 2012.

[16] M. De Stefano, T. Wendt, C. Yang, S. Grivet-Talocia, and C. Schuster, "Regularized and compressed large-scale rational macromodeling: Theory and application to energy-selective shielding enclosures," *IEEE Trans. Electromagn. Compat.*, vol. 64, no. 5, pp. 1365–1379, Oct. 2022.
[17] N. Halko, P. G. Martinsson, and J. A. Tropp, "Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions," *SIAM Rev.*, vol. 53, no. 2, pp. 217–288, Jan. 2011.
[18] P. Bratley and B. L. Fox, "Algorithm 659: Implementing sobol's quasirandom sequence generator," *ACM Trans. Math. Softw.*, vol. 14, no. 1, pp. 88–100, Mar. 1988.
[19] A. Carlucci, S. Grivet-Talocia, S. Mongrain, S. Kulasekaran, and K. Radhakrishnan, "Towards accelerated transient solvers for full system power integrity verification," in *Proc. IEEE 31st Conf. Electr. Perform. Electron. Packag. Syst. (EPEPS)*, San Jose, CA, USA, Oct. 2022, pp. 1–3.



Antonio Carlucci (Graduate Student 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 focuses on large-scale simulation of electronic systems using macromodeling methods.

Mr. Carlucci received the Best Student Paper Award from SPI 2023, the 27th IEEE Workshop on Signal and Power Integrity.

Tommaso Bradde (Member, IEEE) received the B.Sc. degree in electronic engineering from Rome Tre University, Rome, Italy, in 2015, and the M.Sc. degree in mechatronic engineering and the Ph.D. degree in electrical, electronic and communications engineering from the Politecnico di Torino, Turin, Italy, in 2018 and 2022, respectively.

He is currently a Researcher and an Assistant Professor with the Politecnico di Torino. His current research is focused on data-driven parameterized macromodeling and its applications to system level signal and power integrity assessments, with the inclusion of active devices.

Dr. Bradde was a co-recipient of the 2018 Best Paper Award from the IEEE International Symposium on Electromagnetic Compatibility, the 2020 and 2022 Best Paper Awards from the IEEE Conference on Electrical Performance of Electronic Packaging and Systems, and the Best Student Paper Award from the 23rd IEEE Workshop on Signal and Power Integrity.

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, in 2007, serving as the President until its acquisition by CST in 2016. He has authored about 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.

Dr. Grivet-Talocia was a co-recipient of the 2007 Best Paper Award from the IEEE TRANSACTIONS ON ADVANCED PACKAGING. He received the IBM Shared University Research Award in 2007, 2008, and 2009. He was the General Chair of the 20th and 21st IEEE Workshops on Signal and Power Integrity (SPI2016 and SPI2017). He was an Associate Editor of the IEEE TRANSACTIONS ON ELECTROMAGNETIC COMPATIBILITY from 1999 to 2001 and he is currently serving as an Associate Editor for the IEEE TRANSACTIONS ON COMPONENTS, PACKAGING AND MANUFACTURING TECHNOLOGY.

Scott Mongrain, photograph and biography not available at the time of publication.

Sid Kulasekaran, photograph and biography not available at the time of publication.

Kaladhar Radhakrishnan (Senior Member, IEEE) received the Ph.D. degree in electrical engineering from the University of Illinois at Urbana-Champaign, Champaign, IL, USA.

He joined Intel in 2000. He is an Intel Fellow and a Power Delivery Architect with the Technology Development group at Intel. He has played a significant role in shaping and driving power delivery technologies for Intel microprocessors. He has authored four book chapters, over 50 technical papers in peer reviewed journals, and has been awarded 40 US patents. His areas of expertise are in integrated voltage regulators, advanced packaging and passives technologies.

Dr. Radhakrishnan is a two-time recipient of the Intel Achievement Award.

Open Access funding provided by 'Politecnico di Torino' within the CRUI CARE Agreement