Machine learning for the performance assessment of high-speed links

Machine Learning for the Performance Assessment of High-Speed Links

Riccardo Trinchero, *Member, IEEE*, Paolo Manfredi, *Member, IEEE*, Igor S. Stievano, *Senior-Member, IEEE*, Flavio G. Canavero, *Fellow, IEEE*

***Abstract*—This paper investigates the application of support vector machine to the modeling of high-speed interconnects with largely varying and/or highly uncertain design parameters. The proposed method relies on a robust and well-established mathematical framework, yielding accurate surrogates of complex dynamical systems. An identification procedure based on the observation of a small set of system responses allows generating compact parametric relations, which can be used for design optimization and/or stochastic analysis. The feasibility and strength of the method are demonstrated based on a benchmark function and on the statistical assessment of a realistic printed circuit board interconnect, highlighting the main features and benefits of this technique over state-of-the-art solutions. Emphasis is given to the effects of the initial sample size and of input noise on the model estimation.**

***Index Terms*—High-speed interconnect, machine learning, parameterized modeling, SVM regression, uncertainty.**

I. Introduction

The availability of mathematical tools for the parametric analysis of complex dynamical system has become an important resource for the design of next generation electrical and electronic equipment. Manufacturing process, uncontrollable parameters, and uncertain device characteristics lead to possibly large variations in the circuit responses that need to be accurately predicted during the design phase. Within the EMC scenario, tolerance analysis, design exploration, and design optimization became crucial during the design workflow [1].

To this end, in the past years a number of techniques have been consolidated as viable approaches for both the parametric analysis and the statistical assessment of the behavior of generic electronic circuits. These include Monte Carlo (MC) or its improved variants, which can be undoubtedly assumed as the standard reference tools for design exploration [1], [2], parameterized macromodeling [3]– [6], polynomial chaos (PC) [7]– [13], worst-case methods [14]– [19], and generative models [20].

Even though MC is accurate, it requires a high computational cost without providing a parametric surrogate of the system responses. On the other hand, none of the other mentioned techniques provide an ultimate solution

1) for problems with a large variability of the input parameters (e.g., 50%);
2) to generate models starting from a very limited number of observations of the system responses;
3) providing a parameterized macromodel;
4) with a proven robustness to the effect of possible noise corrupting sampled data.

The above considerations are representative of a number of real life applications (e.g., data from measurements) in which only a relatively small number of realizations are available due to budget and/or time constraints, and provide a strong motivation to systematically investigate the feasibility and strengths of other approaches.

In this framework, machine learning methods are a mature alternative that have been seldom applied to this class of problems. They consist of a set of powerful tools that find successful applications in many engineering areas, mainly for classification purposes [21]- [31]. Support vector machine (SVM) is probably one of the most popular examples, with applications for both classification and regression [24]– [28]. It relies on a strong mathematical background [32]- [34] and ready-to-use algorithms embedded in almost any software for data processing.

This paper contributes with a systematic and unambiguous discussion of the main features of the SVM technique, stressing its advantages and limitations w.r.t. another approach having similar features, i.e., PC. The paper is organized as follows. The problem statement and the goals of the proposed investigation are stated in Section II. Section III briefly outlines the mathematical framework of PC. Section IV introduces the fundamentals of the SVM regression. Sections V and VI discuss the accuracy and features of the surrogate models constructed with both the PC and SVM techniques by considering a synthetic function and a high-speed link, respectively, each with uncertain parameters. Section VII concludes the paper.

II. Problem statement

Let us consider the following parametric system:

$$y = \mathcal{M}(\mathbf{x}), \tag{1}$$

where $y \in \mathbb{R}$ is the output of the system and $\mathbf{x} = \{x_1, x_2, \ldots, x_d\} \in \mathcal{X}$, with $\mathcal{X} \subset \mathbb{R}^d$, is a vector collecting the system parameters. The goal is to build a compact and accurate mathematical representation $\tilde{\mathcal{M}}$ of the system (1), starting from a limited set of realizations $\{(\mathbf{x}_i, y_i)\}_{i=1}^{L}$ such that:

$$y \approx \tilde{\mathcal{M}}(\mathbf{x}), \tag{2}$$

Manuscript received ...; revised ...

R. Trinchero, P. Manfredi, I. S. Stievano and F. G. Canavero are with the Department of Electronics and Telecommunications, Politecnico di Torino, Torino 10129, Italy (e-mail: riccardo.trinchero@polito.it, paolo.manfredi@polito.it, igor.stievano@polito.it, flavio.canavero@polito.it).

where $\tilde{\mathcal{M}}$ is the so called macromodel or surrogate model of the real system $\mathcal{M}$.

This paper considers and compares two different techniques to achieve the above goal: the PC expansion and the SVM regression. The accuracy and convergence of these two methods, as well as their robustness to noise, are carefully assessed.

III. Polynomial Chaos Expansion

According to the PC theory, the output of the system (1) is approximated by an expansion

$$y \approx \tilde{\mathcal{M}}_{PC}(\mathbf{x}) = \sum_{k=0}^{K} c_k \varphi_k(\mathbf{x}), \tag{3}$$

where the functions $\{\varphi_k\}_{k=0}^{K}$ are polynomials with a maximum total degree p that form an orthonormal basis. The number of terms in (3) is related to the maximum degree and the number of parameters n as [36]

$$K+1 = \frac{(p+n)!}{p!n!}. \tag{4}$$

Since the PC is specifically designed for variability analysis, the basis functions are taylored so that (3) is accurate in a statistical sense. Nevertheless, (3) can be also thought of as a parametric representation of the system output. The choice of the basis functions is related to the *probability distribution* of the parameters $\mathbf{x}$. Denoting with $w(\mathbf{x})$ the joint probability density function (PDF) of $\mathbf{x}$, the following inner product is defined

$$\langle f, g \rangle = \int_{-\infty}^{+\infty} f(\mathbf{x}) g(\mathbf{x}) w(\mathbf{x}) d\mathbf{x}. \tag{5}$$

By choosing polynomials that are orthogonal based on the inner product (5), the approximation error

$$||y - \tilde{\mathcal{M}}_{PC}(\mathbf{x})||_{L_2} = \sqrt{\langle y - \tilde{\mathcal{M}}_{PC}(\mathbf{x}), y - \tilde{\mathcal{M}}_{PC}(\mathbf{x}) \rangle} \tag{6}$$

converges exponentially to zero for smooth functions [36]. For standard PDFs and statistically independent parameters, the orthogonal polynomials are well known in mathematics. These include for example Hermite polynomials for Gaussian random parameters ($w(\mathbf{x}) = e^{-\frac{1}{2}||\mathbf{x}||^2}/\sqrt{(2\pi)^n}$) and Legendre polynomials for uniform random parameters ($w(\mathbf{x}) = (1/2)^n$ for $|x_i| \leq 1$, $i = 0, \ldots, n$).[1]

With the above definitions, the expected value and the variance of y can be obtained from the PC expansion (3) as

$$\mathrm{E}\{y\} \approx \mathrm{E}\{\tilde{\mathcal{M}}(\mathbf{x})\} = \int_{-\infty}^{+\infty} \tilde{\mathcal{M}}(\mathbf{x}) w(\mathbf{x}) d\mathbf{x} = c_0 \tag{7}$$

and

$$\begin{aligned} \mathrm{Var}\{y\} &\approx \mathrm{Var}\{\tilde{\mathcal{M}}(\mathbf{x})\} \\ &= \int_{-\infty}^{+\infty} (\tilde{\mathcal{M}}(\mathbf{x}) - \mathrm{E}\{\hat{y}\})^2 w(\mathbf{x}) d\mathbf{x} = \sum_{k=1}^{K} c_k^2, \end{aligned} \tag{8}$$

[1]For the sake of simplicity, and without loss of generality, all the parameters are assumed to be normalized so that they have zero mean and the same standardized variance.

respectively. Other statistical moments can be obtained numerically by integrating or sampling the surrogate model (3).

Several techniques can be used to estimate the PC coefficients in (3), including the stochastic Galerkin method [8], [10], stochastic testing [7], [11] and other sampling-based approaches based on linear regression [9], [12], [13]. Without loss of generality, the linear regression approach is adopted here, since it operates in a similar fashion as the SVM regression that is discussed next. The original system (1) is evaluated for L randomly chosen samples of the parameters $\mathbf{x}$, producing a set of observations $\{(\mathbf{x}_i, y_i)\}_{i=1}^{L}$. Next, the PC expansion (3) is required to fit these observations in a least-square sense. Therefore, the system

$$\mathbf{A}\mathbf{c} = \mathbf{b}, \tag{9}$$

where $\mathbf{c} \in \mathbb{R}^{K+1}$ is a vector collecting the unknown PC expansion coefficients, $\mathbf{b} \in \mathbb{R}^{L}$ is a vector collecting the observations $\{y_i\}_{i=1}^{L}$, and $\mathbf{A} \in \mathbb{R}^{L \times (K+1)}$ is a Vandermonde-like matrix with the basis polynomials evaluated at the random parameter samples, i.e., $[\mathbf{A}]_{ik} = \varphi_k(\mathbf{x}_i)$, is solved by regression as

$$\mathbf{c} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{b}. \tag{10}$$

For an accurate evaluation, the system (9) needs to be *overdetermined* (i.e., $L \gg K+1$). A number of observations that is double the number of unknown PC coefficients (i.e., $L = 2(K+1)$) is typically suggested [9].

IV. SVM Regression: Essential Background

The aim of this section is to provide a quick overview of the basic concepts behind the SVM regression (also known as ε-regression), which is a well-consolidated technique belonging to the class of machine learning tools [33], [34].

The discussion starts by considering a training set of observations $\{(\mathbf{x}_i, y_i)\}_{i=1}^{L}$. The SVM regression allows building a compact approximation $\tilde{\mathcal{M}}_{SVM}$ of the original system (1), which approximates the given training pairs with a maximum deviation ε from the target samples y_i. This means that a function $\tilde{\mathcal{M}}_{SVM}(\mathbf{x})$ must be searched for, such that:

$$|y_i - \tilde{\mathcal{M}}_{SVM}(\mathbf{x}_i)| \leq \varepsilon. \tag{11}$$

A. Linear Regression

The simplest solution is to define $\tilde{\mathcal{M}}_{SVM}$ in terms of the linear regression

$$\tilde{\mathcal{M}}_{SVM}(\mathbf{x}) = \langle \mathbf{w}, \mathbf{x} \rangle + b, \tag{12}$$

where $\mathbf{w} = [w_1, \ldots, w_d] \in \mathbb{R}^d$, while

$$\langle \mathbf{w}, \mathbf{x} \rangle = \sum_{n=1}^{d} w_n x_n \tag{13}$$

denotes in this case the inner product in the $\mathbb{R}^d$-space. With the above definition, equation (12) can be rewritten as

$$\tilde{\mathcal{M}}_{SVM}(\mathbf{x}) = \sum_{n=1}^{d} w_n x_n + b. \tag{14}$$

In order to ensure a sufficient flatness of the function $\tilde{\mathcal{M}}_{SVM}$, the Euclidean norm $\|\mathbf{w}\|^2$ must be kept as small as possible, leading to the following optimization problem:

$$\begin{aligned} &\text{minimize } \frac{1}{2}\|\mathbf{w}\|^2 \\ &\text{subject to } \begin{cases} y_i - \langle \mathbf{w}, \mathbf{x}_i \rangle - b \leq \varepsilon \\ \langle \mathbf{w}, \mathbf{x}_i \rangle + b - y_i \leq \varepsilon. \end{cases} \end{aligned} \tag{15}$$

In the above equation, we are implicitly assuming that the linear form (12) approximates all the training data with a precision bounded by ε [32]. In other words, we are assuming that the convex optimization problem (15) is feasible. However, this is not necessarily true for a generic set of training data coming from realistic (e.g., nonlinear) problems.

To relax the error constraint in (11), the optimization problem (15) is recast as the minimization of the following risk function:

$$\begin{aligned} R_{emp}(\mathbf{w}, b) =& \frac{1}{L}\sum_{i=1}^{L} |y_i - \tilde{\mathcal{M}}_{SVM}(\mathbf{x}_i)|_\varepsilon \\ =& \frac{1}{L}\sum_{i=1}^{L} |y_i - \langle \mathbf{w}, \mathbf{x}_i \rangle - b|_\varepsilon, \end{aligned} \tag{16}$$

where $|y_i - \tilde{\mathcal{M}}_{SVM}(\mathbf{x}_i)|_\varepsilon$ is the so-called linear ε-intensive loss function [33], which is defined as

$$\begin{aligned} &|y_i - \tilde{\mathcal{M}}_{SVM}(\mathbf{x}_i)|_\varepsilon = \\ &= \begin{cases} 0, & \text{if } |y_i - \tilde{\mathcal{M}}_{SVM}(\mathbf{x}_i)| \leq \varepsilon \\ |y_i - \tilde{\mathcal{M}}_{SVM}(\mathbf{x}_i)| - \varepsilon, & \text{otherwise.} \end{cases} \end{aligned} \tag{17}$$

Minimizing the risk function (16) is equivalent to finding the pair $(\mathbf{w}, b)$ in (12) that minimizes the deviation of the macro-model from the training samples outside the ε-intensive zone. This can be done via the following optimization problem [33]

$$\begin{aligned} &\text{minimize } \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{L}(\xi_i + \xi_i^*) \\ &\text{subject to } \begin{cases} y_i - \langle \mathbf{w}, \mathbf{x}_i \rangle - b \leq \varepsilon + \xi_i \\ \langle \mathbf{w}, \mathbf{x}_i \rangle + b - y_i \leq \varepsilon + \xi_i^* \\ \xi_i, \xi_i^* \geq 0, \end{cases} \end{aligned} \tag{18}$$

where ξ_i, ξ_i^* are slack variables that indicate a positive and negative deviation of the training samples lying outside the ε-intensive zone, whereas C is a parameter providing a trade-off between the accuracy of the model and its flatness. As an example, a large value of C can be used to enforce the model to exactly fit all the training samples disregarding the flatness of the model. This parameter needs to be empirically chosen by the user [24]. Fig. 1 shows a graphical illustration of the slack variables ξ_i and ξ_i^* along with their role in the ε-intensive loss function (17).

The optimization problem with inequality constraints in (18) is transformed into its dual problem [24]- [34] and then solved by minimizing the corresponding Lagrangian function.

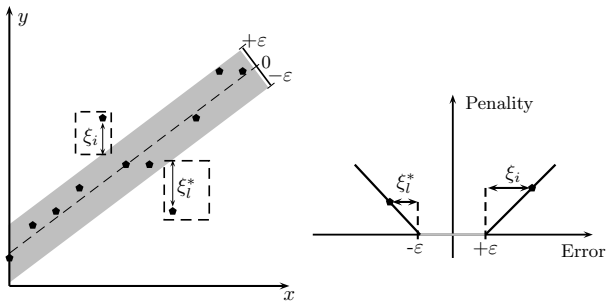

Fig. 1. Graphical interpretation of the minimization problem (16) and of the role of function (17) (inspired by [24], [28]).

The solution allows estimating the optimum $\mathbf{w}$ via a linear combination of the training parameters $\mathbf{x}_i$, which reads

$$\mathbf{w} = \sum_{i=1}^{L}(\alpha_i - \alpha_i^*)\mathbf{x}_i, \tag{19}$$

where $\alpha_i, \alpha_i^* \in [0, C]$ are the pertinent Lagrange multipliers related to the constraints of the optimization problem. Additional details on the Lagrangian minimization and on the dual problem formulation are available in [33]- [32] and are out of the scope of this short overview.

Substituting (19) into (12) yields

$$\tilde{\mathcal{M}}_{SVM}(\mathbf{x}) = \sum_{i=1}^{L}(\alpha_i - \alpha_i^*)\langle \mathbf{x}_i, \mathbf{x} \rangle + b. \tag{20}$$

The above equation is the so-called *support vector expansion*. It is important to remark that the complexity of $\tilde{\mathcal{M}}_{SVM}$ does not depend on the dimensionality d, but only on the number of training samples L [32].

B. Non-Linear Regression

The linear formulation in (20) can be generalized to the case of nonlinear regression [24], [34] as follows:

$$\tilde{\mathcal{M}}_{SVM}(\mathbf{x}) = \langle \tilde{\mathbf{w}}, \mathbf{\Phi}(\mathbf{x}) \rangle + b, \tag{21}$$

where $\mathbf{\Phi}(\mathbf{x}) = [\phi_1(\mathbf{x}), \ldots, \phi_D(\mathbf{x})]$ is a nonlinear map $\mathbb{R}^d \rightarrow \mathbb{R}^D$, $\tilde{\mathbf{w}} \in \mathbb{R}^D$ is a generalization of $\mathbf{w}$ in (12), and $\langle \tilde{\mathbf{w}}, \mathbf{\Phi}(\mathbf{x}) \rangle$ is the inner product in $\mathbb{R}^D$. Hence, the above equation can be rewritten as:

$$\tilde{\mathcal{M}}_{SVM}(\mathbf{x}) = \sum_{n=1}^{D} \tilde{w}_n \phi_n(\mathbf{x}) + b. \tag{22}$$

It is important to remark that the dimensionality D of the feature space is defined by the nonlinear map $\Phi(\mathbf{x})$, and it is completely independent from the number of training pairs L and from the the number of system parameters d. Also, it should be noted that (22) is linear w.r.t the transformation $\mathbf{\Phi}(\mathbf{x})$. Therefore, the parameters $\tilde{\mathbf{w}}$ can be estimated again as in (19):

$$\tilde{\mathbf{w}} = \sum_{i=1}^{L}(\alpha_i - \alpha_i^*)\mathbf{\Phi}(\mathbf{x}_i). \tag{23}$$

Substituting (23) in (21) leads to

$$\begin{aligned}\tilde{\mathcal{M}}_{SVM}(\mathbf{x}) &= \sum_{i=1}^{L} \beta_i \langle \mathbf{\Phi}(\mathbf{x}_i)\mathbf{\Phi}(\mathbf{x})\rangle + b \\ &= \sum_{n=1}^{D}\sum_{i=1}^{L} \beta_i \Phi_n(\mathbf{x}_i)\Phi_n(\mathbf{x}) + b \\ &= \sum_{i=1}^{L} \beta_i K(\mathbf{x}_i,\mathbf{x}) + b, \end{aligned} \tag{24}$$

where $\beta_i = (\alpha_i - \alpha_i^*)$. The so-called kernel function $K(\mathbf{x}_i,\mathbf{x}) = \langle \mathbf{\Phi}(\mathbf{x}_i), \mathbf{\Phi}(\mathbf{x})\rangle$ is defined as the inner product in the feature space $\mathbb{R}^D$ between the functions $\mathbf{\Phi}(\mathbf{x})$ evaluated at a given training sample $\mathbf{x}_i$ and the same function at a generic point $\mathbf{x} \in \mathbb{R}^d$, i.e.,

$$K(\mathbf{x}_i,\mathbf{x}) = \sum_{n=1}^{D} \phi_n(\mathbf{x}_i)\phi_n(\mathbf{x}). \tag{25}$$

The computation of the above inner product is extremely inefficient when the dimensionality of the feature space D is high (for some particular kernels, D can even grow to infinity!) However, it is important to point out that (24) does not require an explicit calculation of the inner product (25), but it can be efficiently estimated by operating directly in the d-space through the kernel functions $K(\mathbf{x}_i,\mathbf{x})$. Indeed, differently from the nonlinear map $\mathbf{\Phi}$, the kernel is a simple multivariate function $\mathbb{R}^d \rightarrow \mathbb{R}$, which is independent on the dimensionality D of the feature space. This means that the SVM nonlinear regression is suitably defined by the kernel without requiring an explicit definition of the nonlinear transformation $\mathbf{\Phi}(\mathbf{x})$. This is the so-called *kernel trick*.

The most common kernels are listed below [33]:

- linear[2]: $K(\mathbf{x}_i,\mathbf{x}) = \mathbf{x}_i^T\mathbf{x}$;
- polynomial of order q: $K(\mathbf{x}_i,\mathbf{x}) = (1 + \mathbf{x}_i^T\mathbf{x})^q$;
- Gaussian: $K(\mathbf{x}_i,\mathbf{x}) = \exp\left(-\|\mathbf{x}_i - \mathbf{x}\|^2\right)$.

It is important to point out that the SVM regression is already a mature tool that is included in commercial software like MATLAB. For instance, the MATLAB machine learning toolbox includes the SVM training function `fitrsvm`, embedding the aforementioned three classes of kernels. The default values of the parameters are $C = 1$ and $\varepsilon \propto \text{Var}\{y_i\}$. The corresponding surrogate model can be evaluated for an arbitrary value of the parameters $\mathbf{x}$ via the function `predict`.

V. Benchmark analytical function

As a first illustrative example, the PC expansion and SVM regression are applied to the synthetic function

$$y = \mathcal{M}(\mathbf{x}) = \frac{-1}{1.2 + 0.6x_1^2 + 0.3x_2}, \tag{26}$$

where the parameters $\mathbf{x} = [x_1, x_2]$ are uniformly distributed in the domain $\mathcal{X} = [-1, 1]^2$.

The PC expansion and the SVM regression presented in Sections III and IV are used to build the two different surrogate models denoted with $\tilde{\mathcal{M}}_{PC}$ and $\tilde{\mathcal{M}}_{SVM}$, respectively. Owing to the uniform distribution of the parameters, the PC model is constructed by using Legendre polynomials. A maximum total degree $p = 6$ is considered. For the SVM regression, a polynomial kernel of the same order (i.e, $q = 6$) is used, with $C = 1$ and $\varepsilon = \texttt{eps}$ (i.e., the machine precision) to maximize the model accuracy. Both models are estimated starting from the same set of *randomly* chosen samples $\{(\mathbf{x}_i, y_i)\}_{i=1}^{L}$,for increasing set sizes L.

The two surrogate models are used to predict the probability density function of y in (26). Since the surrogates necessarily depend on the specific training samples that are used to build them, ten different realizations of the training set are considered for a given size L. Fig. 2 compares the PDF of y obtained from 10000 MC evaluations of (26) (solid black line) with those estimated from the surrogate models trained with $L = 20$, 30 and 40 samples (dashed lines). Even though this is a simple analytical example, it is representative of a number of real life applications (e.g., data from measurements) in which only a relatively small number of random realizations are available due to budget and/or time constraints. For each model, the two lines represent the minimum and maximum value of the PDF as resulting from the ten different realizations of the training set. A small number of training samples leads to a larger fluctuation of the model response. Nonetheless, the above comparison highlights a remarkable accuracy of the SVM regression in capturing the main features of the reference PDF even when the size of the training set is small.

The reason of this different performance resides in the inherent features of the two methods. In the case of the PC expansion, the model parameters are computed via the solution of a least-square problem, which – as discussed in Section III – yields better results for overdetermined systems (i.e., when the number of training samples is greater than the number of unknowns in the polynomial expansion). According to (4), for a PC expansion of order $p = 6$ and $n = 2$ random parameters, the number of PC expansion coefficients is 28. On the other hand, the SVM model is the result of an optimization problem, which is undoubtedly less efficient than a least-square solution, but it exhibits a better convergence. Additional details on both the efficiency and the convergence rate of the two methods are given in the next section.

Fig. 3 further compares the surface of (26) in the $\mathcal{X}$-space obtained from the PC and SVM models trained with $L = 40$ samples. Additionally, the color provides the information on the relative error w.r.t. the actual value of the function. The plots highlight the good capability of both techniques in reconstructing the shape of the nonlinear function (26) with a relative error $< 3\%$. For this sample size, the two models are essentially equivalent in terms of relative error.

VI. Application Test Case

This section discusses the application of the PC and SVM techniques to the case of a realistic high-speed communication link, like the one depicted in Fig. 4. The network consists of the interconnection of three PCB traces and lumped elements. The traces are modeled as three uncoupled sections of lossy transmission lines [35]. The value of some lumped

[2]The linear kernel corresponds to the linear regression in (12).

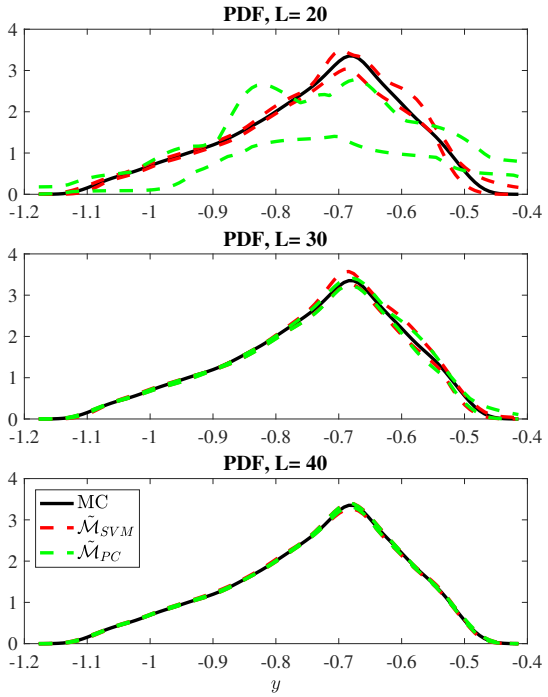

Fig. 2. PDF of the benchmark function (26). The result computed from 10000 MC samples (solid black curves) is compared with the predictions achieved via the SVM regression (dashed red curves) and the PC expansion (dashed green curves). The three panels correspond to different training set sizes.

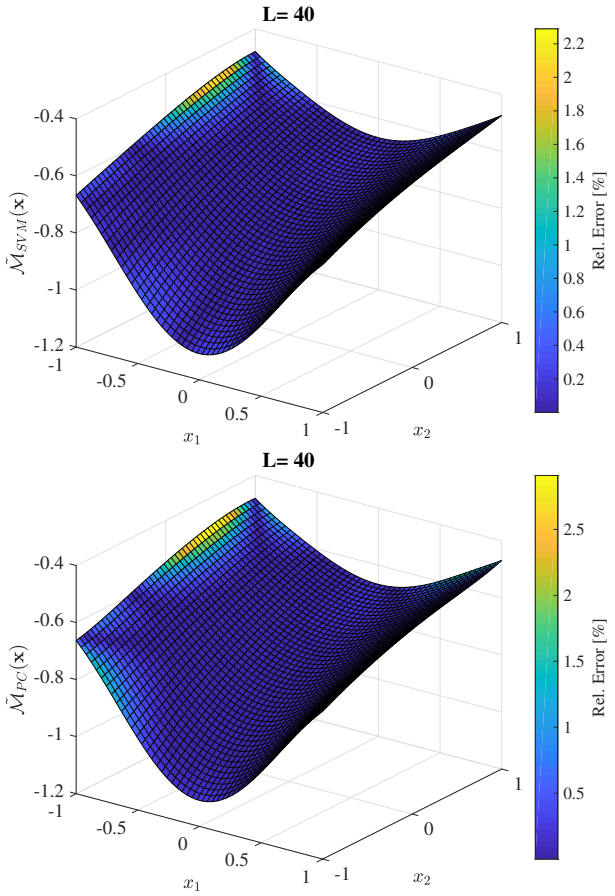

Fig. 3. Parametric responses of models $\tilde{\mathcal{M}}_{SVM}$ and $\tilde{\mathcal{M}}_{PC}$, both constructed based on the same 40 training pairs, and corresponding relative errors.

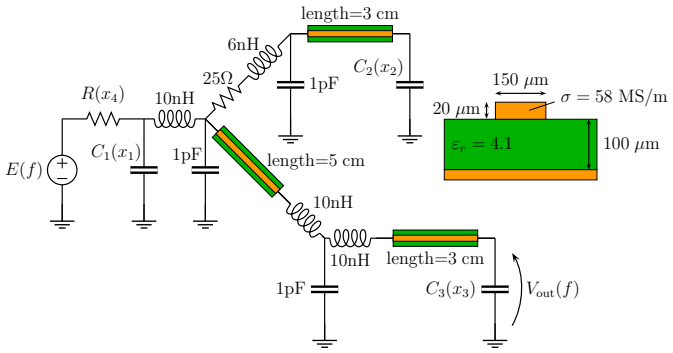

Fig. 4. Schematic of the high-speed link with lossy transmission lines.

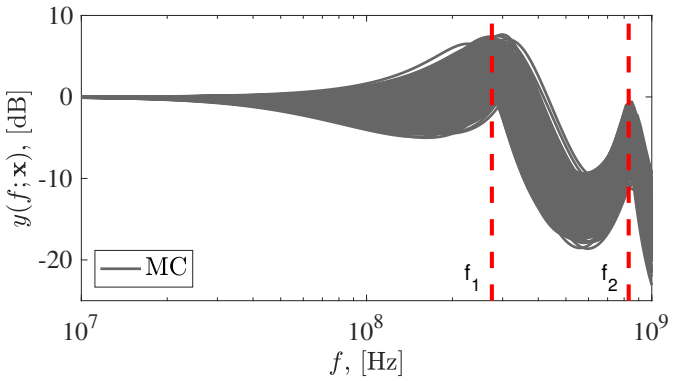

Fig. 5. Variability of the frequency-domain response of the circuit of Fig. 4.

components is parametric, namely $C_1(x_1) = (5 + 2.5x_1)$ pF, $C_2(x_2) = (5 + 2.5x_2)$ pF, $C_3(x_3) = (5 + 2.5x_3)$ pF and $R(x_4) = (50 + 25x_4)\,\Omega$. The parameters $\mathbf{x} = [x_1, x_2, x_3, x_4]$ are uniformly distributed in $\mathcal{X} = [-1, 1]^4$. Therefore, each parameter exhibits a uniform variation of $\pm 50\%$ around its central value. The system output to be modeled is the transfer function

$$y(f;\mathbf{x}) = \mathcal{M}(f;\mathbf{x}) = \left| \frac{V_{out}(f;\mathbf{x})}{E(f)} \right| \tag{27}$$

between the output voltage V_{out} and the excitation $E(f)$.

Fig. 5 shows the large spread of the transfer function (27) resulting from the high variability of the four stochastic parameters, and obtained by MC simulation in the frequency range $f \in [10\,\text{MHz}, 1\,\text{GHz}]$.

A small subset of the MC realizations has been used to train the PC and SVM surrogate models, denoted again as $\tilde{\mathcal{M}}_{PC}(f;\mathbf{x})$ and $\tilde{\mathcal{M}}_{SVM}(f;\mathbf{x},)$, respectively. The former uses Legendre polynomials with maximum degree $p = 5$. On the other hand, the SVM macromodel uses a polynomial kernel of degree $q = 5$, with $C = 1$ and $\varepsilon = \texttt{eps}$ to maximize the model accuracy. In order to assess the convergence rate of the two methods, the models are trained with an increasing number of samples, i.e., $L = 30, 50, 100, 150$. Since the accuracy of both the PC and SVM regression depends on the training samples, also in this case, ten different realizations of the training set are examined for a given size L.

Next, the surrogate models are used to compute the PDF of the transfer function (27) at the frequencies $f_1 = 275\,\text{MHz}$ and $f_2 = 830\,\text{MHz}$, which are indicated by the dashed vertical

TABLE I
COMPARISON ON THE COMPUTATIONAL COST AND THE ACCURACY OF THE PC AND SVM MACROMODELS AT $f_2 = 830\,\text{MHz}$.

Method	# of training samples L	Model construction [s]	Model evaluation [s]	Max rel. error%
SVM (order 5)	30	0.08s	1.60s	100%
	50	0.01s	1.63s	99%
	100	0.09s	1.64s	33%
	150	4.30s	1.66s	13%
PC (order 5)	30	0.07s	0.04s	>100%
	50	0.11s	0.04s	>100%
	100	0.22s	0.03s	>100%
	150	0.33s	0.04s	17%

lines in Fig. 5. The results are collected in Fig. 6 and highlight again the better accuracy of the SVM regression (dashed red lines) w.r.t. the PC expansion (dashed green line) in predicting the reference MC result (solid black line) from a very small training set size (e.g., $L = 30$ or 50). Indeed, the PC model starts to converge to the actual PDF only when the number of training samples is greater than the number of unknowns in the regression problem, i.e., for $L > 126$ in this case. For a large number of training samples, both models converge to the actual PDF.

The convergence rate and the computational cost of the two macromodels are systematically investigated in Table I. The table considers the computational time required to build and evaluate the PC and the SVM models at a given point $\mathbf{x}$ in the parameter space $\mathcal{X}$, as well as the maximum error of the parametric models over ten different realizations of the training set at the frequency $f_2 = 830\,\text{MHz}$. The figures highlight the better efficiency of the PC approach, but also confirm the fast convergence of the SVM regression w.r.t. the training set size.

It is important to stress that the presented tools are sampling based and hence consider the system as a "black box". Therefore, their performance is virtually independent from the complexity of the circuit being modeled. The computational effciency only depends on the number of design variables, on the kernel and on the number of training samples.

Finally, to investigate the robustness of the presented techniques to noisy data, the training samples are corrupted by a Gaussian noise as follows:

$$\tilde{y}_i(f;\mathbf{x}_i) = y_i(f;\mathbf{x}_i) \times (1+\xi_n), \tag{28}$$

where $\xi_n \sim \mathcal{N}(0,\sigma_n^2)$ is a Gaussian variable with $\sigma_n = 0.025$.

As in the previous case, the surrogate models $\tilde{\mathcal{M}}_{PC}(f;\mathbf{x})$ and $\tilde{\mathcal{M}}_{SVM}(f;\mathbf{x})$ are estimated with an increasing number of training samples, i.e., $L = 50, 100, 150, 200$, by considering ten different realizations for each case. In the presence of noise, it is useful to set a non-negligible value of ε in the SVM regression. Indeed, according to the ε-intensive loss function (17), a deviation of the training samples within the bound $[-\varepsilon, +\varepsilon]$ does not affect the SVM model. The empirical relation $\varepsilon = 3\sigma_n\sqrt{\log(L)/L}$ is suggested in [24] to minimize the perturbation of the model accuracy due to noise. In such a way, the width of the ε-intensive zone decreases by increasing the number of training samples.

Fig. 7 compares the PDFs previously obtained with MC (without noise) against the ones predicted by the PC and SVM macromodels constructed from noisy training samples $(\tilde{y}_i(f), \mathbf{x}_i)$. These results confirm the assessment of the noiseless case. Indeed, the SVM regression provides again a better convergence than the PC expansion when a small set of training samples is available (e.g., $L = 50$ and 100). However, as expected, when the number of samples is increased, the two macromodels converge to the same solution.

VII. CONCLUSIONS

This paper discusses an efficient approach for the generation of a surrogate model of a system that depends on highly variable parameters, starting from a limited number of response samples. The approach is based on the SVM ε-intensive regression, which is a powerful tool belonging to the class of machine learning methods.

The main features of the SVM regression have been systematically discussed and investigated by comparing its accuracy, convergence and robustness to noise against one of the most important state-of-the-art techniques, namely PC. From the illustrated results, the SVM regression can be considered as a viable solution for the parametric macromodeling of systems with a limited number of uncertain parameters characterized by a large variability, especially when a small number of training samples is available. Indeed, in this case the SVM regression substantially outperforms the PC expansion method in terms of accuracy. An extensive analysis of the performance of the advocated SVM approach w.r.t. the dimensionality of the parameter space has still to be performed.

REFERENCES

[1] R. Spence and R. S. Soin, *Tolerance Design of Electronic Circuits.* London: Imperial College Press, 1997.

[2] H. Kettani and B. Barmish, "A new Monte Carlo circuit simulation paradigm with specific results for resistive networks," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 53, no. 6, pp. 1289–1299, Jun. 2006.

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

[4] D. Deschrijver and T. Dhaene, "Fully parameterized macromodeling of S-parameter data by interpolation of numerator & denominator," *IEEE Microw. Wireless Compon. Lett.*, vol. 22, no. 6, pp. 309–311, Jun. 2012.

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

[6] S. Grivet-Talocia and R. Trinchero, "Behavioral, Parameterized, and Broadband Modeling of Wired Interconnects With Internal Discontinuities," *IEEE Trans. Electromagn. Compat.*, vol. 60, no. 1, pp. 77–85, Feb. 2018.

[7] Z. Zhang, T. A. El-Moselhy, I. M. Elfadel, and L. Daniel, "Stochastic testing method for transistor-level uncertainty quantification based on generalized polynomial chaos," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 32, no. 10, pp. 1533–1545, Oct. 2013.

[8] M. R. Rufuie, E. Gad, M. Nakhla, and R. Achar, "Generalized Hermite polynomial chaos for variability analysis of macromodels embedded in nonlinear circuits," *IEEE Trans. Compon. Packag. Manuf. Techol.*, vol. 4, no. 4, pp. 673–684, Apr. 2014.

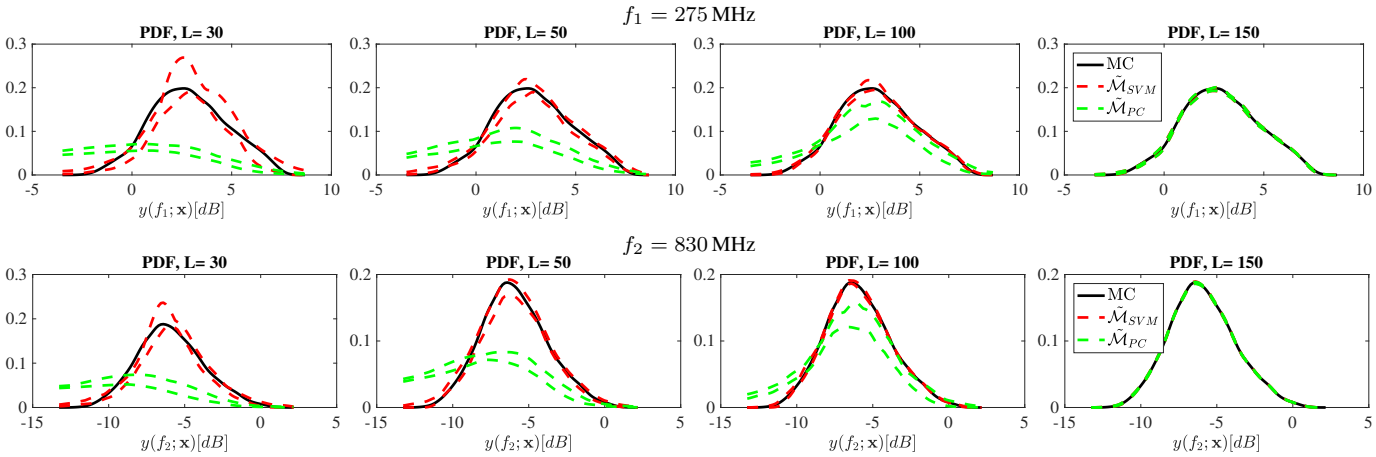

Fig. 6. PDF of the transfer function (27) evaluated at $f_1 = 275$ MHz and $f_2 = 830$ MHz. The result obtained from 10000 MC simulations (solid black curves) is compared with the predictions from the SVM regression (dashed red curves) and the PC expansion (dashed green curves) for an increasing number of training pairs.

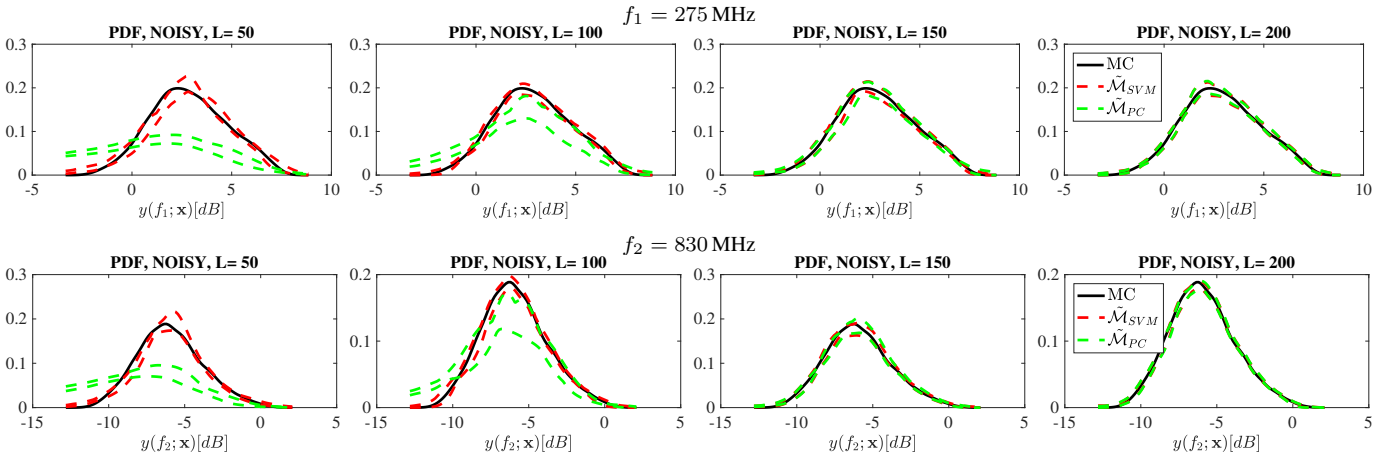

Fig. 7. PDF of the transfer function (27). The result obtained from noise-free MC samples (solid black curves) is compared with the predictions from the SVM regression (dashed red curves) and the PC expansion (dashed green curves) constructed from noisy training samples of increasing size.

[9] D. Spina, D. De Jonghe, D. Deschrijver, G. Gielen, L. Knockaert, and T. Dhaene, "Stochastic macromodeling of nonlinear systems via polynomial chaos expansion and transfer function trajectories," *IEEE Trans. Microw. Theory Techn.*, vol. 62, no. 7, pp. 1454–1460, Jul. 2014.

[10] P. Manfredi, D. Vande Ginste, D. De Zutter, and F. G. Canavero, "Stochastic modeling of nonlinear circuits via SPICE-compatible spectral equivalents," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 61, no. 7, pp. 2057–2065, Jul. 2014.

[11] P. Manfredi, D. Vande Ginste, D. De Zutter, and F. G. Canavero, "Generalized decoupled polynomial chaos for nonlinear circuits with many random parameters," *IEEE Microw. Wireless Compon. Lett.*, vol. 25, no. 8, pp. 505–507, Aug. 2015.

[12] M. Ahadi and S. Roy, "Sparse linear regression (SPLINER) approach for efficient multidimensional uncertainty quantification of high-speed circuits," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 35, no. 10, pp. 1640–1652, Oct. 2016.

[13] M. Larbi, I. S. Stievano, F. G. Canavero and P. Besnier, "Variability Impact of Many Design Parameters: The Case of a Realistic Electronic Link," *IEEE Trans. Electromagn. Compat.*, vol. 60, no. 1, pp. 34-41, Feb. 2018.

[14] N. Femia and G. Spagnuolo, "Genetic optimization of interval arithmetic based worst case circuit tolerance analysis," *IEEE Trans. Circuits Syst. I, Fundam. Theory Appl.*, vol. 46, no. 12, pp. 1441–1456, Dec. 1999.

[15] N. Femia and G. Spagnuolo, "True worst-case circuit tolerance analysis using genetic algorithms and affine arithmetic," *IEEE Trans. Circuits Syst. I, Fundam. Theory Appl.*, vol. 47, no. 9, pp. 1285–1296, Sep. 2000.

[16] M. Wu, D. G. Beetner, T. H. Hubing, H. Ke, and S. Sun, "Statistical prediction of "reasonable worst-case" crosstalk in cable bundles," *IEEE Trans. Electromagn. Compat.*, vol. 51, no. 3, pp. 842–851, Aug. 2009.

[17] T. Ding, R. Trinchero, P. Manfredi, I. S. Stievano, and F. G. Canavero, "How affine arithmetic helps beat uncertainties in electrical systems," *IEEE Circuits Syst. Mag.*, vol. 15, no. 4, pp. 70–79, Nov. 2015.

[18] R. Trinchero, P. Manfredi, T. Ding, and I. S. Stievano, "Combined parametric and worst-case circuit analysis via Taylor models," *IEEE Trans. Circuits Syst. I, Fundam. Theory Appl.*, vol. 63, no. 7, pp. 1067–1078, Jul. 2016.

[19] R. Trinchero, P. Manfredi, and I. S. Stievano, "TMsim: an algorithmic tool for the parametric and worst-case simulation of systems with uncertainties," *Math. Problems Eng.*, vol. 2017, Article ID 6739857, 12 pages, 2017.

[20] S. De Ridder *et al.*, "A generative modeling framework for statistical link analysis based on sparse data," *IEEE Trans. Compon. Packag. Manuf. Techol.* (in press).

[21] M. Sabzekar, H. Sadoghi Yazdi and M. Naghibzadeh, "Relaxed constraints support vector machines for noisy data," *Neural Comput. & Applic.*, vol. 20, no. 5, pp. 671–685, 2011.

[22] J. M. Moguerza and A. Muoz, "Support Vector Machines with Applications," *Statist. Sci.*, vol 21, no. 3, pp. 322–336, 2006.

[23] Y. B. Yuan, W. Q. Qiu, Y. J. Wang, J. Gao and P. He, "Classification of heart failure with Polynomial Smooth Support Vector Machine," in *Proc. Int. Conf. on Machine Learning and Cybernetics*, Ningbo, China, 2017, pp. 48-54.

[24] V. Cherkassky and Y Ma, "Practical selection of SVM parameters and noise estimation for SVM regression", *Neural Networks*, vol. 17, no. 1, pp. 113–126, 2004.

[25] M. Cordina and C. J. Debono, "A support vector machine based sub-

band CQI feedback compression scheme for 3GPP LTE systems," in *Proc. 2017 Inter. Symp. on Wireless Communication Systems*, Bologna, Italy, 2017, pp. 325–330.
[26] M. Arefi and B. Chowdhury, "Ensemble adaptive neuro fuzzy support vector machine for prediction of transient stability," in *Proc. of North American Power Symposium*, Morgantown, WV, USA, 2017, pp. 1–6.
[27] L. Wang, D. Wang and C. Hao, "Intelligent CFAR Detector Based on Support Vector Machine," *IEEE Access*, (early access).
[28] V. R. Kohestani and M. Hassanlourad, "Modeling the mechanical behavior of carbonate sands using artificial neural networks and support vector machine," *Int. J. of Geomech.*, vol. 16, no. 1, pp. 1–9, 2016.
[29] D. Plets *et al.*, *Surrogate modeling based cognitive decision engine for optimization of WLAN performance, Wireless Networks*, vol. 23, no. 8, pp. 2347–2359, May. 2017.
[30] 'E. Debry and V. Mallet, *Ensemble forecasting with machine learning algorithms for ozone, nitrogen dioxide and PM_{10} on the Prev'Air platform, Atmospheric Environment*, vol. 91, pp. 71–84, 2014.
[31] T. Baoping, S. Tao, L. Feng and D. Lei, "Fault diagnosis for a wind turbine transmission system based on manifold learning and Shannon wavelet support vector machine," *Renewable Energy*, vol. 62, pp. 1–9, 2014.
[32] A. J. Smola and B. Schlkopf, "A tutorial on support vector regression", *Statistics and Computing*, vol. 14, no. 3, pp. 199–222, 2004.
[33] V. Vapnik, *The Nature of Statistical Learning Theory*, 2nd edition, Springer, 1999.
[34] V. Vapnik, *Statistical Learning Theory*, John Wiley and Sons, NY, 1998.
[35] C. R. Paul, *Analysis of Multiconductor Transmission Lines*, John Wiley and Sons, NY, 1994.
[36] D. Xiu and G. E. Karniadakis, "The Wiener-Askey polynomial chaos for stochastic differential equations," *SIAM, Journal of Sci. Computation*, vol. 24, no. 2, pp. 619–622, 2002.

Riccardo Trinchero (M'16) was born in Casale Monferrato, Italy in 1987. He received the M.Sc. and the Ph.D. degrees in Electronics and Communication Engineering from Politecnico di Torino, Torino, Italy, in 2011 and 2015, respectively.

He is currently a Researcher within the EMC Group with the Department of Electronics and Telecommunications at the Politecnico di Torino. His research interests include the analysis of linear time-varying systems, modeling and simulation of switching converters and statistical simulation of circuits and systems.

Paolo Manfredi (S'10-M'14) received the M.Sc. degree in electronic engineering from the Politecnico di Torino, Turin, Italy, in 2009, and the Ph.D. degree in information and communication technology from the Scuola Interpolitecnica di Dottorato, Politecnico di Torino, in 2013. From 2013 to 2014, he was a Post-Doctoral Researcher with the EMC Group, Politecnico di Torino. From 2014 to 2017, he was a Post-Doctoral Research Fellow with the Research Foundation-Flanders with the Electromagnetics Group, Department of Information Technology, Ghent University, Ghent, Belgium. He is currently an Assistant Professor with the EMC Group, Department of Electronics and Telecommunications, Politecnico di Torino. His research interests include the several aspects of circuit and interconnect modeling and simulation, including statistical and worst-case analysis, signal integrity, and electromagnetic compatibility.

Dr. Manfredi received the Best Paper Award at the 2016 IEEE Electrical Design of Advanced Packaging and Systems Symposium, the Best Oral Paper Award and the Best Student Paper Award at the 22nd and 19th IEEE Conference on Electrical Performance of Electronic Packaging and Systems, respectively, the Young Scientist Award at the XXX International Union of Radio Science General Assembly and Scientific Symposium, and an Honorable Mention at the 2011 IEEE Microwave Theory and Techniques Society International Microwave Symposium.

Igor S. Stievano (M?98 ? SM?08) received the Master degree in electronic engineering and the Ph.D. degree in electronics and communication engineering from the Politecnico di Torino, Torino, Italy, in 1996 and in 2001, respectively. Currently he is an Associate Professor of Circuit Theory with the Department of Electronics and Telecommunications, Politecnico di Torino. He is the author or coauthor of more than 100 papers published in international journals and conference proceedings. His research interests are in the field of Electromagnetic Compatibility and Signal Integrity, where he works on the modeling and characterization of linear and nonlinear circuit elements. His recent activities include the behavioral modeling of digital ICs, transmission lines and PLC channels, the modeling and simulation of switching converters and the development of stochastic methods for the statistical simulation of circuits and systems with the inclusion of the effects of device or manufacturing uncertainties.

Flavio G. Canavero (SM'99-F'07) received his electronic engineering degree from Politecnico (Technical University) of Torino, Italy, and the PhD degree from the Georgia Institute of Technology, Atlanta, USA, in 1986. Currently he is a Professor of Circuit Theory with the Department of Electronics and Telecommunications, Politecnico di Torino, where he serves also as the Director of the Doctoral School. He is an IEEE Fellow. He has been the Editor-in-Chief of IEEE Transactions on Electromagnetic Compatibility, V.P. for Communication Services of the EMC Society and Chair of URSI Commission E. He received several Industry and IEEE Awards, including the prestigious Richard R. Stoddard Award for Outstanding Performance, which is the EMC Society?s highest technical award, and the Honored Member Award of EMC Society. His research interests include signal integrity and EMC design issues, interconnect modeling, black-box characterization of digital integrated circuits, EMI and statistics in EMC.