POLITECNICO DI TORINO
Repository ISTITUZIONALE

Macromodeling of I/O Buffers via Compressed Tensor Representations and Rational Approximations

Terms of use:

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

(Article begins on next page)

10 April 2024

Macromodeling of I/O Buffers via Compressed Tensor Representations and Rational Approximations

Gianni Signorini, *Member, IEEE,* Claudio Siviero, *Member, IEEE,* Stefano Grivet Talocia, *Senior Member, IEEE,* Igor S. Stievano, *Senior Member, IEEE*

Abstract—This paper addresses the generation of accurate and efficient macromodels of high-speed input/output buffers. The proposed modeling approach extends the state-of-the-art methods that are currently available, yielding to a modular and scalable tool for model generation. The modeling procedure applies to both single-ended and differential devices, possibly exhibiting a rich dynamical behavior due to large supply fluctuations or internal voltage regulators. The models are defined by the combination of static surfaces described via compact tensor approximations and linear dynamical state-space relations generated using a robust time-domain vector fitting algorithm. A simple and effective solution is adopted to account for the overclocking operation of output buffer models as well. The feasibility and strength of the proposed method are demonstrated using real devices and complex application test-cases for signal and power integrity co-simulations.

I. INTRODUCTION

The amount of data being transferred across different components of the latest computing platforms is continuously growing. High-speed wired communication interfaces have been constantly improving over the last years, increasing transmission data-rate, minimizing pin-count and reducing power consumption; this is particularly true in mobile platforms (cellular phones, smart-phones, tablets, etc.), where more specialized communication protocols have been recently designed for a further optimization of power efficiency and prevention of Electromagnetic Interference (EMI). These interfaces can support different speeds and voltage levels, adjusted according to the target application (communication to other chips, cameras, displays, batteries, transceivers, etc.). The high-level of integration in state-of-the-art Printed Circuit Boards (PCB), Packages, Systems-on-a-Package (SiP) and Systems-on-Chip (SoC), combined with the inevitable presence of resistive, inductive and capacitive parasitic components of the interconnection structures, often leads to severe system performance degradation.

Signal and Power Integrity simulations are required to study the impact of interconnection non-idealities on communication reliability (electrical levels, signal distortions, power-supply fluctuations, Bit-Error-Rate (BER), etc.). Macromodel-based simulations appear to be the only viable approach to deal

Gianni Signorini is with Intel Corporation, Munich, Germany (e-mail: gianni.signorini@intel.com). Claudio Siviero, Igor S. Stievano and Stefano Grivet-Talocia are with the Dept. Electronics and Telecommunications, Politecnico di Torino, Torino, Italy (e-mail: {claudio.siviero,igor.stievano,stefano.grivet}@polito.it).

with the complexity of such analyses: transmitter and receiver circuits (I/Os), usually described by detailed transistor-level netlists, are substituted with accurate and efficient equivalent representations; simulations run much faster, accuracy is guaranteed and system-level verification coverage can be extended. However, I/O macromodeling has become more and more a challenging task: the increasing communication speed (up to 10Gbps) and the reduction of signal amplitudes require outstanding model accuracy; furthermore, several detrimental effects can be induced by supply-voltage fluctuations, originated by the combination of power distribution network (PDN) non-idealities and I/O dynamic current consumption.

Nowadays, the standard approach for I/O buffer modeling is offered by the Input/Output Buffer Information Specification (IBIS) [1]. IBIS suggests simplified circuit equivalents of typical buffer structures and provides detailed guidelines for the collection of relevant device features via a ready-to-use extraction procedure (*e.g.,* the static characteristics of the output port current, the equivalent capacitance of the silicon die, ...). This specification has been massively used for generating buffer models and it has been continuously updated with additional features and enhancements, becoming a *de-facto* standard. In spite of the widely recognized importance and diffusion of IBIS, some specific features of modern I/O devices cannot be accurately reproduced; mostly, the inaccuracies appear to be related with the power supply currents and the dynamic dependence of I/O and supply voltages on circuit behavior.

In literature, other approaches that complement IBIS and provide improved model accuracy with reasonable efficiency are available [2], [3], [4], [5], [6]. However, all these approaches do not offer a conclusive reliable framework that accommodates for both single ended and differential technologies, and providing a robust modeling procedure that guarantees an adequate level of accuracy for high-performance devices in complex simulation scenarios.

Enhanced model structures have been recently proposed, based on custom solutions aimed at providing improved model accuracy, accounting for temperature or supply variations, and to support overclocking [7], [8]. Inspired by the IBIS philosophy, these approaches suggest to modify specific blocks that demand for refinements, but still show critical issues. Mainly, in [7], the proposed modeling efforts are limited to the output port current only, hence preventing the use of the models for SI/PI co-simulations in which an extreme accuracy at output and supply port models is required. In [8], a gray-

box procedure is adopted: internal nodes of the transistor-level descriptions of the modeled devices must be extracted and suitably driven. Hence, this procedure is non-general and intrusive, requiring intimate knowledge of internal device descriptions. As an example, in order to prevent any further attempt of separating device blocks, in several design examples and in post-layout netlists the pre-driver and the driver stages are melt together, thus making a gray-box approach unfeasible.

This paper suggests a more general and modular modeling alternative that is able to meet all the behavioral requirements, and that requires the observation of only external device port responses. The model structure is defined by either single- or two-piece representations, that combine multivariate *static* surfaces and linear dynamical state-space subsystems. The static parts are conveniently approximated by compressed tensor approximations, thus facilitating their SPICE equivalent synthesis that can run in any commercial circuit simulation tool. The dynamical parts are identified using robust time-domain vector fitting algorithms. Overclocking operation is supported and handled by imitating the behavior of the real switching mechanism of output buffers via a simple yet reliable solution. Once implemented, the models offer remarkable accuracy and good efficiency figures.

Section II outlines the key features of the current state-of-the-art modeling approaches, mainly discussing accuracy limitations shared the different methods. Section III introduces the proposed model structure and its basic building blocks. Details on the compact representations adopted for the approximation of the multivariate surfaces is provided in Sec. IV. Section V summarizes the step-by-step procedure for model generation; Sec. VI discusses the validity of the proposed method, demonstrated with practical examples that use real I/O buffers in complex application test-cases. Final remarks and conclusions are given in Sec. VII.

II. STATE-OF-THE ART MACROMODELS

For the sake of conciseness, the discussion is based on a generic single-ended transceiver topology shown in Fig. 1. All the comments and remarks are however general and applicable to other devices like differential transceivers or having multiple supply terminals.

In the above scheme, the block labeled as "I/O" represents a typical buffer structure consisting of a number of cascaded inverter gates, aimed at decoupling the internal logic part (here identified by the signal v_{in}) to the external interconnect. This structure interacts with its environment through the I/O and supply terminals via the port variables (v, i) and (v_{dd}, i_{dd}), respectively. When the transceiver operates in driver mode, the input signal $v_{in}(t)$ comes into play and behaves as a digital (binary) control signal for the output stage, triggering rising and falling state transition events. In addition, due to the inherent decoupling feature of buffer circuits, small dynamical variations of v_{in} (or i_{in}) implies only negligible variations on the other variables (e.g., v and i). Furthermore, in SI/PI co-simulations, the input pin is usually driven by an ideal voltage source that mimics a transmitted bit-stream using a trapezoidal waveform. The above observations indicate that any attempt

to model the input port in a more detailed way would just increase model complexity, reduce efficiency, and would not provide any significant benefit in the resulting SI/PI simulation accuracy.
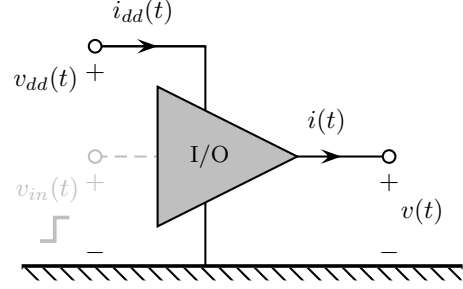

Fig. 1: Generic single-ended I/O buffer with its relevant port electrical variables.

Summarizing, for both input and output buffers, a macro-model is any relation mimicking the nonlinear dynamic behavior of the port currents $i(t)$ and $i_{dd}(t)$ as functions of the port voltages:

$$\begin{cases} i(t) & = & f(v, v_{dd}, v_{in}; \mathsf{D}) \\ i_{dd}(t) & = & f_{dd}(v, v_{dd}, v_{in}; \mathsf{D}) \end{cases} \tag{1}$$

where f and f_{dd} are single- or multi-piece relations either defined by simplified equivalent circuits or black-box mathematical models and D denotes the time derivative operator that possibly applies to all the involved variables. The above admittance-like relation is appropriate under the assumption of a voltage-controlled device operation.

A. Model structure

Most of the state-of-the-art approaches, including IBIS [1], Mπlog [2] and others [3], [5], [6] suggest a two-piece structure for the macromodeling of output buffers, that for the current at output and supply ports writes:

$$\begin{cases} i(t) & = & w_H(t, v_{dd}) f_H(v, v_{dd}; \mathsf{D}) + \\ & + & w_L(t, v_{dd}) f_L(v, v_{dd}; \mathsf{D}) \\ i_{dd}(t) & = & w_H(t, v_{dd}) f_{dd,H}(v, v_{dd}; \mathsf{D}) + \\ & + & w_L(t, v_{dd}) f_{dd,L}(v, v_{dd}; \mathsf{D}) + \delta_i(t, v_{dd}) \end{cases} \tag{2}$$

where w_ν are weighting signals playing the role of the unmeasurable input voltage v_{in}, f_ν and $f_{dd,\nu}$ are nonlinear dynamical submodels defined by either simplified circuits or parametric relations, and accounting for the device operation in the fixed high ($\nu = H$) or low ($\nu = L$) logic states. In the expression of the supply current i_{dd}, the additive term $\delta_i(t)$ accounts for the crowbar current drawn by the supply terminal during state switching that does not contribute to the output current i.

The rationale of the above two-piece representation resides in the inherent digital nature of output buffers, that most of the time operate in a fixed logic state. This simplification facilitates the estimation of model parameters from the observation of the external port transient responses only and has

been successfully used in real application scenarios, yielding accurate and efficient models (*e.g.,* [9]).

Specifically, the parameters defining the submodels f_ν are computed applying fitting techniques to a suitable set of output port voltage and current responses, obtained when the device is forced in the fixed $\nu = \{H, L\}$ logic states. Once f_ν are completely identified, the weighting coefficients w_ν are computed via linear inversion of (2) from a set of switching voltage $v(t)$ and current $i(t)$ waveforms recorded during low-to-high and high-to-low state transition events. As an example, Fig. 2 provides a graphical illustration of the complete weighting signal $w_H(t)$ needed to reproduce arbitrary bit streams (`0101' in the figure) via a suitable concatenation of the basic up ($\uparrow$) and down ($\downarrow$) weighting signals, respectively.
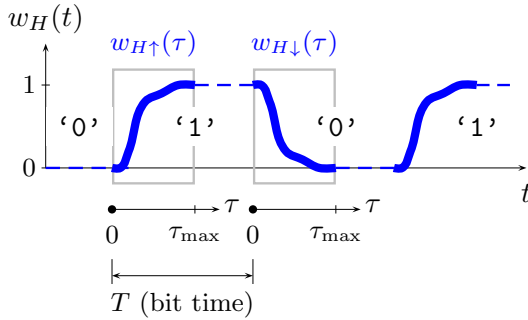

Fig. 2: Schematic illustration of the juxtaposition in time of the basic up ($w_{H\uparrow}(\tau)$) and down ($w_{H\downarrow}(\tau)$) weighting functions for the generation of the complete signal $w_H(t)$ accounting for a generic bit pattern (`0101' in the figure).

Some efforts have been spent to replace the two-piece representation (2) with a single piece model including $v_{in}(t)$ explicitly (*e.g.,* see [4]), and on the application of general purpose neural networks techniques. However, even if some good and promising results have been obtained, these modeling approaches still have delicate aspects that require special care and expert users; the methods are not easily scalable to a higher number of ports, and a SPICE synthesis of the corresponding model equations is non-trivial. Furthermore, the accuracy of the models heavily depends on the definition of excitation stimuli, potentially leading to spurious dynamics. As a final remark, it is important to notice that, in all the state-of-the-art approaches, input buffers are simply modeled via single-piece relation (*e.g.,* equation (2) with $w_H = 1$, $w_L = 0$ and $f_H = f_L$).

B. IBIS and Mπlog basics

Without loss of generality, the discussion below is based on the complementary approaches IBIS and Mπlog, with the aim of highlighting the main features and limitations of the different modeling tools.

IBIS suggests a topological buffer modeling in which the basic building block refers to the behavior of a CMOS inverter [1]. Suitable modifications have been introduced to represent other features or device technologies. For the output port of single-ended devices, the two-piece structure (2) is

used, where f_ν are suitable approximations of the multivariate static characteristic $F_\nu(v, v_{dd})$ of the buffer in the fixed state $\nu = \{H, L\}$. Specifically,

$$f_\nu(v, v_{dd}) = k_\nu(v_{dd}) \cdot F_\nu(v, V_{DD}), \quad (3)$$

where the actual two-dimensional mapping F is computed at the nominal supply voltage value V_{DD}, and where $k_\nu(v_{dd})$ are tabular coefficients that modulate the effects of the v_{dd} variable on the static output currents [1]. The above equation separates the static effects of the two voltage variables v and v_{dd} and provides a simple viable solution that requires two curves only, thus simplifying the implementation of the above block into any commercial electronic design automation tool (*e.g.,* via an equivalent SPICE-like interpretation).

As an example, Fig. 3 shows the reference and the predicted static surfaces of the output port current of an example test case forced in the high output state. The example considered in this first comparison is a single-ended output buffer implemented in a leading-edge silicon CMOS technology and operating with nominal supply voltage $V_{DD} = 1.2\,$V. The figure clearly highlights the differences between the real and the approximated characteristics via (3), and justifies the static errors that possibly arise when both the output and the power supply voltage vary.

As far as the dynamical effects are considered, IBIS assumes a simple linear derivative term $C_{\text{comp}}dv/dt$ that complements (2) and that accounts for the dominant capacitive effect of the I/O pin. In addition, for state switching, the weighting coefficients $w_H(t)$ and $w_L(t)$ in (2) are computed for a fixed power supply value. Both the two assumptions above (*i.e.,* a capacitor only for the dynamical effects and weighing functions at nominal supply voltage) unavoidably lead to inaccuracies of the model responses, that will be illustrated on a more realistic example in the next section.

The Mπlog methodology adopts a similar but more general modeling strategy, where the main difference with respect to IBIS mainly arises from a different definition of the submodels f_ν that do not assume a specific, *i.e.,* rigid, structure [2]. System identification tools and general purpose parametric relations (such as sigmoidal- or radial-based models) have been successfully used to automatically take into account the simultaneous effects of both output and supply voltage variations [9]. The weighting coefficients w_ν are computed for the nominal value of the power supply as done in IBIS, and the effects of static supply variation is modeled by means of additional delays on the transition events with the information stored in look-up tables.

C. Performance and limitations

In order to stress the performance and limitations of the selected state-of-the-art models, the same example buffer of the previous section is here revisited. The device is supplied by either an ideal battery or a realistic power distribution network, and its output is connected to an ideal 50Ω-transmission line ($500\,$ps delay), terminated with a $2.5pF$ capacitor load. This scenario represents an hypothetical single-line memory link interconnection for low-power applications. Figure 4 focuses on
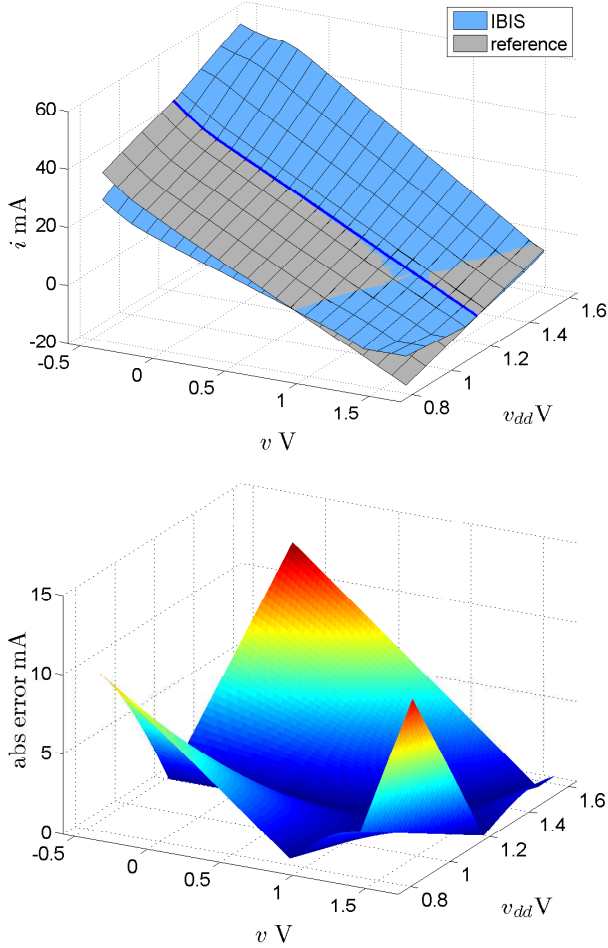
Fig. 3: Two-dimensional output static characteristic: transistor-level response (reference) and predicted surface via (3) in IBIS (top panel); absolute error surface (bottom panel).

Fig. 4: Output port voltage response $v(t)$ (top panel) and supply-current profile $i_{dd}(t)$ (bottom panel) of the example output buffer supplied with an ideal voltage source that assumes 90% (blue), 100% (black), 110% (red) of the nominal V_{DD} (1.2 V). The buffer produces a '010' bit pattern on a transmission line load terminated by a $C_L = 2.5$ pF capacitor. Solid lines: reference; dashed curves: IBIS.

the ideal supply case, where the device performs a logic-state transition a '010'; the reference transistor level response is compared with the predictions obtained using IBIS models; Figure 5 reports the same comparison done with Mπlog [2] models. The top and the bottom panels report, respectively, the near-end output voltage and the supply current, obtained using different static supply voltage values, *i.e.,* 90%, 100% and 110% of the nominal V_{DD}, respectively. The curves highlight the impact of static supply voltage variations on output and supply-current, both in terms of attenuation/amplification and transition delays; Mπlog models demonstrate an improved behavior compared to IBIS, but the accuracy of both the representations appears unsatisfactory.

As a second and more interesting test, Fig. 6 shows the reference and the model responses when a non-ideal supply is used. The above comparison clearly stresses the lack of accuracy of both the IBIS and the Mπlog models.

Also, even if Mπlog models offer a better prediction, the specific assumptions on the model structures discussed in the previous section and that are in part shared by the two approaches unavoidably lead to spurious dynamics. Mainly, the rough approximations of the multivariate buffer static
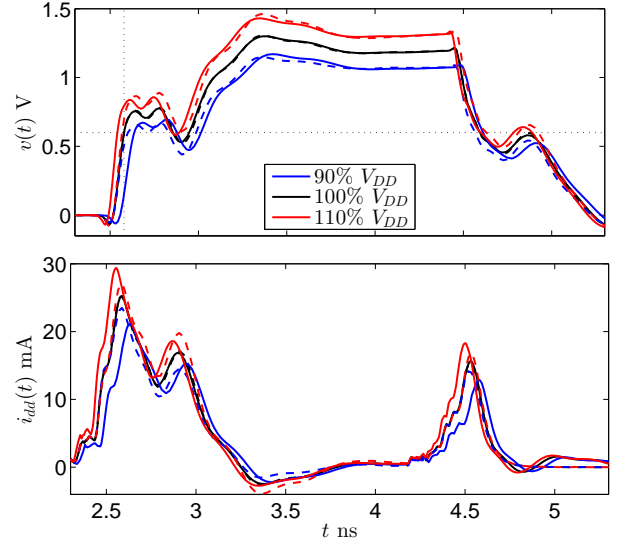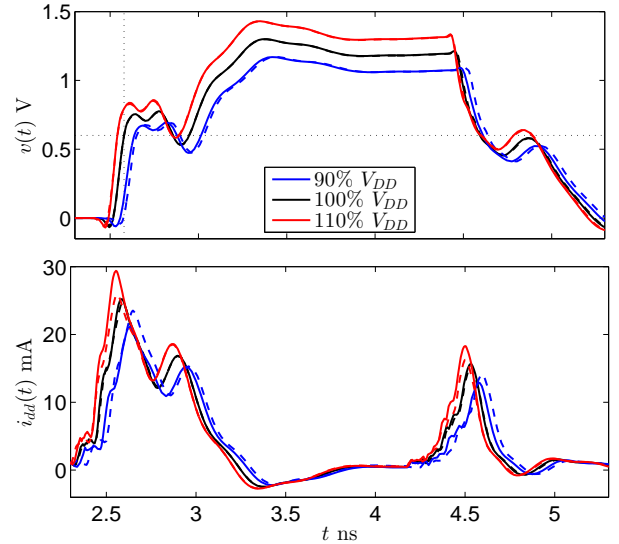

Fig. 5: Output port voltage response $v(t)$ (top panel) and supply-current profile $i_{dd}(t)$ (bottom panel) of the example output buffer supplied with an ideal voltage source that assumes 90% (blue), 100% (black), 110% (red) of the nominal V_{DD} (1.2 V). The buffer produces a '010' bit pattern on a transmission line load terminated by a $C_L = 2.5$ pF capacitor. Solid lines: reference; dashed curves: Mπlog [2].
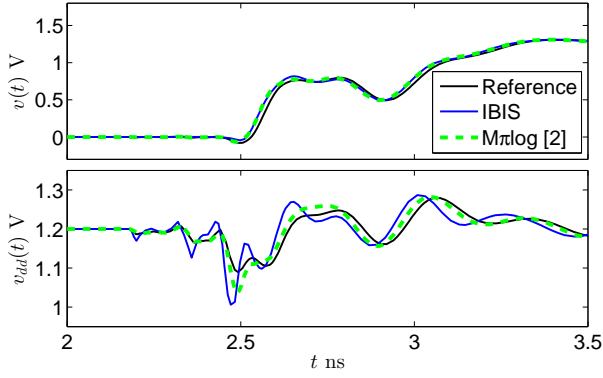
Fig. 6: Output port voltage response of the example output buffer supplied by a real power distribution network. Solid black lines: reference; solid blue curves: IBIS; dashed green curves: Mπlog [2].

characteristics and the possible limitations in the inclusion of the power supply effects on both the state switching and the dynamical submodels at fixed state do not allow to mimic the buffer behavior accurately, demanding for model enhancements.

It is important to notice that [7] proposes a recent approach that partially addresses the aforementioned issues and provides improved accuracy. The model enhancements are focused on the static characterization of the output port current via system identification toolboxes and full multivariate representations in place of (3). However, other critical key aspects like dynamic supply-current and supply-voltage dynamic dependency have not been addressed yet and still need to be carefully taken into account in order to yield accurate models for reliable model-based SI/PI co-simulations.

III. ENHANCED MACROMODELS

This section proposes a model generalization aimed at overcoming the main limitations outlined in the previous section, *i.e.,* the need of multivariate approximation of device static characteristics, the inclusion of the v_{dd} variable in the weighting coefficients, and the modeling of the possible simultaneous dynamical effects of the port variables v and v_{dd} on the buffer behavior.

The same two-piece representation (2) is used, where the submodels $f_\nu(v, v_{dd}; \mathsf{D}), \nu = H, L$ are split into the sum of static and dynamic parts,

$$\begin{cases} f_\nu(v, v_{dd}; \mathsf{D}) & = F_\nu(v, v_{dd}) + g_\nu(v, v_{dd}, \mathsf{D}) \\ f_{dd,\nu}(v, v_{dd}; \mathsf{D}) & = F_{dd,\nu}(v, v_{dd}) + g_{dd,\nu}(v, v_{dd}, \mathsf{D}) \\ & \quad + \delta_i(t, v_{dd}) \end{cases}$$

(4)

1) *Multivariate Surfaces approximations.* Compact multivariate approximations are used to model the effects of additional variables (as v_{dd}) on both the static characteristics F_ν and the weighting functions w_ν, either for the output and supply port models. The latter also includes the crowbar term δ_i, that is handled similarly. It is important to notice that multivariate mappings have

already been used for the static parts only in [7]. The above solution, however, leads to models that still exhibit inaccuracies in reproducing the rich dynamical behavior of the port currents and the very complex influences of v_{dd} on switching characteristics. A major accuracy improvement is in fact provided by the explicit inclusion of the v_{dd} variable into the transient functions $w_\nu(t)$ and $\delta_i(t)$. As an example, according to Fig. 2, the basic up/down weighting functions and the crowbar current profile can be readily computed for different static power supply values and can be considered as multivariate surfaces that depend on the timebase τ (defined below) and v_{dd} variables (see Fig. 8 and 9).

2) *Rational-based dynamical models.* Rational approximations are used to mimic the dynamical behavior of the buffer in the fixed logic state (*i.e.,* , the terms g_ν in (4), here assumed to be LTI submodels). Among the large number of linear system identification methods, we adopt here the Time-Domain Vector Fitting (TD-VF) algorithm [14]. The main advantages offered by this method are its robustness and its ability to handle stiff systems, with poles having different orders of magnitude. This is a specific feature of many of the systems under consideration, as low-voltage differential transceivers with internal voltage regulator. The TD-VF identification algorithm processes a set of device stimuli and responses uniformly sampled over time. Therefore models g_ν are formulated in discrete-time $t_k = k\Delta t$, such that their output at a given time step t_k can be computed based on their past samples, as well as present and past input samples. It is important to remark that the use of linear rational models is adopted based on the analysis of a number of current application devices. For those devices, the residual non-linearity in the dynamical part of the response is negligible; linear rational models appear to be the best compromise to generate accurate models while using a well-defined and robust estimation procedure. However, the proposed modeling flow is general and if needed linear relations can be suitably replaced by non-linear parametric models estimated via standard system identification techniques [16], [17].

As an additional feature, the proposed model of output buffers suggests a simple yet effective mechanism to account for possible *overclocked device operation*; for *overclocked device operation* we intend the use of the model with an input bit pattern having a toggling period smaller than that maximum duration of up/down switching profiles and crowbar current signature. According to the concatenation scheme of Fig. 2, the values of up /down weighting functions are obtained via an internally-generated timebase variable $\tau(t)$. If a transition event occurs before the completion of the previous event (as highlighted by the blue box in Fig. 7), the τ variable is realigned to either the 0 or $\tau_{\max}$ values and the observation of the weighting function for that specific event starts again from the beginning. This solution can be easily implemented a simple analog circuit in any SPICE engine or coded using basic keywords of most diffused hardware

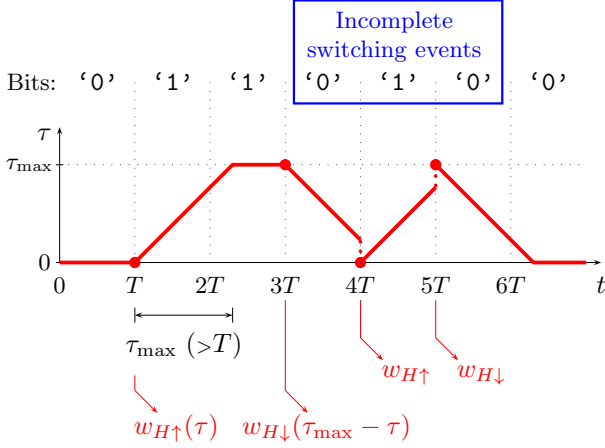description languages (*e.g.,* VHDL-AMS or Verilog-A).

Fig. 7: Mechanism used in the proposed models to generate the complete weighting signals ($w_H(t)$ in this example) accounting for incomplete switching events arising from *overclocked device operation*. The above scheme shows the timebase $\tau(t)$ used to drive either the basic up or down weighting coefficients $w_{H\uparrow}(\tau)$, $w_{H\downarrow}(\tau)$.

IV. SURFACE CHARACTERIZATION VIA TENSOR COMPRESSION

As outlined in the previous Section, a key aspect of the proposed modeling approach is the approximation of the device static characteristics and weighting functions by means of suitable multivariate compact representations. A number of alternative approximations have been proposed in the past, based on radial or sigmoidal basis functions, global or piecewise polynomials, or even more complex structures belonging to the class of neural networks. All these approaches, however, share the same limitations arising from the application of general nonlinear optimization methods, leading to possible inaccuracies and/or involving a large number of model components. We adopt here the alternative yet effective solution recently proposed in [10], based on the construction of empirical basis functions obtained from a set of "measurements" collected from reference transistor-level SPICE simulations. This approach is general and robust. In addition, it offers intrinsic compression capabilities, since a minimal number of basis functions is obtained via standard Singular Value Decomposition (SVD) or its higher-dimensional tensor generalization. What is even more important, the SPICE implementation of the model turns to be very efficient and achieved via standard voltage controlled sources.

A. Problem statement

Any of the aforementioned multivariate surfaces can be described in abstract notation as a multivariate map

$$y = F(\boldsymbol{x}), \qquad (5)$$

where the input vector $\boldsymbol{x} = (x_1, \ldots, x_N)^\mathsf{T}$ collects all port (output and supply) voltages, as well as additional parameters.

An accurate behavioral macromodel should capture the variation of the output variables y with respect to all components x_n. To this end, standard characterization approaches perform a set of DC simulations, by computing the static response of the device by fixing each of the independent variables $x_n = X_{j_n}$ with $j_n = 1, \ldots, J_n$ and $n = 1, \ldots, N$. The result is a multidimensional tensor $\mathcal{Y}$ with elements

$$Y_{j_1, \ldots, j_N} = F(X_{j_1}, \ldots, X_{j_N}) \qquad (6)$$

with N being the *order*, and J_n being the dimension (number of components) along the n-th *mode* or *direction*. The complexity of the resulting dataset, which includes $|\mathcal{Y}| = \prod_{n=1}^{N} J_n$ independent data points, must be reduced in order to obtain a tractable model.

Standard macromodeling approaches compute a parametric representation of $\mathcal{Y}$, *e.g.,* as a superposition of some multivariate basis functions, through some fitting process. Such an approach is however hardly scalable to orders N larger than 2 or 3 at most, due to a curse of dimensionality. The approach that we pursue in this paper is aimed at a dimensionality reduction of the tensor $\mathcal{Y}$, by seeking an approximate representation that compresses the data-set before proceeding to any subsequent parametric identification and/or approximation. The process is described for the case $N = 2$ in Sec. IV-B and extended to the general case in Sec. IV-C.

B. The two-dimensional case

This scenario applies, *e.g.,* to the case of a single-ended driver with a single power supply. In such case, the two independent variables are the output voltage $x_1 = v$ and the supply voltage $x_2 = v_{dd}$. A double DC sweep leads to a tensor data-set with order $N = 2$, which is nothing else than a matrix $\mathbf{Y} \in \mathbb{R}^{J_1 \times J_2}$, with elements Y_{j_1, j_2}. A well-known result in linear algebra states that the optimal approximation of $\mathbf{Y}$ with a matrix $\bar{\mathbf{Y}}$ having fixed rank ρ is provided by the truncated Singular Value Decomposition (SVD)

$$\mathbf{Y} \approx \bar{\mathbf{Y}} = \mathbf{U}_1 \boldsymbol{\Sigma} \mathbf{U}_2^\mathsf{T}, \qquad (7)$$

where $\boldsymbol{\Sigma} = \mathrm{diag}\{\sigma_1, \ldots, \sigma_\rho\}$ collects the largest ρ singular values. The columns $\boldsymbol{u}_{n,\ell_n}$ with $\ell_n = 1, \ldots, \rho$ of the two (orthogonal) matrices $\mathbf{U}_n \in \mathbb{R}^{J_n \times \rho}$ for $n = 1, 2$ collect the corresponding singular vectors. The above approximation minimizes the induced 2-norm of the residual $\|\bar{\mathbf{Y}} - \mathbf{Y}\|_2$ and is therefore optimal [11].

C. The general case

Here, we discuss how to generalize the SVD-based approximation (7) to a generic higher order $N > 2$. We start by rewriting (7) in the more abstract form

$$\mathbf{Y} \approx \bar{\mathbf{Y}} = \boldsymbol{\Sigma} \times_1 \mathbf{U}_1 \times_2 \mathbf{U}_2, \qquad (8)$$

where the operator $\times_n$ performs matrix multiplication along the n-th direction (here, $n = 1$ for rows and $n = 2$ for columns). The generalization of (8) to approximate a given tensor $\mathcal{Y}$ with order $N > 2$ is straightforward [12], [13], as

$$\boldsymbol{\mathcal{Y}} \approx \bar{\boldsymbol{\mathcal{Y}}} = \boldsymbol{\mathcal{S}} \times_1 \mathbf{U}_1 \times_2 \mathbf{U}_2 \cdots \times_N \mathbf{U}_N \qquad (9)$$

where the orthogonal matrices $\mathbf{U}_n \in \mathbb{R}^{J_n \times \rho_n}$ for $n = 1, \ldots, N$ multiply the *core* tensor $\boldsymbol{\mathcal{S}} \in \mathbb{R}^{\rho_1 \times \rho_2 \cdots \times \rho_N}$ along the n-th direction. Note that, differently from (8), the core tensor $\boldsymbol{\mathcal{S}}$ is in general full and can be characterized by a different size ρ_n along each direction. The columns $\boldsymbol{u}_{n,\ell_n}$ of each matrix $\mathbf{U}_n$ can be interpreted as an orthogonal basis of the subspace that approximates the collection all vectors obtained by freezing all indexes of $\boldsymbol{\mathcal{Y}}$ except along the n-th direction (also called n-th mode *fibers*). The component-wise expansion of (9) reads

$$\bar{Y}_{j_1, j_2, \ldots, j_N} = \sum_{\ell_1 = 1}^{\rho_1} \sum_{\ell_2 = 1}^{\rho_2} \cdots \sum_{\ell_N = 1}^{\rho_N} S_{\ell_1, \ell_2, \ldots, \ell_N}$$
$$(\mathbf{U}_1)_{j_1, \ell_1} (\mathbf{U}_2)_{j_2, \ell_2} \ldots (\mathbf{U}_N)_{j_N, \ell_N} \qquad (10)$$

This expression shows that the original tensor $\boldsymbol{\mathcal{Y}}$ is represented by a much smaller tensor $\boldsymbol{\mathcal{S}}$ with $|\boldsymbol{\mathcal{S}}| = \prod_{n=1}^{N} \rho_n$, plus a collection of N basis sets, each having ρ_n vector elements. An effective data compression is achieved if $\rho_n \ll J_n$ for each direction n. The quality of the approximation can be measured by the Frobenius norm, defined as

$$\|\bar{\boldsymbol{\mathcal{Y}}} - \boldsymbol{\mathcal{Y}}\|_F^2 = \sum_{j_1, \ldots, j_N} |\bar{Y}_{j_1, j_2, \ldots, j_N} - Y_{j_1, j_2, \ldots, j_N}|^2 \qquad (11)$$

The computation of (9) is here performed according to an Alternating Least Squares (ALS) algorithm [12], which refines an initial estimate of the matrices $\mathbf{U}_n$ by iterative re-projection of the original tensor along the subspaces available from previous iterations.

D. Static model construction

Once the approximation (9) is available, the components of $\mathbf{U}_n$ are combined with the corresponding input parameter values to construct a collection of one-dimensional data-sets $\Omega_{n,\ell_n} = \{[X_{j_n}, (\mathbf{U}_n)_{j_n, \ell_n}], \ j_n = 1, \ldots, J_n\}$, with one data-set for each $n = 1, \ldots, N$ and $\ell_n = 1, \ldots, \rho_n$. A corresponding parametric submodel $\varphi_{n,\ell_n}(x_n)$ is obtained through a piecewise linear interpolation process applied to Ω_{n,ℓ_n}, and the approximation to the original map (5) is constructed as

$$y \approx \sum_{\ell_1 = 1}^{\rho_1} \cdots \sum_{\ell_N = 1}^{\rho_N} S_{\ell_1, \ldots, \ell_N} \varphi_{1,\ell_1}(x_1) \ldots \varphi_{N,\ell_N}(x_N). \qquad (12)$$

This result is an approximation based on one-dimensional sub-models, assembled through a multidimensional tensor product, with coefficients available from the core tensor $\boldsymbol{\mathcal{S}}$. Due to the limited number of such coefficients, an equivalent circuit implementation of (12) becomes viable through behavioral voltage-controlled current sources.

E. Complexity

As far as the compactness is concerned, Tab. I collects the total number of points defining the original surfaces shown in Fig. 3, 8 and 9 and the corresponding SVD-based approximations. Specifically, for the latter case, the number of entries of $\boldsymbol{\Sigma}$, $\mathbf{U}_1$ and $\mathbf{U}_2$ matrices in (7) is reported. Furthermore, the fourth column of the table includes the information on the ratio between the above two numbers, and provides a compact information about the residual reduced complexity of the proposed SVD-based approximations. The proposed algorithm works as follows: (i) the raw data defining the surface (*i.e.,* matrix $\mathbf{Y}$) is pre-processed via a decimation algorithm that selects a smaller set of points generating a piece-wise approximation of the original surface with a given accuracy; (ii) the piece-wise approximation of the previous step feeds the method of Sec. IV-B. In both the two steps above, the relative accuracy selected during the modeling process is such to guarantee a given maximum relative error (e.g., 10^{-3}) across the entire surface. Summarizing, Tab. I highlights an overall saving on the number of original points, whose impact significantly contributes to accelerate the model-based simulations.

TABLE I: Complexity of the proposed SVD approximation obtained using a maximum relative approximation error of 10^{-3}.

Surface	# original points	# SVD points	complexity (%)
Fig. 3	2500	189	8.0
Fig. 8	74988	2475	3.3
Fig. 9	74988	12456	16.6

V. MODELING PROCESS

The key steps of the proposed modeling procedure are summarized below. The discussion takes as reference single-ended devices, but similar concepts are applicable to differential drivers as well, with the addition of a further dimension introduced by the presence of the complementary output pin.

1) Fig. 10 (a) shows the setup required to collect the data for the static characteristic identification. The output and supply pin of the device-under-modeling (DUM) are connected to independent .DC sources, V and V_{dd} respectively, whose values are swept within suitable ranges, $[V_{min}, ..., V_{max}]$ and $[V_{DDmin}, ..., V_{DDmax}]$ respectively. The sweep is repeated forcing a logic-low and a logic-high input state to the DUM. For each sweep point and each input logic-state, the current through output and supply ports are recorded, resulting in four current surfaces $I_\nu(V, V_{dd})$ and $I_{dd,\nu}(V, V_{dd})$. These current surfaces are properly stored in four matrices, $\mathbf{Y}_\nu, \mathbf{Y}_{dd,\nu} \in \mathbb{R}^{N \times M}$, where N and M are respectively the number of swept voltage values at output and supply ports. Each matrix is then approximated using the SVD process reported in Sec. IV. For example, for input logic-state $\nu = \{H, L\}$, the SVD approximation of the 3D output static characteristic $\mathbf{Y}_\nu(V, V_{dd})$ writes:

$$\mathbf{Y}_\nu(V, V_{dd}) \cong \bar{\mathbf{Y}}_\nu(V, V_{dd}) = \sum_{n=1}^{\rho} \sigma_n^\nu \varphi_n^\nu(V) \psi_n^\nu(V_{dd}). \quad (13)$$

The order ρ of the SVD approximation is defined for each surface in order to guarantee a target relative error constraint. 2) Fig. 10 (b) shows the setup required to collect the data for the rational approximation of the dynamic part via TD-VF algorithm. Multi-level noisy time-domain stimuli signals are defined for output and supply ports. These stimuli are applied to the DUM for both H and L input state. The resulting currents at output and supply ports are then recorded for the computation of their dynamical models. For example, for input logic state $\nu = \{H, L\}$, the output port TD-VF model $g_\nu(v, v_{dd}, \mathsf{D})$ is obtained from $y_\nu = i_\nu - F_\nu(v, v_{dd})$, where i_ν is the current response to the voltage stimuli v and v_{dd}, while F_ν corresponds to the static output current contribution. Models g_ν and $g_{dd,\nu}$ in (4) are expressed as multiple-input single-output (MISO) state-space representations

$$\begin{cases} \mathbf{x}(t_k) = A_\nu\mathbf{x}(t_{k-1}) + B_\nu\mathbf{u}(t_{k-1}) \\ \hat{y}_\nu(t_k) = C_\nu\mathbf{x}(t_k) + D_\nu\mathbf{u}(t_k) \end{cases} \quad (14)$$

for $\nu = \{H, L\}$ and $\mathbf{u} = [v, v_{dd}]^T$. For state-of-the-art I/O-buffers, very good fittings are obtained with a reduced order (up to 3 or 4); the model generation process automatically tunes the required numbers of poles in order to guarantee specific target accuracy values.

3) Fig. 10 (c) shows the setup required to collect the data for the switching function calculation. A digital waveform `010` is applied at the input pin of the DUM, supplied with a voltage V_{dd} and connected to an output load Z_i. This test is repeated using N values of V_{dd} linearly spaced in the range $[V_{DDmin}, ..., V_{DDmax}]$, once for three different loads $Z_i = [Z_A, \ Z_B \ , Z_C]$. For this set of three loads Z_i at each supply voltage V_{dd}, output voltages and currents are recorded and processed to calculate weighting functions $w_\nu(V_{dd}, t)$ by solving the following system in least squares sense

$$\begin{bmatrix} i_A(V_{dd}, t) \\ i_B(V_{dd}, t) \\ i_C(V_{dd}, t) \end{bmatrix} = \begin{bmatrix} f_H(v_A(t), V_{dd}) & f_L(v_A(t), V_{dd}) \\ f_H(v_B(t), V_{dd}) & f_L(v_B(t), V_{dd}) \\ f_H(v_C(t), V_{dd}) & f_L(v_C(t), V_{dd}) \end{bmatrix} \\ \cdot \begin{bmatrix} w_H(V_{dd}, t) \\ w_L(V_{dd}, t) \end{bmatrix} \quad (15)$$

Repeating this process for every V_{dd} value, 3D switching characteristics $w_\nu(v_{dd}, t)$ are obtained. Each surface can be then approximated using the SVD process in Sec. IV, subsequently used for the SPICE/Verilog-A equivalent synthesis. As an example, for the second single-ended output buffer introduced in Sec. VI, Fig. 8 shows the 3D surface representing the weighting coefficient $w_H(t, v_{dd})$ for a `01` (rising) event.

4) The crowbar supply-current component and its supply dependency can be calculated by re-using the results from the setup in Fig. 10 (c); since this current component is independent from the load, any of the results out of the three load configurations A, B or C can be used. For each supply
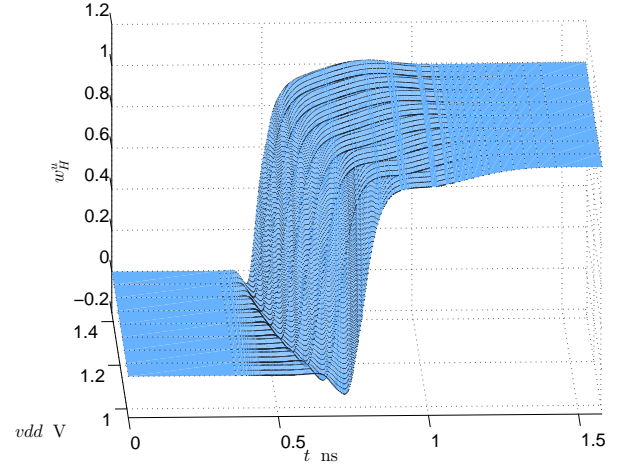

Fig. 8: 3D surface representing the weighting coefficient $w_H(t, v_{dd})$ for a `01` (rising) event.

voltage V_{dd}, selecting e.g. $Z_i = Z_C$, we can write the crowbar supply current $\delta_i(t, V_{dd})$ as:

$$\begin{aligned} \delta_i(t, V_{dd}) = \ & i_{dd,C}(t, V_{dd}) \\ & - f_{dd,H}(v_C(t), V_{dd}) \cdot w_H(t, V_{dd}) \\ & - f_{dd,L}(v_C(t), V_{dd}) \cdot w_L(t, V_{dd}). \end{aligned} \quad (16)$$

Repeating this calculation for each V_{dd} point, 3D crowbar supply-current characteristic $\delta_i(t, v_{dd})$ is calculated. As for the weighting functions, also this 3D-surface can be approximated using the SVD process in Sec. IV, subsequently implemented in SPICE or Verilog-A. Considering once again the device addressed by Fig. 8, a sample surface of the pre-driver/crowbar supply-current $\delta_i(t, v_{dd})$ for a `01` (rising) event is depicted in Fig. 9.
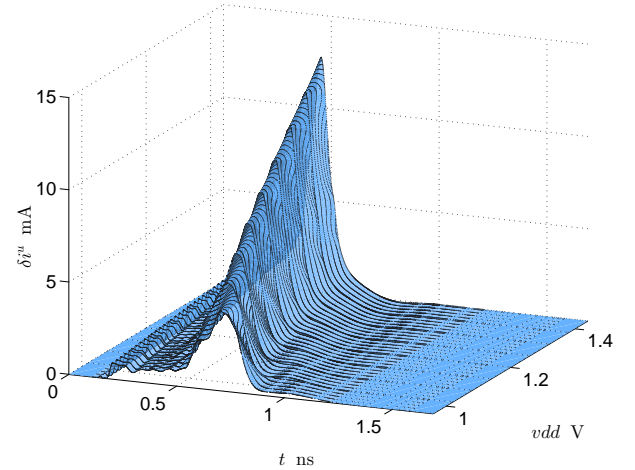

Fig. 9: 3D surface representing pre-driver/crowbar supply-current profile $\delta_i(t, v_{dd})$ for a `01` (rising) event.

The time required to complete the model generation depends on the number of points used in the nested .DC sweeps and in the number of V_{dd} points selected; in general, the simulation run-time required for DUM characterization increases with the complexity of the transistor-level netlist description of

the DUM (*e.g.,* including or not post-layout including RC-parasitics). For practical cases, the simulation run-time required for the modeling process ranges from minutes to a few hours with commercial SPICE solvers running on a workstation, depending on size and complexity of DUMs transistor-level netlist. Such simulation run-times are reasonable and should only be performed once for each DUM. Simulation data collection and post-processing (SVD, weighting functions calculations, etc.) requires only a few minutes on a generic laptop.

VI. APPLICATIONS

The effectiveness of the proposed macromodeling methodology is demonstrated on several examples from real state-of-the-art I/O-buffer and SI&PI simulations. Circuits are implemented on leading-edge CMOS technology, while the interconnection parasitics used in SI&PI scenarios are extracted from real Package/PCB for mobile platforms.

A. Single-ended test cases

In order to verify the improvement provided by the enhanced Mπlog model class (labeled eMπlog in the following) the single-ended I/O-buffer introduced in Sec. II-C has been modeled and the validation test with the buffer driven by the '010' pattern and supplied with different static voltage values has been repeated (see Fig. 5). The proposed models are obtained using a maximum relative static approximation error of 10^{-3}. Fig. 11 depicts the responses obtained with the eMπlog model (either with its SPICE or Verilog-A implementation) and clearly highlights the accuracy improvement provided by eMπlog.

A second commercial device for a high-speed low-power memory interface has been selected to stress the eMπlog modeling procedure with a more compelling benchmark. As a first validation test, the case presented in Sec. II-C with the power-supply port of the driver connected to a real power delivery network has been considered. Fig. 12 and its zoomed extract in Fig. 13 illustrate the outstanding accuracy of the derived eMπlog macromodel in reproducing DUM's supply current i_{dd} and output voltage v, with a significant improvement with respect to the corresponding IBIS responses. Figure 14 also shows the response of the former Mπlog model [2], thus confirming the benefits of the proposed enhancements.

As a second validation, a more complex SI&PI simulation has been performed using realistic interconnection parasitics on both signals and supply. Several instances of the driver are simultaneously switching and transmitting different bit-patterns. In Fig. 15, the responses of the eMπlog macromodels are again compared with the corresponding ones obtained using DUM's transistor-level netlist and IBIS models, confirming an outstanding accuracy.

B. Performance assessment

In order to provide a clearer picture on model performance, Table II quantifies model accuracy and efficiency as a function of three different maximum relative errors used in the approximation of eMπlog static surfaces (see the first column of the table). The identification testbench in Fig. 10(b) is considered in this comparison. The example device of Sec. VI-A is kept in the fixed high logic state and its output and supply ports are driven by multi-level time-domain stimuli. The second and third columns of Table II collect the root mean square errors (RMSE) computed from the difference between reference and model responses of the output and supply currents. The last column includes the CPU time required for the simulation of the different models. As expected, the numbers in the table confirm an improved model accuracy together with lower approximation errors and an unavoidable decrease of model efficiency.

In order to better understand the impact of the static approximation errors on dynamic model response, Fig. 16 shows the visual comparison between the reference and the aforementioned predicted supply current $i_{dd}(t)$. In the top panel, the black curve corresponds to the reference transistor-level response. The other curves in the top and bottom panels are obtained using the eMπlog models generated with a maximum relative error in the static approximations of, respectively, 10^{-1} (dashed red line), $5 \cdot 10^{-2}$ (dotted blue line) and 10^{-3} (dashed green line).

TABLE II: Model performance (second example of Sec. VI-A) for three different accuracies used in the approximation of the eMπlog static surfaces.

Maximum rel. error	RMSE i (μA)	RMSE i_{dd} (μA)	CPU time (.TRAN, SPICE)
10^{-1}	219	213	7 s
$5 \cdot 10^{-2}$	36	66	18 s
10^{-3}	32	34	1 min 21 s

As a final cross-comparison, Table III provides a summary of model performance for the second test case of Sec. VI-A: accuracy and efficiency of both IBIS and eMπlog models are reported. This comparison highlights the excellent performance (with user-controlled accuracy) of the proposed models, thus confirming that the eMπlog approach offers a very effective and promising alternative for SI/PI co-simulations. The eMπlog models are implemented into SPICE via either the classical interpretation of model equations in terms of standard circuit elements or as a Verilog-A metalanguage description. The latter yield simulation times that are comparable with IBIS (*i.e.,* 0.67 s vs 0.25 s), notwithstanding the obvious accuracy improvement. In any case, even for the classical circuit-based implementation, the speed-up appears to be good enough for enabling an accurate analysis of complex design scenarios. In addition, we remark that the overhead required by the eMπlog model generation according to the procedure described in Sec. V is negligible and it is comparable to the time required to generate IBIS models. The modeling procedure is carried out once and it is independent from the accuracy chosen for the the surface approximations.
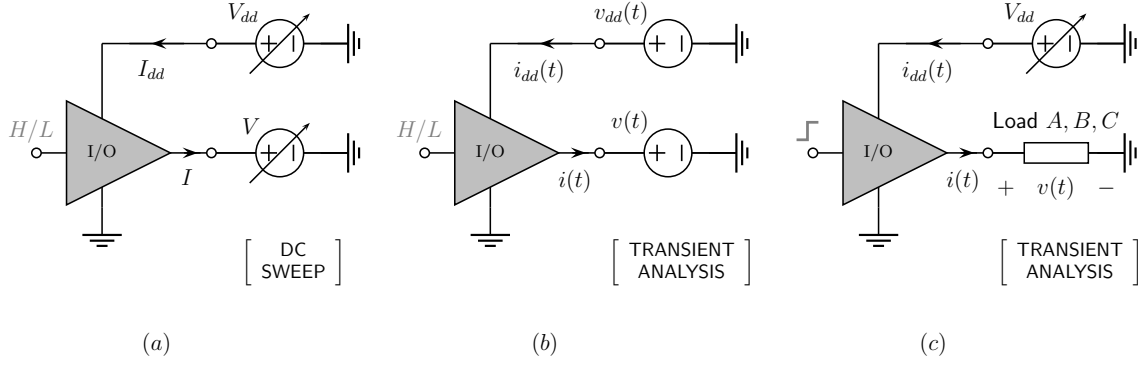
Fig. 10: Testbenches used for the collection of device responses required for the model generation.

TABLE III: Accuracy and efficiency of the models involved in the cross-comparison of Fig. 13

Device Model	Maximum rel. error	RMSE (mA)	CPU time (.TRAN, SPICE)	CPU time (.TRAN, Verilog-A)
Transistor-level	–	–	$\sim 2\,\mathrm{h}$	–
IBIS	–	2.68	0.25 s	–
eMπlog	10^{-1}	0.56	16 s	0.49 s
	$5\cdot 10^{-2}$	0.44	1 min 43 s	0.54 s
	10^{-3}	0.37	5 min 2 s	0.67 s
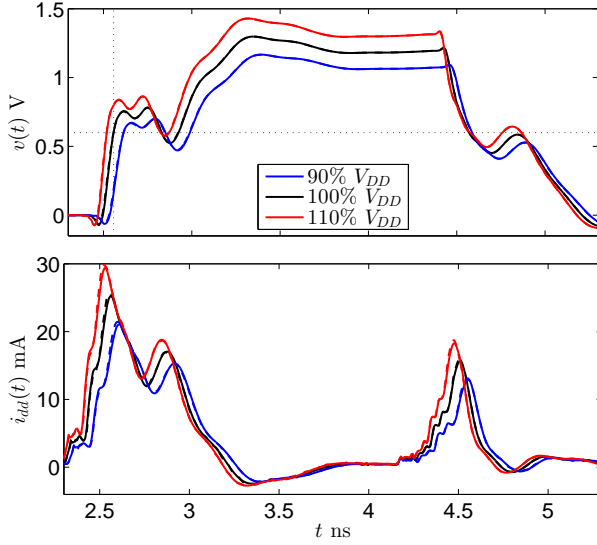
Fig. 11: Output port voltage response $v(t)$ (top panel) and supply-current profile $i_{dd}(t)$ (bottom panel) of the example output buffer supplied with an ideal voltage source that assumes 90% (blue), 100% (black), 110% (red) of the nominal V_{DD} (1.2 V). The buffer produces a '010' bit pattern on a transmission line load terminated by a $C_L = 2.5\,\mathrm{pF}$ capacitor. Solid lines: reference; dashed curves: eMπlog.

Fig. 12: Single-ended example: output port voltage and supply port voltage and current responses of the example driver connected to a distributed interconnect and supplied by a realistic power distribution network. Solid black lines: reference; solid blue curves: IBIS, dashed green curves: eMπlog.

C. Differential test case

The eMπlog macromodeling is here applied to a CMOS low-power high-speed voltage-mode differential I/O-buffer; this particular class of transceivers is commonly used in modern low-power differential serial links for mobile platforms
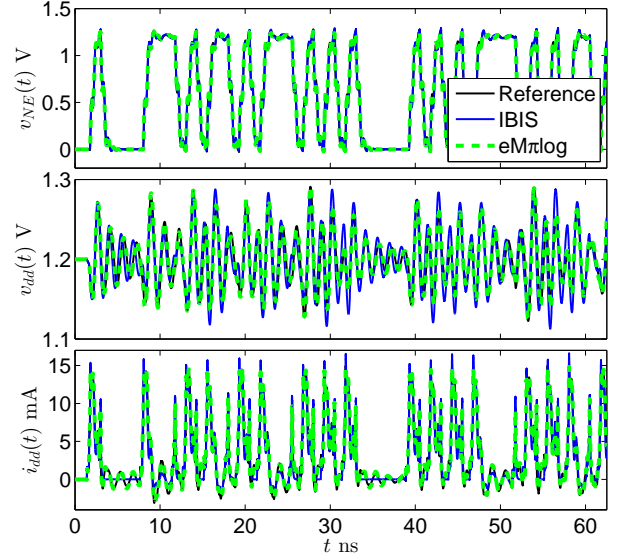
[15]. The low-power voltage-mode topology is characterized by the presence of an internal voltage regulator (LDO): this device provides a programmable supply-voltage to the output stage, resulting in tunable output swing and common-mode voltage levels. The internal LDO introduces rich dynamical components on output currents, consisting of both faster and slower time constants. The faster dynamics are related to
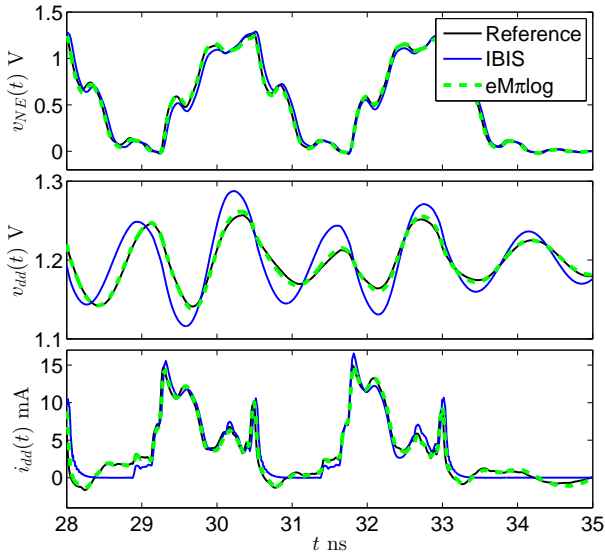
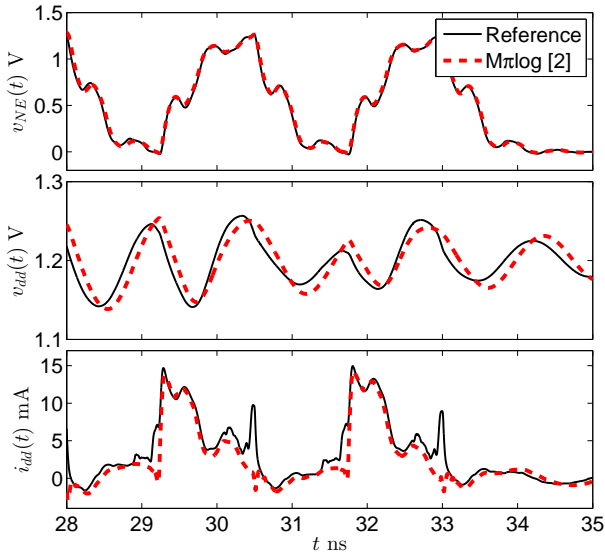Fig. 13: Single ended test case: close up view of the responses of Fig. 12.

Fig. 14: Single ended test case: close up view of the reference responses of Fig. 12 compared to the prediction obtained via the former Mπlog model [2].
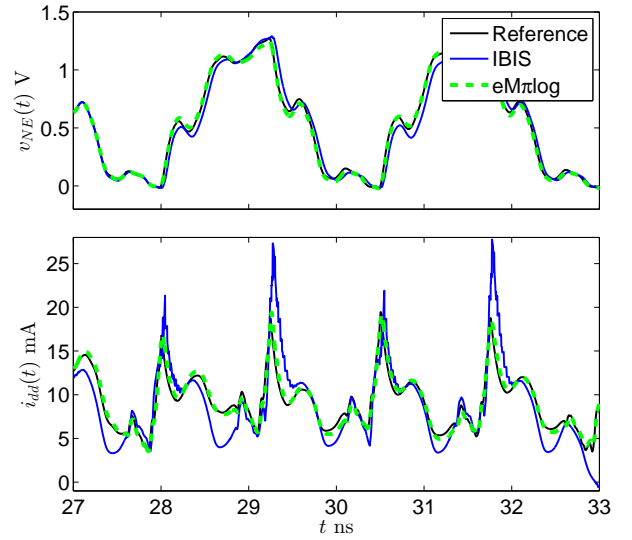

Fig. 15: Single-ended example: result of a second validation test involving the simultaneous switching activity of more drivers connected to a distributed interconnect and supplied by a common realistic power distribution network.
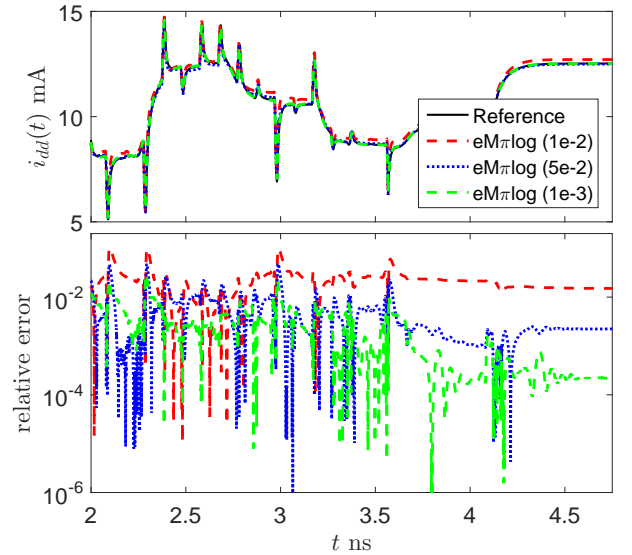

Fig. 16: Supply port current response of the second example device in Sec. VI-A as resulting from the identification setup of Fig. 10b (top panel) and its corresponding relative approximation error computed form the difference between the reference and the model responses (bottom panel). See text for details.

the parasitic capacitance of the output pins, while the slower dynamics are the result of feedback regulation mechanisms at LDO's output voltage, affected by bounces due to output switching activity. Thanks to the excellent fitting capabilities of the TD-VF algorithm, such complex dynamical responses can be correctly reproduced by eMπlog models (see Fig. 17).

A comparative signal integrity simulation has been performed by connecting DUM's transistor-level netlist, IBIS and eMπlog models (SPICE and Verilog-A implementations) to a differential channel, extracted from a PCB design of a tablet, terminated with a differential 100Ω resistor and $1.5pF$ single-ended capacitance; data-rate is fixed to 3Gbps. Fig. 18-19 report, respectively, the far-end single-ended components of

the differential signals ($V_{FE,P/N}$), and the far-end common-mode voltage $V_{FE,CM} = 0.5 \cdot (V_{FE,P} + V_{FE,N})$.

Single-ended signals clearly show the impact of LDO output bouncing on signal distortions and jitter. Models belonging to the eMπlog class show a superior accuracy compared to IBIS models, whose simplistic output capacitive approximation appears to be inadequate. LDO-induced slower dynamics on output signals are clearly visible on the far-end common-mode voltage: this phenomenon cannot be reproduced by IBIS models, while it appears to be well approximated by using
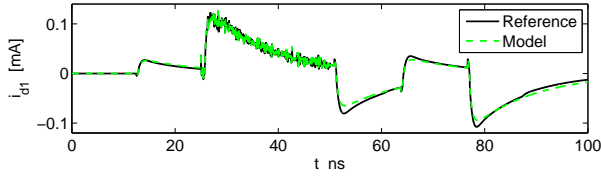
Fig. 17: Dynamic output-port current model for a differential driver with internal voltage regulator
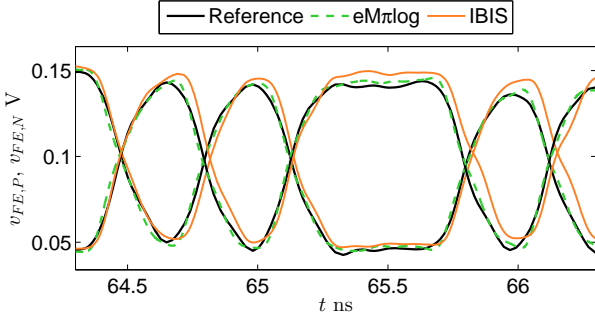

Fig. 18: Far-end single-ended output signals ($V_{FE,P/N}$) of a differential low-power voltage-mode driver with internal voltage regulator: Reference, eMπlog and IBIS responses.

eMπlog macromodels.

VII. CONCLUSIONS

This paper presented a systematic methodology for the generation of accurate and efficient Mπlog macromodels of advanced high-speed input/output buffers. Main state-of-the-art macromodeling techniques have been presented, and their limitations have been discussed and proven with example testcases. Innovative improvements have been applied to the
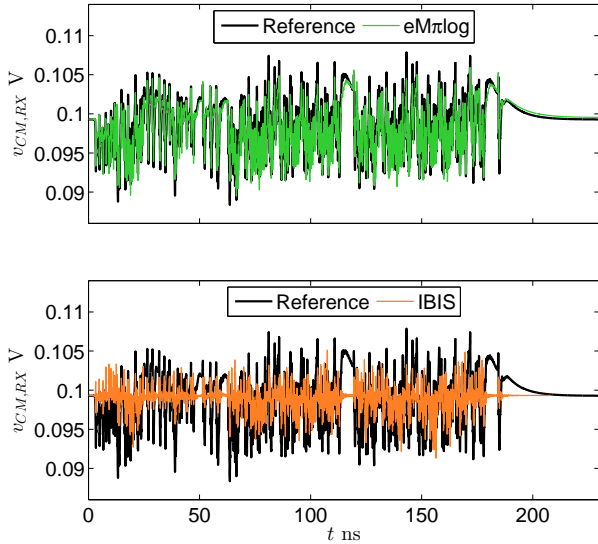

Fig. 19: Far-end common-mode voltage $V_{FE,CM}$: Reference (black), eMπlog (top panel, green), IBIS (bottom panel, orange) responses.

classic Mπlog macromodel structure, resulting in outstanding accuracy in reproducing device voltage and currents at both input/output and supply ports. Multivariate surfaces have been described via compact tensor approximations to reproduce static characteristics, weighting functions and supply-current profiles with their dependency on supply voltage; linear dynamical state-space relations are fitted and modeled using a robust time-domain vector fitting algorithm. The feasibility and the strength of the eMπlog macromodeling framework have been demonstrated using real industrial CMOS devices, both single-ended and differential, in complex application test-cases for system-level Signal and Power Integrity co-simulations.

REFERENCES

[1] IBIS (I/O Buffer Information Specification) – Ver. 5.1, http://www.eda.org/ibis/, Aug. 24, 2011.

[2] I. S. Stievano, I. A. Maio, and F. G. Canavero, "Mπlog, macromodeling via parametric identification of logic gates", *IEEE Trans. on Adv. Packag.*, Vol. 27, No. 1, pp. 15–23, Feb. 2004.

[3] B. Mutnury, M. Swaminathan, and J. P. Libous, "Macromodeling of nonlinear digital I/O drivers," *IEEE Trans. on Adv. Packag.*, Vol. 29, No. 1, pp. 102–113, Feb. 2006.

[4] Y. Cao, Q.-J. Zhang, "A New Training Approach for Robust Recurrent Neural-Network Modeling of Nonlinear Circuits", IEEE Transactions on Microwave Theory and Techniques, Vol. 57, No. 6, pp. 1539–1553, June 2009.

[5] A. K. Varma, M. Steer, and P. D. Franzon, "Improving Behavioral IO Buffer Modeling Based on IBIS", IEEE Trans. on Advanced Packaging, Vol. 31, No. 4, Nov. 2008.

[6] W. Dghais, T. R. Cunha, J. C. Pedro, "Reduced-Order Parametric Behavioral Model for Digital Buffers/Drivers With Physical Support", IEEE Trans. Compon., Packag. Manuf. Technol., Vol. 12, No. 12, Dec. 2012.

[7] T. Zhu, M. B. Steer, P. D. Franzon, "Accurate and Scalable IO Buffer Macromodel Based on Surrogate Modeling", IEEE Trans. Compon., Packag. Manuf. Technol., Vol. 1, No. 8, pp. 1240–1249, 2011.

[8] W. Dghais, T. R. Cunha, J. C. Pedro, "A Novel Two-Port Behavioral Model for I/O Buffer Overclocking Simulation", IEEE Trans. Compon., Packag. Manuf. Technol., Vol. 3, No. 10, October 2013.

[9] T. R. Cunha, H. M. Teixeira, J. C. Pedro, I. S. Stievano, L. Rigazio, F. G. Canavero, A. Girardi, R. Izzi, F. Vitale, "Validation by Measurements of an IC Modeling Approach for SiP Applications", IEEE Transactions on Components, Packaging and Manufacturing Technology, Vol. 1, No. 8, pp. 1214–1225, 2011.

[10] G. Signorini, C. Siviero, S. Grivet-Talocia, I. S. Stievano, "Power and Signal Integrity co-simulation via compressed macromodels of high-speed transceivers" Proc. of the *2015 IEEE 18th Workshop on Signal and Power Integrity (SPI)*, Berlin, Ge, May. 10-13, 2015.

[11] G. W. Stewart and J. Sun, *Matrix perturbation theory*, Academic Press, Boston, 1966.

[12] T. G. Kolda and B. W. Bader, "Tensor Decompositions and Applications", *SIAM Review*, Vol. 51, No. 3, 2009, pp. 455–500.

[13] L. R. Tucker, "Some mathematical notes on three-mode factor analysis", *Psychometrika*, Vol. 31, 1966, pp. 279–311.

[14] S. Grivet-Talocia, "Package Macromodeling via Time-Domain Vector Fitting", IEEE Microw. and Wireless Comp. Lett., Vol. 13, No. 11, pp. 472–474, Nov. 2003.

[15] G. Signorini, C. Siviero, I. S. Stievano, S. Grivet-Talocia, "Enhanced Macromodels of High-speed Low-Power Differential Drivers", Proc. of the *2015 IEEE Electrical Performance of Electronic Packaging and Systems (EPEPS)*, Santa Clara, California, USA, Oct. 25-28 2015.

[16] J. Sjoberg et al., "Nonlinear black-box modeling in system identification: A unified overview," Automatica, Vol. 31, No. 12, pp. 1691–1724, 1995.

[17] S. Haykin, *Neural Networks – Comprehensive Foundation*, Englewood Cliffs, NJ: Prentice Hall, 1999.