M[pi]log, Macromodeling via parametric identification of logic gates

$\mathrm{M}\pi\mathrm{log}$, Macromodeling Via Parametric Identification of Logic Gates

Igor S. Stievano, *Member, IEEE*, Ivan A. Maio, *Member, IEEE*, and Flavio G. Canavero, *Senior Member, IEEE*

***Abstract*—This paper addresses the development of computational models of digital integrated circuit input and output buffers via the identification of nonlinear parametric models. The obtained models run in standard circuit simulation environments, offer improved accuracy and good numerical efficiency, and do not disclose information on the structure of the modeled devices. The paper reviews the basics of the parametric identification approach and illustrates its most recent extensions to handle temperature and supply voltage variations as well as power supply ports and tristate devices.**

***Index Terms*—Circuit modeling, digital integrated circuits, electromagnetic compatibility, I/O ports, macromodeling, signal integrity, system identification.**

I. Introduction

PRESENT and future information technology devices must satisfy higher and higher design requirements imposed by performance and technology constraints. The consequence for the designers is to perform a large number of assessments of the signal quality on the most critical paths. This, in turn, implies an increase of the physical simulations at the device and system levels, in order to be able to detect and circumvent those sensitive effects like crosstalk and radiation that may seriously compromise the achievement of the design objectives. In these simulations, the numerical models representing the communication buffers of digital integrated circuits (IC) play a key role, since they are shaping the signals on the system interconnections. The most common way to specify IC buffers is to generate behavioral models performing like the real system, without disclosing any proprietary information. Historically, the input/output buffer information specification (IBIS) standard [1] was the first attempt to provide a device description. This specification defines data sets that characterize the device behavior under specific load conditions and allows the construction of simplified models. The models are based on a prescribed form of the equivalent circuit and their construction is an easy task already implemented in several commercial tools. Unfortunately, the accuracy at which these models mimic the behavior of the actual digital devices proves to be insufficient for most recent technologies characterized by even higher data communication speed and more sophisticated devices. In fact, models based on simplified equivalent circuits may be affected by significant sensitivity to the attached load and may be unable to reproduce all the relevant device dynamics. This is because the circuit structure defining the model decides *a-priori* the physical effects to be considered, leaving no possibilities to reproduce other effects inherent to a specific device.

An alternative approach for the construction of input/output (I/O) behavioral models is the use of nonlinear parametric models and identification methods. Parametric identification is a powerful and rigorous resource for the behavioral modeling of generic dynamic nonlinear systems. The Authors started to pursue this approach several years ago [2], and recently proved its effectiveness for I/O buffers of complementary metal oxide semiconductor (CMOS) technology [3], [4]. The macromodeling technique proposed by the authors and named $\mathrm{M}\pi\mathrm{log}$ (macromodeling via parametric identification of logic gates) provides improved accuracy, while the complexity and computational effort remain comparable with those required by conventional equivalent-circuit-based modeling approaches. The models generated with the $\mathrm{M}\pi\mathrm{log}$ approach are rigorous mathematical approximations of the whole dynamic behavior of the nonlinear devices, and are therefore more promising than the previous ones, without conveying any structural information on the devices. In addition, they have reduced load sensitivity and can be easily estimated from both reference transistor-level models and measured transient responses.

In this research, the $\mathrm{M}\pi\mathrm{log}$ methodology has been extended to take into account additional effects as the device temperature, the power supply voltage and the supply current drawn by buffers. The paper is organized as follows. Section II and III review the basics of nonlinear parametric identification and of parametric macromodels for I/O ports, respectively. Section IV illustrates the key elements of the extensions developed in this research and provides full references to the papers documenting the technical details of the extensions. Finally, Section V summarizes the results obtained.

II. Modeling Method

The modeling of ICs for signal integrity and electromagnetic compatibility simulations amounts to finding suitable port relationships (which we refer to as "constitutive"), for a known logic activity of the ICs. This key idea is best illustrated by some examples. The typical structure of a digital IC is shown in Fig. 1, where the parts relevant to this discussion, i.e., input buffers (in_1, in_2), internal logic circuits and output buffers (out_1) are highlighted. Input buffers are two-port elements bridging the world outside the IC to the internal logic. Since input buffers

Manuscript received November 1, 2003; revised January 6, 2004. This work was supported in part by a Faculty Grant from IBM Corporation, the Italian Ministry of University (MIUR) under a Program for the Development of Research of National Interest PRIN Grant 2002093437, and by CERCOM, Center for Multimedia Radio Communications of the Electronics Department, Politecnico di Torino.

The authors are with the Dipartimento di Elettronica, Politecnico di Torino, Torino I-10129, Italy (e-mail: igor.stievano@polito.it).

Digital Object Identifier 10.1109/TADVP.2004.825475

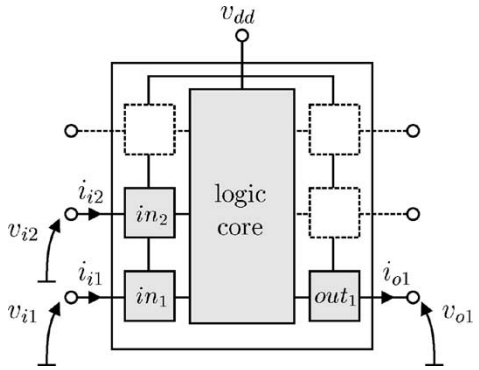

Fig. 1. Block diagram of a generic digital IC.

are unidirectional, the IC logic activity hardly influences the voltage and current of the input port. Thus the electric behavior of a generic input port (e.g., in_1) can be described by the constitutive relation of a two-terminal element

$$F_i(i_{i1}, v_{i1}) = 0 \tag{1}$$

where i_{i1} and v_{i1} are the port voltage and current variables. As a further refinement, the coupling between neighboring input buffers can be taken into account. In this case, the coupled input ports can be described by the constitutive relation of a multiport element

$$\begin{cases} F_{i1}(i_{i1}, v_{i1}, i_{i2}, v_{i2}, \ldots) = 0 \\ F_{i2}(i_{i1}, v_{i1}, \ldots) = 0 \\ \ldots \end{cases} \tag{2}$$

where the nth equation is the constitutive relation of the nth port, that relates port variables i_{in} and v_{in} to the variables of the other input ports. Similarly, output buffers are two-port elements whose input port is connected to IC internal logic and the output coincides with the IC output pins. Again, the electrical behavior of these elements is described by constitutive relations, and every further extension of the modeling (e.g., the effects of the supply voltage fluctuations) amounts to extending the set of variables constrained by constitutive relations.

In this study, buffer constitutive relations are sought as dynamic nonlinear parametric equations, in order to exploit a well-established theory successfully employed in many areas of interest, like those concerned with the identification of mechanical systems, economic trends, etc. The advantages of parametric models are their rigorous mathematical foundations, their identificability from external observations as well as the good performances for the problem at hand and the ability to hide the internal structure of the modeled devices. Finally, the identified models can be easily converted into SPICE-like subcircuits that can be used in any circuit simulation environment for the assessment of signal integrity and electromagnetic compatibility effects in fast digital circuits.

A. Parametric Models

A parametric equation is any relation between a set of variables, that depends on a set of parameters. As the parameters

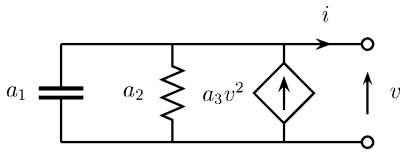

Fig. 2. Equivalent circuit implementation of model (3).

are varied, the relation can model a whole class of systems. For example, the following simple input/output parametric equation defines a parametric model:

$$i = -\frac{a_1 dv}{dt} - a_2 v + a_3 v^2 \tag{3}$$

where a_k, $k = 1,2,3$ are the model parameters, and v and i are the input and output variables of the model, respectively.

The use of parametric equations to models physical systems is conceptually simple, and amounts to applying the following procedure.

1) *Model selection*, i.e., the search for a functional form of the model equation (referred to as model representation).
2) *Parameter estimation*, i.e., the computation of the model parameter values so that the model responses mimic well those of the device under modeling.
3) *Model implementation*, i.e., the translation of the model in a circuit simulation environment by representing its equations with equivalent circuits [e.g., see Fig. 2 as an implementation of (3)].

Step three of the above modeling procedure is straightforward and is thoroughly discussed in [3]. The rest of this section deals with the model selection (step one) and the parameter estimation (step two).

B. Model Selection

The selection of the model representation is the most critical step of the modeling process, since a model representation far from the functional form of the real system can hardly reproduce its behavior. The model representation suitable for a buffer is searched for within the class of discrete-time parametric models. This is mainly due to the large availability of methods for the estimation of the models of this class [5], [6]. Besides, this is the natural choice when the raw data, i.e., the external responses of the system, are known as sampled waveforms. We remark that an additional back conversion to continuous time is needed for the implementation of the model as a subcircuit, yet this is easily carried out via standard methods in step three of the modeling procedure [3].

For the sake of simplicity, in the following general description we concentrate on scalar systems with a single output y and a single input u. In this case, a very general equation defining most input/output model representations is [6]

$$y(k) = g\left(\mathbf{x}(k), \boldsymbol{\Theta}\right) \tag{4}$$

where vector $\boldsymbol{\Theta}$ collects the model parameters and the scalar function g maps the present and past samples of the input and the past samples of the output into the present sample of the

output. The mapped input/output samples are collected in the regressor vector

$$\mathbf{x}(k)=[u(k),u(k-1),\ldots,u(k-r),y(k-1),\ldots,y(k-r)]^T \tag{5}$$

where index r is referred to as the dynamic order of the model. In Sections III and IV, parametric models defined by (4) and (5) are simply indicated in a more compact form as $g(\boldsymbol{\Theta};u)$.

As outlined in [6], (4) provides a unified framework to handle models from both system identification area and from other areas like neural networks, wavelets and fuzzy systems. A general way to define nonlinear mappings to be used in (4) is to exploit sums of nonlinear functions of regressors as

$$g(\mathbf{x},\boldsymbol{\Theta})=\sum_{n=1}^{m}\alpha_n\phi_n\left(\eta(\mathbf{x})\right) \tag{6}$$

where ϕ_n is the nth basis function, obtained from a single mother generating function $\phi(\eta)$ by changing its dilation and its position (nonlinear parameters), α_n is a linear coefficient (linear parameter) and m is the total number of components (model size). Many different basis functions can be used in (6), giving rise to model representations with significantly different properties. Possible choices of basis functions and their properties can be found in [6]–[8].

Parametric macromodels based on Gaussian radial basis function (RBF) expansions have been successfully applied to the macromodeling of the ports of digital ICs [3], [4]. For these models, the mother generating function is $\phi(\eta)=\exp(-\eta^2/2)$ and the nth basis function ϕ_n is defined by the argument

$$\eta(\mathbf{x})=\frac{|\mathbf{x}(k)-\mathbf{v}_n|}{b_n} \tag{7}$$

where $|\cdot|$ denotes the Euclidean norm and $\mathbf{v}_n$ and b_n are the nonlinear parameters defining the position and dilation, respectively, of the nth basis function. Gaussian RBF models offer remarkable advantages. Mainly, they are robust and have a regular and smooth behavior outside the fitting domain and the estimation of model parameters relies on simple and efficient algorithms.

For the problem at hand, Sigmoidal basis functions (SBF) are useful alternative to RBF. They are defined by a mother function $\phi(\eta)=\tanh(\eta)$ and by

$$\eta(\mathbf{x})=\mathbf{v}_n^T\mathbf{x}(k)+b_n \tag{8}$$

where $\mathbf{v}_n$ and b_n are the nonlinear parameters of the basis function ϕ_n.

In contrast to RBF, that have a spherical symmetry, SBF have a planar (ridge type) symmetry and unbounded support. These properties are more suitable for fitting the actual constitutive relations of IC ports and usually lead to simpler (more efficient) macromodels. This holds especially for nearly linear constitutive relations, whose RBF fittings require a large number of basis functions. Furthermore, the algorithms themselves for the estimation of SBF models, even if requires more complex and fully nonlinear procedures, allows for more accurate estimates.

C. Parameter Estimation

The unknown parameters defining a parametric model are estimated from transient responses of the device under modeling (referred to as the estimation data set). They are computed by minimizing a suitable error function of the estimation data set and of the responses of the model.

In the framework outlined in previous sections, the starting point is a completely defined model representation, i.e., a regressor vector and a set of basis functions. The structure of the regressor vector, or equivalently the dynamic order of the model is rather a property of the device under modeling, and can be determined *a priori* from the device responses [9], or simply postulated and verified a posteriori. Then the size m of the model and its linear and nonlinear parameters must be estimated. An estimation data set is obtained by measuring a suitable response of the device under modeling and is indicated as

$$\mathbf{Z}_e^N=\{y(k),\mathbf{x}(k)\},\quad k=1,\ldots,N \tag{9}$$

where N is the size of the data set. For a given m, the parameter vector is estimated as a solution of the following nonlinear approximation problem, usually cast as a least square problem

$$\mathbf{Y}=\begin{bmatrix}y(1)\\ \vdots\\ y(N)\end{bmatrix}\approx g(\mathbf{X},\boldsymbol{\Theta})=\begin{bmatrix}g\left(\mathbf{x}(1),\boldsymbol{\Theta}\right)\\ \vdots\\ g\left(\mathbf{x}(N),\boldsymbol{\Theta}\right)\end{bmatrix}. \tag{10}$$

Specific algorithms are available to solve this problem, that depend on the specific choice of the family of basis functions [6].

Of course, the size N of the estimation data set and the shape of the input waveforms used to stimulate the device under modeling influence the quality of the resulting model. In order to give rise to good models, the estimation waveforms must contain as much information on the device behavior as possible. This is obtained by using as input waveforms noisy multilevel signals spanning the range allowed for the input signals. The design of input estimation waveforms is a matter of repeated estimation experiments, where the shape of the waveforms is tuned to yield models with small estimation errors. Specific guidelines to generate estimation signals for the modeling of IC buffers are provided in [3].

In order to decide the most suitable size m of the model, a comparison of models with different m values is performed, with respect to their estimation errors. As a rule of thumb, good m values are the smallest ones leading to a good reproduction of the estimation data set. Suitable statistical indexes help the selection of m [5].

The estimation of Gaussian RBF models relies on simple and efficient algorithms [10], [11] in which the nonlinear least squares problem (10) is cast as an equivalent linear problem. This can be done since RBF models are weakly sensitive to position and dilation parameters. This allows to *a priori* select the position and dilation parameters, so that the nonlinear problem (10) is reduced to the estimation of the linear parameters only. Instead, the estimation of SBF models requires a fully nonlinear estimation algorithm to solve problem (10). Within the many possible estimation algorithms, we found good results with the

Levenberg–Marquardt method [12], in conjunction with the procedure for the selection of the initial guess of parameters suggested in [13].

Finally, the ability of the estimated model to mimic the modeled device for input signals different from those contained in the estimation data set must be ascertained. This is obtained by computing the model error for input signals outside of the estimation data set. This process is referred to as model validation.

III. Review of Basic Macromodels

The parametric identification approach has been initially applied to the macromodeling of input and output ports of CMOS ICs. This Section shortly reviews the model representations used for these basic modeling problems. A complete discussion of input and output port modeling as well as several modeling examples involving commercial devices and starting from both transistor-level simulations and actual measurements are reported in [3], [4].

A. Receivers

The modeling of an uncoupled input port is conceptually simple, since it amounts to looking for an approximation of the constitutive relation between the port voltage and current variables. As the behavior of the port in the range of the power supply voltage is mainly linear and dynamic, and since it becomes strongly nonlinear for voltages outside the power supply voltage range, we use a model representation defined by

$$i(k) = i_l(\boldsymbol{\Theta}_l; v) + i_{nl}(\boldsymbol{\Theta}_{nl}; v) \tag{11}$$

where i and v denote the port current and voltage variables, respectively, defined by associate reference directions. Here we assume that the current i splits into the sum of two contributions, of which i_l is a linear parametric model defined by an auto regressive with extra input model (ARX) scheme [5], and accounting for the linear behavior of the port; i_{nl} is a nonlinear (e.g., SBF) piecewise model accounting for the port behavior in the voltage range where the effects of protection circuits take place [14], [15].

It is ought to remark that in spite of its simplicity, the modeling of input ports may be a challenging problem, because the linear region may have high dynamic order and the onset of the nonlinear regime my be abrupt and may introduce additional and slow time constants. In spite of the above critical points, model representation (11) and a careful tuning of the modeling process allow to obtain good models for most cases of practical interest.

B. Drivers

Fig. 3 shows the typical structure of a digital driver made of cascaded stages. The electrical variables relevant for the model, i.e., the voltage and current of the output and of the power supply ports (denoted by v, i, v_{dd}, and i_{dd}) are defined in Fig. 3, as well.

The key problem for the development of a driver model is that the internal (logic) signal feeding the buffer is not a measurable quantity. In spite of this, the obtained model must allow for variations of the logic state. For constant values of the power supply port voltage, a parametric macromodel of the output port has been presented and thoroughly discussed in [3]. It approximates the output port constitutive relation with a two-piece model

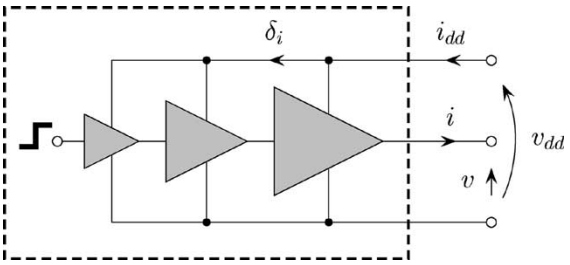

Fig. 3. Typical structure of a driver circuit and its relevant (output and power supply) port electrical variables.

$$i(k) = w_1(k) f_1(\boldsymbol{\Theta}_1; v) + w_2(k) f_2(\boldsymbol{\Theta}_2; v) \tag{12}$$

where f_1 and f_2 are nonlinear parametric models accounting for the port behavior in a fixed HIGH or LOW logic state, respectively; w_1 and w_2 are weighting coefficients for state switchings.

Piecewise models similar to (12) occur also in conventional modeling approaches (e.g., IBIS). However, they stem from heuristic assumptions (e.g., for fixed logic state, the dynamic behavior of the devices is accounted for empirically) and suffer from the limitations mentioned in Section I. In contrast, (12) arises rigorously from the properties of parametric models, and their submodels f_1 and f_2 take into account both the nonlinear and dynamic behavior of the device for fixed logic state.

C. Performances

In order to assess the performances of the basic macromodels and highlight the differences among the possible choices of basis functions, the same example high-speed IBM CMOS transceiver ($V_{dd} = 1.8$ V) of [3] is considered. For this assessment, the output port of the device and the basic driver macromodel (12) are considered, as well. The estimation data set needed for the development of the macromodel are obtained from the transient responses of a detailed transistor-level model of the device. All the simulations needed to generate the estimation data set and to validate the estimated macromodels are carried out by means of PowerSPICE. For this device, both the RBF and the SBF parametric models are estimated (the dynamic order of submodels is $r = 1$).

Table I shows the mean squared errors (MSE) computed during the estimation procedure for submodel f_1, i.e., the dynamic $i - v$ port relation when the driver is forced in the fixed HIGH output state, and different values of the number of basis functions m_1. MSE values on the order of 1E-6 lead to macromodels reproducing the behavior of the device very well and make the predicted responses almost indistinguishable from the reference ones. It is worth noting that the same good accuracy obtained with RBF models for $m_1 = 7$ is achieved with SBF models for $m_1 = 1$, only.

Also, Table II compares the complexity and efficiency of the driver macromodels. The table lists the number of basis functions m_1 and m_2, for the submodels f_1 and f_2, respectively, and the CPU time required to compute a simple transient simulation test using the reference transistor-level models and two SPICE-like implementations of the macromodels. The

TABLE I
MEAN SQUARED ERROR (MSE) BETWEEN THE OUTPUT IDENTIFICATION SEQUENCE AND THE MODEL RESPONSE TO THE IDENTIFICATION SEQUENCES (APPROXIMATION ERROR). BOTH THE RBF AND THE SBF TYPE MODELS ARE CONSIDERED. THE TOTAL NUMBER OF BASIS FUNCTIONS m_1 INCLUDED IN THE MODELS IS IN THE RANGE [1], [10]

m_1	MSE(RBF)	MSE(SBF)
1	4.12e-1	**2.07e-6**
2	7.22e-2	8.30e-7
3	3.25e-4	8.28e-7
4	3.14e-4	8.11e-7
5	1.40e-4	8.78e-7
6	2.40e-5	8.48e-7
7	**1.63e-6**	8.87e-7
8	1.59e-6	8.82e-7
9	1.44e-6	8.88e-7
10	1.34e-6	8.65e-7

test setup consists of the driver being modeled connected to a 100-Ω load resistor and producing a logic high pulse (bit pattern "010"). The speedup factor introduced by the macromodels can be clearly appreciated.

As a realistic validation setup, the example driver applying a logic high pulse on an open-ended ideal transmission line ($Z_0 = 100\ \Omega$, $T_d = 100$ ns) is considered. Fig. 4 shows the near-end transient voltage waveform computed for PowerSPICE by using both the reference transistor-level model and the SBF type macromodel. Curves obtained using the RBF type of models are not reported since they are indistinguishable from that of the SBF type.

The accuracy of the proposed basic macromodel has been quantified by computing the timing error, that is expressed as the maximum delay between the reference and the macromodel responses measured for a suitable voltage crossing (e.g., 50% of the voltage swing). For a large set of experiments involving different devices, we found timing errors on the order of $5 \div 20$ ps (with an average of 10 ps). As an example, for the curves of Fig. 4 the timing error is 7 ps (0.35% of the bit time).

Finally, an additional index for the assessment of our methodology is the efficiency, in terms of the CPU time required by the estimation of the basic macromodels. For instance, a typical identification process for the example device takes some ten seconds on a Pentium IV at 1 GHz.

IV. EXTENSIONS

Recently, our approach has been extended to handle additional effects and different device types. The new applications involve the effects of device parameters and extra variables, as well as the modeling of power supply ports and tristate drivers.

A. Extra Parameters: The Temperature

This subsection deals with the extension of macromodels to include the effects of static or slowly varying device parameters, like the device temperature and manufacturing process parameters. If we further assume to deal with device parameters having a weak and continuous influence on the device behavior, the extension is straightforward and amounts to estimating the parameters of the models as functions of the device parameters

TABLE II
NUMBER OF BASIS FUNCTIONS OF THE POSSIBLE MACROMODELS OF THE EXAMPLE DRIVER AND CPU TIME COMPARISON FOR A SIMPLE TRANSIENT SIMULATION TEST (SEE TEXT)

Macromodel	m_1	m_2	CPU time (PowerSPICE)	CPU time (Pspice)
reference	-	-	27 s	-
RBF	7	8	0.42 s	1.77 s
SBF	1	2	0.22 s	0.50 s

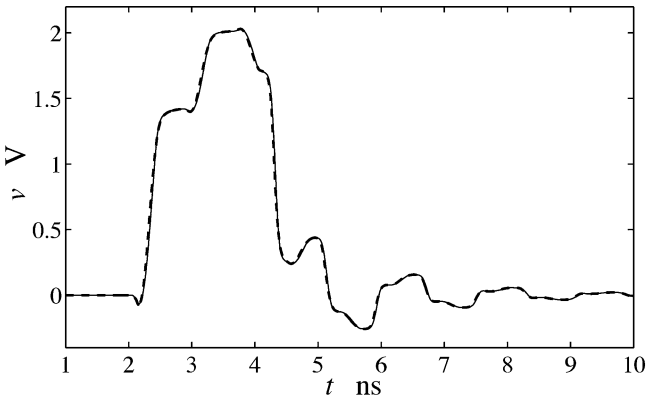

Fig. 4. Near-end voltage waveform on an ideal transmission line driven by the example driver (see text). Solid line: reference; dashed line: SBF macromodel.

to be taken into account. As an example, the effects of a set of parameters $\mathbf{P}$ could be included in model (12) by estimating the model parameters as a function of $\mathbf{P}$, i.e., $\boldsymbol{\Theta}_n = \boldsymbol{\Theta}_n(\mathbf{P})$ and $w_n = w_n(\mathbf{P})$, $n = 1, 2$. However, when the elements of $\mathbf{P}$ have a weak influence, the estimation of the model parameters as a function of $\mathbf{P}$ is not efficient. Besides, even if the identifiability of the model assumes that (10) has a unique solution, in practice the numerical solution may be a local minima and spurious discontinuities of computed model parameters may arise.

In order to avoid these problems, function $\boldsymbol{\Theta}_n = \boldsymbol{\Theta}_n(\mathbf{P})$ is approximated by $\overline{\boldsymbol{\Theta}}_n = \{\alpha_{nj}(\mathbf{P}), v_{nj}(\mathbf{P}_o), b_{nj}(\mathbf{P}_o)\}$, where only the linear parameters α_{nj} are estimated as functions of $\mathbf{P}$ whereas the position and dilation parameters are set to the values estimated for some $\mathbf{P}_o$ located in the center of the domain of variation of $\mathbf{P}$. Usually the variation of linear parameters is sufficient to take into account weak variations of the model behavior and is easy to compute, because, for given position and dilation values, the linear parameters can be obtained as solution of a linear least square problem [10]. Also for continuous variations of the model behavior, $\alpha_{nj}(\mathbf{P})$ are continuous functions.

The parameter *device temperature* (symbol T) and RBF models are used to exemplify this approach. We estimate the complete set of parameters from the transient identification waveforms recorded at a nominal temperature value, which is normally the middle point of the explored temperature range. We use the position and dilation parameter values at that temperature, for the entire temperature span, but we re-estimate the linear parameters for other temperatures, as solution of a linear least square problem. Besides, we find that the dependence of the linear parameters α_{nj} and w_n on T can

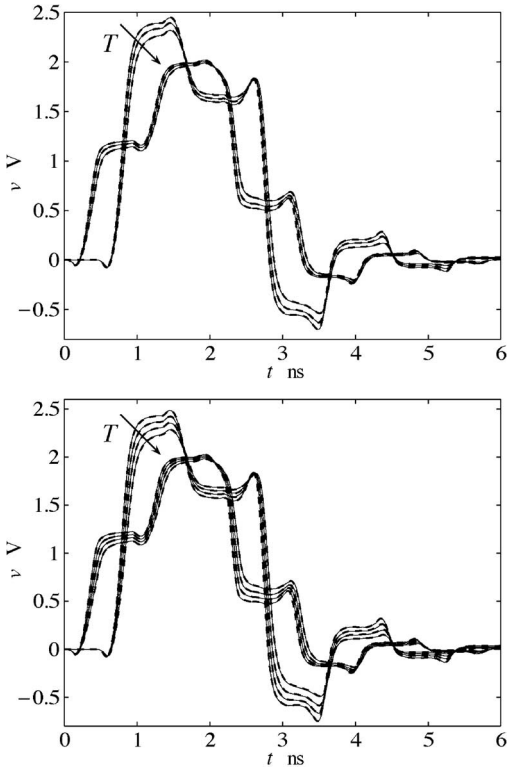

Fig. 5. Validation of a temperature-dependent driver macromodel: near- and far-end voltage waveforms on an ideal transmission line driven by the output port of the example device and loaded by a capacitor (details in Section IV-A). Solid curves: reference responses; dashed curves: model responses. Top panel: curves computed at $T = 10$, 40, 80 °C (minimum, nominal and maximum temperature of the model, respectively); bottom panel: curves computed at $T = -10$, 20, 60, 100 °C.

be well approximated by piecewise linear functions, i.e., for any T_1, T_2, and $T \in [T_1, T_2]$

$$\boldsymbol{\alpha}_n(T) \approx \left(\frac{T-T_2}{T_1-T_2}\right)\boldsymbol{\alpha}_n|_{T_1} + \left(\frac{T-T_1}{T_2-T_1}\right)\boldsymbol{\alpha}_n|_{T_2} \quad (13)$$

$$w_n(k;T) \approx \left(\frac{T-T_2}{T_1-T_2}\right) w_n(k)|_{T_1} + \left(\frac{T-T_1}{T_2-T_1}\right) w_n(k)|_{T_2} \quad (14)$$

where $\boldsymbol{\alpha}_n$ is the vector of all linear parameters of submodel n, $\boldsymbol{\alpha}_n|_{T_1}$ and $\boldsymbol{\alpha}_n|_{T_2}$ are the vector of linear parameters computed at $T = T_1$ and $T = T_2$, respectively, and $w_n(k)|_{T_1}$ and $w_n(k)|_{T_2}$ are the weighting coefficients of (12) computed at the same extreme temperatures.

The experience based on repeated approximations brings us to claim that the accuracy required by common simulation problems can be obtained from two-piece linear approximations defined by three temperature values, corresponding to the minimum, nominal and maximum values of the IC operating temperature.

Fig. 5 shows an example of validation for a temperature-dependent driver model. The device under modeling is the output port of the example device introduced in Section III-C and the model is built as outlined above. The test circuit is composed of an ideal transmission line (characteristic impedance $Z_o = 50\,\Omega$ and delay $T_d = 40$ ps) driven by the modeled device and loaded by a 1 pF capacitor. The curves of Fig. 5 are the near- and far-end voltage waveforms predicted by using the reference and the estimated model of the driver when they apply a logic high pulse (bit pattern "010") to the transmission line. It is worth to notice that the agreement of the reference and model responses remains good also for $T = -10\,°\mathrm{C}$ and $T = 100\,°\mathrm{C}$, that are outside the range [10 °C, 80 °C], used to define the model. Finally, it should be remarked that, since the structure of the proposed temperature-dependent model is similar to the structure of the basic model, the cost of generation and the efficiency of temperature-dependent models are close to those of basic models.

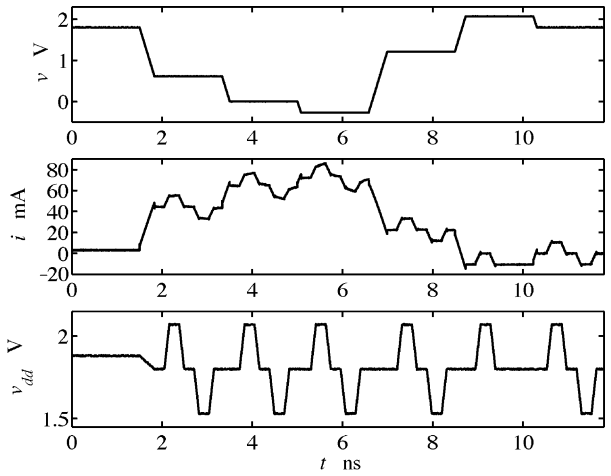

Fig. 6. Estimation signals for the submodel f_1 of a driver model including the effects of the power supply voltage v_{dd}.

B. Extra Variables: Power Supply Voltage

This subsection deals with the extension of macromodels to include the effects of additional variables, like the voltage applied to the power supply port. In principle, the effects of any rapidly varying physical quantity can be included in a parametric model by adding such quantity to the input variables of the model. Once the proper supplemental input variables are defined, the practical problem turns into the estimation of the parameters of a multi-input model. This requires a careful design of input estimation waveforms to excite every significant dynamic behavior that can be caused by the selected input variables.

As an example, we will consider the inclusion of the power supply voltage in the driver model. This inclusion is important, because it allows to take into account of the power supply port fluctuations on the switching of logic devices (of course, this requires a joint model of the power supply port itself, enabling the simulation of the simultaneous switching noise (SSN) as described in the next subsection). The inclusion of the power supply voltage v_{dd} in driver models is obtained by adding an input variable to the submodels f_n of (12), as

$$f_n = f_n(\boldsymbol{\Theta}_n; v, v_{dd}) \quad n = 1, 2. \quad (15)$$

The weighting coefficients describing state switching $w_n(k)$, instead, are considered constant with respect to v_{dd}, owing to the properties of model representation (12). For the estimation of the parameters of (15), we use as input estimation waveforms multilevel noisy signals for both $v(k)$ and $v_{dd}(k)$ and apply the

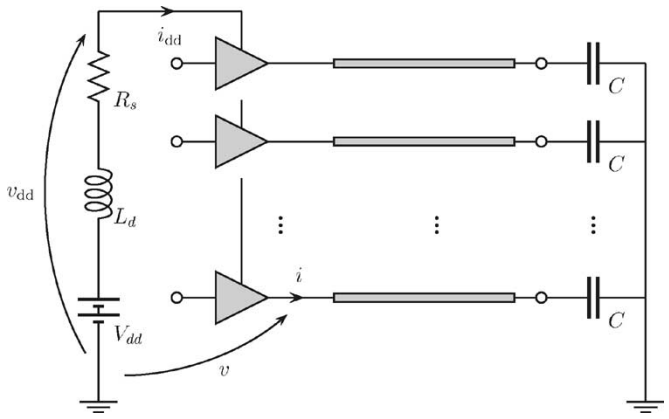

Fig. 7. Test structure for the validation of macromodels (15) and (16).

same estimation algorithms used for the single-input case. As an example, Fig. 6 shows the estimation waveforms used for the estimation of submodel f_1 in the case of the example device introduced in Section III-C. The macromodel obtained for this device turns out to have a dynamic order $r = 2$, and submodels f_1 and f_2 composed of nine and six basis functions, respectively. A numerical example of the performance of this macromodel is given in the next subsection, whereas more details of this modeling problem are in [16].

C. Modeling of Power Supply Ports

The modeling of the power supply port of drivers is of paramount importance for the simulation of switching noise effects. In order to model these ports, we need to express the driver supply current i_{dd} as a function of the supply port voltage v_{dd} and of the output port voltage v (see Fig. 3). The model representation exploited for the power supply port of drivers is

$$i_{dd}(k) = F_{dd}(k) + \delta_i(k) \tag{16}$$

where $\delta_i(k)$ takes into account the supply current drawn by the driver stages that precede the last one, and F_{dd} represents the supply current drawn by the last inverter stage. We assume that the current of the preceding stages is independent of v and v_{dd}, and evolves with the driver logic state only. On the contrary, F_{dd} is modeled by a two-piece parametric relation

$$F_{dd}(k) = w_{d1}(k) f_{d1}(\boldsymbol{\Theta}_{d1}; v, v_{dd}) + w_{d2}(k) f_{d2}(\boldsymbol{\Theta}_{d2}; v, v_{dd}) \tag{17}$$

where f_{d1} and f_{d2} are the parametric submodels (e.g., SBF) of the current of the last driver stage when it operates in the LOW and HIGH logic states, respectively, and w_{d1} and w_{d2} are the usual weighting coefficients describing state switchings.

The obtained macromodel (17) for the IBM example device introduced in Section III-C turns out to have dynamic order $r = 2$, and submodels f_{d1} and f_{d2} composed of nine and seven basis functions, respectively. More details on the estimation of model parameters are in [16].

In order to show the capabilities of the models described in this and in the previous subsection, we apply them to the test circuit of Fig. 7. This test circuit is composed of eleven ideal transmission lines loaded by capacitors and driven by eleven identical replicas of the example driver device introduced in Section III-C. The power supply port of each driver is connected to a common power supply network schematically represented by the battery V_{dd} and a lumped load (R_s, L_d). The values of the circuit parameters are: transmission line characteristic impedance $Z_0 = 50\ \Omega$, time delay $T_d = 0.5$ ns; $C = 1$ pF, $R_s = 1\ \text{m}\Omega$, $L_d = 0.1$ nH, and $V_{dd} = 1.8$ V. The simulation is conducted for all eleven drivers applying simultaneously the same logic high pulse to the transmission lines. Validation results are shown in Fig. 8 for one driver port and for the power supply port: voltage and current waveforms predicted by models (15) and (16) are compared against references generated by a detailed transistor-level description of the circuit. The simulations carried out for this test circuit confirm that our approach can be successfully applied to accurately model complex structures. Speed-up factors on the order of 20 ÷ 100 with respect to the reference simulations are obtained, as well.

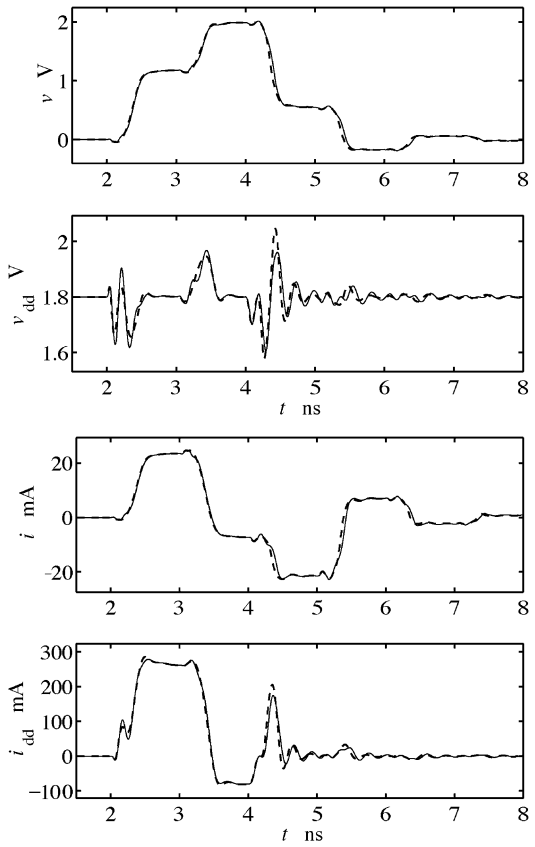

Fig. 8. Validation of macromodels (15) and (16): output port voltage v and current i waveforms and power supply voltage v_{dd} and current i_{dd} waveforms for the structure of Figs. 7 and 11 simultaneously switching drivers. Reference responses: solid thin lines; model responses: dashed thick lines.

A real application of driver macromodels including the power supply port characterization for the SSN prediction in a complex system is discussed in [17]. In this paper, the accuracy of predictions is verified by comparing simulation results and measurements.

D. Modeling of Tristate Devices

An additional extension of the Mπlog methodology refers to the modeling of tristate drivers. These devices are drivers whose output ports can operate also in a HIGH IMPEDENCE (HZ)

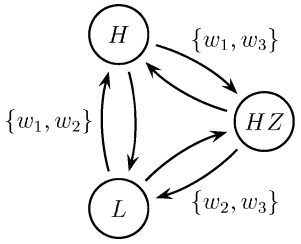

Fig. 9. State transition diagram of a tristate driver with the relevant weighting coefficients (see text for details).

state, thereby approximating open circuits. The model representation proposed for the output port of these devices is the following three-piece parametric relation, that arises as a generalization of (12)

$$i(k) = w_1(k) f_1(\boldsymbol{\Theta}_1; v) + w_2(k) f_2(\boldsymbol{\Theta}_2; v) + w_3(k) f_3(\boldsymbol{\Theta}_3; v). \tag{18}$$

In the above equation, f_1, f_2, and f_3 are parametric submodels reproducing the driver output port current when the driver operates in the LOW, HIGH, and HZ states, respectively, and w_n, $n = 1, 2, 3$ are weighting coefficients describing state switchings. Submodels f_1 and f_2 represent the normal operation of the driver, and are sought in the form of RBF relations as in (12). Submodel f_3, instead, represents the HZ state of the device, and is sought in the form of (11), since the device appears like a receiver, in this case.

The estimation of parametric submodels f_n is carried out as outlined in Section III for drivers and receivers. Once the parametric submodels are known, the weighting coefficients for state switchings are obtained by the same procedure used for driver models. The state diagram of Fig. 9 shows that there are six possible transitions between the device states. Also, for every transition, only a pair of submodels and weighting coefficients are involved (the coefficient of the excluded submodel being set to zero). The pair of weighting sequences of any state transition can be obtained from the port transient responses recorded for two suitable reference loads. Once all transient responses are recorded, a linear inversion of (18) provides the three coefficients w_1, w_2, and w_3. The complete evolution of the weighting coefficients for a given sequence of state transitions is then obtained by concatenating the elementary sequences computed for the six possible transitions. Of course, similarly to the case of normal drivers, model (18) holds only for state transitions starting from steady operation in one of the allowed states.

In the following, we present a tristate model for the output port of the example device introduced in Section III-C. Submodels f_1 and f_2 are already defined in Section III, and submodel f_3 turned out to have dynamic order $r = 3$. In order to show the operation of the proposed model, we use it in the simulation of two test circuits. The first test circuit is composed of the modeled driver connected to an ideal resistor. Fig. 10 shows a set of output voltage waveforms predicted by the proposed and the reference models for a given sequence of state transitions and for two resistance values (50 and 250 Ω). The state transitions of this example are obtained by applying the bit patterns "001111001100000" and "000110011001000" to the driver input and to the driver enable pin, respectively (the modeled

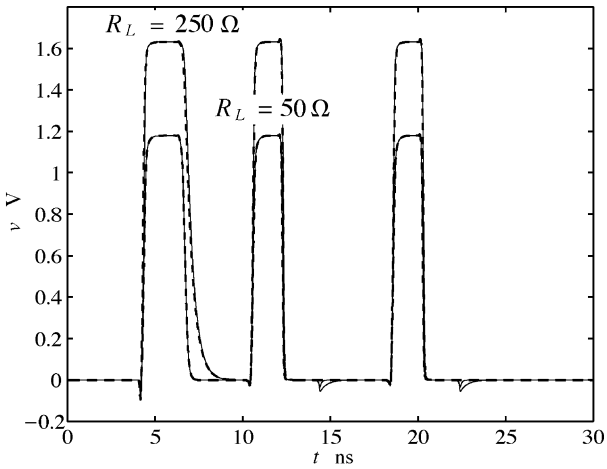

Fig. 10. Validation of macromodel (18): output port voltage waveforms when the modeled device performs a sequence of state transitions (see text) and is connected to an ideal resistor with resistance R_L. Solid lines: reference; dashed lines: macromodel.

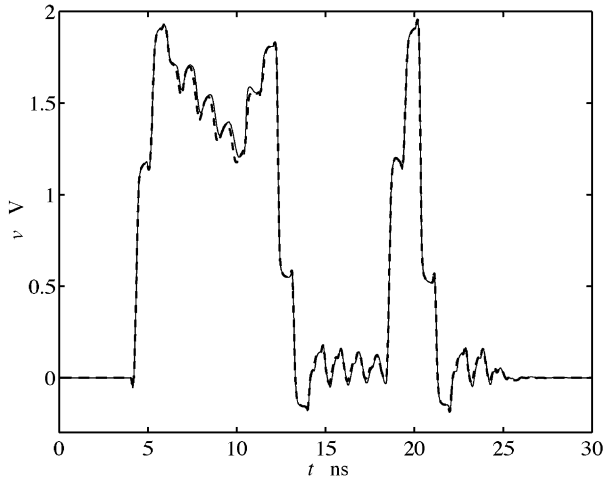

Fig. 11. Validation of macromodel (18): output port voltage waveforms when the modeled device perform a sequence of state transitions (see text) and is connected to an ideal transmission line loaded by a 1-pF capacitor. Solid lines: reference; dashed lines: macromodel.

driver is of noninverting type). The model accuracy is readily appreciated by the superposition of the reference and model curves. The second test circuit is composed of the modeled device driving an ideal transmission line (characteristic impedance $Z_0 = 50\ \Omega$, delay $T_d = 50$ ps) that is loaded by a 1-pF capacitor. Fig. 11 shows the comparison between the responses predicted by using the reference and the proposed model when the devices performs the same sequence of state transitions as above. The good agreement of the responses predicted by the two models can be clearly appreciated.

V. Conclusion

The research carried out shows that the Mπlog approach can be successfully extended to take into account of the temperature and power supply voltage variations. Further extensions, e.g., the inclusion of manufacturing process parameters, are straightforward application of the techniques proposed in this paper. An additional extension presented in this paper is the derivation of

models for driver power supply ports and for tristate devices. A related activity, conducted internally by IBM, experienced the application of our $M\pi log$ methodology to the modeling of dynamic output controlled (DOC) drivers, with good results [18].

The most remarkable features of the proposed $M\pi log$ extensions is that they preserve the good features of basic receiver and driver models of [3], [4], i.e., the simplicity of their representations as well as the good accuracy and efficiency levels. In fact, we found that the modeling efforts to build a power supply port model or a temperature-dependent output port model remain comparable to those required by a plain driver model. Besides, the accuracy of the extended models matches that of the basic models and the speedup factors with respect to the transistor-level models are comparable with those shown in Table II for the basic models.

In summary, this means that $M\pi log$ approach provides a solid framework that can be exploited to address all the behavioral modeling problems usually treated by conventional modeling approaches based on simplified equivalent circuits, with an improved accuracy. Besides, the parametric models produced by $M\pi log$ are expressed by equations or equivalent circuits that can be used in standard circuit environment without any further processing or data interpretation. As an additional benefit, they hide to final users any structural information on the device they describe.

ACKNOWLEDGMENT

The authors wish to thank Dr. G. Katopis, Enterprise System Group, Poughkeepsie, NY, for his support, and Dr. Z. Chen, for helping with the reference models.

REFERENCES

[1] (2004, Feb.) I/O Buffer Information Specification (IBIS) Ver. 4.1. [Online]http://www.eigroup.org/ibis/ibis.htm

[2] I. S. Stievano, "Behavioral modeling of nonlinear circuit elements: Application to signal integrity and electromagnetic compatibility," Ph.D. dissertation, Politecnico di Torino, Turin, Italy, Mar. 2001.

[3] I. S. Stievano, F. G. Canavero, and I. A. Maio, "Parametric macromodels of digital I/O ports," *IEEE Trans. Adv. Packag.*, vol. 25, pp. 255–264, May 2002.

[4] I. S. Stievano, I. A. Maio, and F. G. Canavero, "Behavioral models of I/O ports from measured transient waveforms," *IEEE Trans. Instrum. Meas.*, vol. 51, pp. 1266–1270, Dec. 2002.

[5] L. Ljung, *System Identification: Theory for the User*. Englewood Cliffs, NJ: Prentice-Hall, 1987.

[6] J. Sjöberg *et al.*, "Nonlinear black-box modeling in system identification: a unified overview," *Automatica*, vol. 31, no. 12, pp. 1691–1724, 1995.

[7] I. W. Sandberg, "Approximations for nonlinear functionals," *IEEE Trans. Circuits Syst. I*, vol. 39, pp. 65–67, Jan. 1992.

[8] ——, "Approximation theorems for discrete-time systems," *IEEE Trans. Circuits Syst. I*, vol. 38, pp. 564–566, May 1991.

[9] M. Autin, M. Biey, and M. Hasler, "Order of discrete time nonlinear systems determined from input/output signals," in *Proc. IEEE Int. Symp. Circuits Syst./ISCAS-92*, San Diego, CA, May 10–13, 1992, pp. 296–299.

[10] S. Chen, C. F. N. Cowan, and P. M. Grant, "Ortogonal least squares learning algorithm for radial basis function network," *IEEE Trans. Neural Networks*, vol. 2, pp. 302–309, Mar. 1991.

[11] K. Judd and A. Mees, "On selecting models for nonlinear time series," *Physica D*, vol. 82, pp. 426–444, 1995.

[12] M. T. Hagan and M. Menhaj, "Training feedforward networks with the marquardt algorithm," *IEEE Trans. Neural Networks*, vol. 5, pp. 989–993, Nov. 1994.

[13] D. Nguyen and B. Widrow, "Improving the learning speed of 2-layer neural networks by choosing initial values of the adaptive weights," in *Proc. Int. Joint Conf. Neural Networks (IJCNN)*, San Diego, CA, Jun. 17–21, 1990, pp. 21–26.

[14] I. S. Stievano, D. Becker, F. G. Canavero, Z. Chen, G. Katopis, and I. A. Maio, "Behavioral modeling of IC ports including temperature effects," in *Proc. 11th IEEE Topical Meeting Electr. Perform. Electron. Packag. (EPEP)*, Monterey, CA, Oct. 21–23, 2002, pp. 333–336.

[15] I. S. Stievano, I. A. Maio, and F. G. Canavero, "Temperature-dependent macromodels of digital device," in *Proc. 15th IEEE Int. Zurich Symp. Electromagn. Compat.*, Zurich, Switzerland, Feb. 18–20, 2003, pp. 321–326.

[16] I. S. Stievano, F. G. Canavero, Z. Chen, G. Katopis, and I. A. Maio, "Parametric macromodels of drivers for SSN simulations," in *Proc. IEEE Int. Symp. Electromagn. Compat.*, Boston, MA, Aug. 18–22, 2003, pp. 616–621.

[17] S. Chun, A. Haridass, and C. O'Reilly, "Efficient modeling of simultaneous switching noise in a realistic computer system," in *Proc. 12th IEEE Topical Meeting Elect. Perform. Electron. Packag. (EPEP)*, Princeton, NJ, Oct. 27–29, 2003, pp. 25–28.

[18] Z. Chen, G. Katopis, and D. Becker, "Behavioral models of pre-compensation driver and dynamic termination receiver for packaging system timing and coupled noise simulations," in *Proc. 11th IEEE Topical Meeting Elect. Perform. Electron. Packag. (EPEP)*, Monterey, CA, Oct. 21–23, 2002, pp. 329–332.

Igor S. Stievano (M'98) received the Laurea and Ph.D. degrees in electronic engineering from the Politecnico di Torino, Torino, Italy, in 1996 and 2001, respectively.

Currently, he is an Assistant Professor of circuit theory with the Department of Electronics, Politecnico di Torino. His research interests are in the field of electromagnetic compatibility, where he works on the macromodeling of linear and nonlinear circuit elements with specific application to the behavioral characterization of digital integrated circuits and linear junctions for the assessment of signal integrity and electromagnetic compatibility effects.

Ivan A. Maio (M'98) received the Laurea degree and the Ph.D. degree in electronic engineering from the Politecnico di Torino, Torino, Italy, in 1985 and 1989, respectively.

Currently, he is a Professor of circuit theory with the Department of Electronics, Politecnico di Torino. His research interests are in the fields of electromagnetic compatibility and circuit theory, where he works on interconnect modeling, and linear and nonlinear circuit modeling and identification.

Flavio G. Canavero (SM'99) received the Laurea degree in electronic engineering from the Politecnico di Torino, Torino, Italy, in 1977 and the Ph.D. degree from the Georgia Institute of Technology, Atlanta, in 1986.

Currently, he is a Professor of circuit theory and electromagnetic compatibility with the Department of Electronics, Politecnico di Torino. His research interests include interconnect modeling and digital integrated circuits characterization for signal integrity, field coupling to multiwire lines, and statistical methods in EMC. He is Vice President for Organization, Politecnico di Torino.

Dr. Canavero is Editor of the IEEE TRANSACTIONS ON ELECTROMAGNETIC COMPATIBILITY, and Vice Chair of URSI Commission E. He has been the Organizer of the Workshop on Signal Propagation on Interconnects (SPI), from 2001 to 2003.