POLITECNICO DI TORINO
Repository ISTITUZIONALE

Multiple Delay Identification in Long Interconnects via LS-SVM Regression

Original
Multiple Delay Identification in Long Interconnects via LS-SVM Regression / Treviso, F., Trinchero, R., Canavero, F.G.. - In: IEEE ACCESS. - ISSN 2169-3536. - ELETTRONICO. - 9:(2021), pp. 39028-39042. [10.1109/ACCESS.2021.3063713]

Availability:
This version is available at: 11583/2874375 since: 2021-03-15T10:43:19Z

Publisher:
IEEE

Published
DOI:10.1109/ACCESS.2021.3063713

Terms of use:

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

Publisher copyright

(Article begins on next page)

13 October 2026

IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received February 1, 2021, accepted February 27, 2021. Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2021.3063713

Multiple Delay Identification in Long Interconnects via LS-SVM Regression

FELIPE TREVISO, (Student Member, IEEE), RICCARDO TRINCHERO, (Member, IEEE), AND FLAVIO G. CANAVERO, (Fellow, IEEE)
Department of Electronics and Telecommunication, Politecnico di Torino, 10129 Torino, Italy
Corresponding author: Felipe Treviso (felipe.treviso@polito.it)

ABSTRACT This work presents a novel approach for the accurate estimation of multiple time-delays from the frequency response of a distributed system. The proposed approach is based on a powerful and flexible machine learning technique, namely, the least-square support vector machine (LS-SVM). The LS-SVM regression is used to construct a metamodel of the transfer function describing a generic linear time-invariant system in a delayed-rational form. Specifically, after some manipulation the LS-SVM model precisely identifies the dominant propagation delays of the original system. The essential steps and critical criteria for the delay identification procedure are carefully discussed throughout the paper. Once the system delays have been identified, the rational part of the metamodel expansion is then obtained by means of a progressive application of the conventional vector fitting algorithm. Numerical examples are presented to illustrate the feasibility and performance of the proposed technique and to compare its performances with what is provided by state-of-the-art techniques. The results clearly highlight the capability of the proposed approach to identify the dominant delays in distributed systems, thus allowing to construct compact delayed rational models.

INDEX TERMS Interconnect modeling, delayed rational model, delay identification, machine learning, least-square support vector machine, kernel.

I. INTRODUCTION

Signal integrity is one of the limiting factors of the capacity of data transmission in a high-speed link. Indeed, due to propagation effects such as attenuation, ringing, signal delay, distortion, reflections and crosstalk, electrical interconnects are responsible for a considerable part of signal degradation in high-speed channels [1]. The above effects are particularly important for high-frequency signals, for which even physically short connections can behave as an electrically long channel. Therefore, during the design phase of a high-speed link, accurate models for the simulation of these structures are essential to predict signal integrity within a simulation framework.

A number of methods exist to generate accurate models when dealing with electrically short interconnects, or interconnects that can be classified as lossless or uniform transmission lines. Among such methods we can cite the use of equivalent lumped models, the method of characteristics and the use of reduced order and rational models [1]–[3]. Specifically, these techniques allow a mixed time- and frequency-domain modeling, since the effects of high-frequency interconnect models are better described in the frequency-domain, but the non-linear terminations of such interconnects (e.g., the link transceivers) can only be accounted for via time-domain simulations.

Contrarily, when the channel is electrically long and consists of the interconnection of several components, its characterization is best achieved through 2.5 or 3D electromagnetic simulations providing as result a set of frequency response curves characterizing the signal propagation between every port of the interconnect. Under the assumption of dealing with a linear structure, the standard approach to implement the model of the above structure within a circuit simulation environment is to approximate their transfer functions through a sum of rational functions represented in pole/residue format, which can be identified with the popular vector fitting (VF) algorithm [4]. However, when the propagation delay between ports of the system becomes large, its frequency response presents a phase that continuously changes and thus it requires a high number of poles to be represented in terms of a plain rational model. Also, if this model is used in a

The associate editor coordinating the review of this manuscript and approving it for publication was Wiren Becker.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

transient simulation, spurious oscillations may appear along the interconnect leading to an inaccurate evaluation of the signal integrity of the channel [5], [6].

A model that approximates accurately the interconnect structure described above is the *delayed rational model* (DRM) [7]. This model also allows the transfer function to be represented in terms of a simple equivalent circuit containing only ideal transmission lines and basic lumped elements (resistors, capacitors and inductors) [3], [7]–[10]. The DRM model, with an explicit representation of the delayed part, usually leads to a rational approximation that has a smaller number of poles than a VF rational model. This reduced complexity results in the possibility of building a more efficient model to be used in transient circuit simulations, thus reducing the computation time and required memory [7]–[9].

The DRM requires the estimation of three categories of parameters: poles and residues for its rational term, and the time-delays. The direct estimation of all those parameters at the same time is impractical, as suitable values are needed for the time-delays in the exponential term and for residues and poles in the numerator and denominator of the rational term. The identification of the dominant propagation delays from frequency response samples is an essential aspect for the achievement of a low-order DRM, since suitable approaches for the estimation of the rational part are available [7], [11]. Several of the methods available in the literature perform an optimization on the first identified set of delays in order to achieve the optimal model [9], [11]–[13]. If we are considering multiple delays, the optimization should be carried out in a high-dimensional space subject to local minima, making it a difficult task. Therefore, a precise estimation of the involved delays can provide an easier way to achieve a compact DRM.

The current literature presents some methods for the estimation of one or multiple time-delays [5]–[22]. For the case of a transfer function $H(j\omega)$ with a single time-delay τ, its value in the DRM can be directly optimized given that a reasonable search interval is defined [14]. If we consider a transmission line, this interval can come from the lossless time-delay in the propagation equation [12], [14] or it can be estimated via the magnitude of $H(j\omega)$, by considering that $H(j\omega)e^{j\omega\tau}$ is a minimum phase shift function, in which the phase angle can be calculated directly from the magnitude [13]–[15]. Another solid method applicable when identifying a single time-delay is by considering the causality and passivity of the system, which allows the application of the Hilbert transform to identify its time-delay [5], [6], [16].

On the other hand, the identification of multiple time-delays is more difficult. The Gabor transform, which provides a time-frequency decomposition of $H(j\omega)$, turns out to be the most used method [7], [10], [11], [17], [18]. The underlying idea is to analyze the energy content as a function of the delay, and use the time-delays that have the largest relative contribution to the total energy as dominant propagation delays. Another approach for the time-delay estimation through the time-frequency decomposition of the transfer function uses a wavelet transform performed on the time-domain impulse response of the system [8], [19]. An alternative method considers that the analysis of the time-domain response of the system to a narrow-band input pulse can be used as a way to estimate its corresponding propagation delays [9]. Although this method requires the time-domain response of the system, that response can be obtained through the inverse fast-Fourier transform of frequency-domain data. However, all these techniques estimate the multiple time-delays neglecting that the application of their result is on the estimation of a DRM.

Once the time-delays are identified, the rational parts of the DRM are easily estimated if the original transfer function can be split into the sum of delayed transfer functions. In that case, each part can be fitted individually after removing its individual delay, with any fitting method suitable for lumped structures: the vector fitting [17], the Loewner matrix framework [11], time-domain vector fitting [8] and Prony's method [19]. A previous determination of a minimum phase response of the separated transfer functions using the Hilbert's transform method can be applied before the rational approximation is performed [6], [18], in order to estimate the correct delay that should be used for the respective part of the transfer function.

Another robust method is the so called delayed vector fitting [7], [9], [10], a modified version of VF that uses delayed basis functions and through iterations identify a common set of poles for all the delays. This method provide excellent results when an accurate estimate for the dominant time-delays is provided. Otherwise, an optimization needs to be performed to improve the initial set of time-delay values [7].

In this work, an alternative scheme for the estimation of the dominant propagation delays within a DRM based on the least-squares support vector machine (LS-SVM) regression is presented. The LS-SVM regression is a flexible and powerful Machine Learning (ML) regression, which has been recently adopted for the uncertainty quantification in complex electronic systems [23]. Here, the LS-SVM regression with an ad-hoc kernel is adopted for the identification of multiple delays in a transmission line structure [22]. The proposed kernel has been built by considering an infinite dimensional feature space formed by an infinite number of rational basis, which accounts for a continuous set of delay values within a certain interval. Thanks to the above features, the proposed method is able to identify the dominant propagation delays of the original system. The obtained delays are then used, within an incremental procedure, for the construction of a compact DRM which is suitable for the simulation of long and complex electrical interconnects. The paper is organized as follows: the goal of this work and two alternative delay identification techniques are presented in Section II. Section III describes the basic concepts of the LS-SVM regression. Sections IV and V present the innovative way in which the time-delays for the construction of a DRM are obtained and utilized. Section VI shows the application of the method to some examples.

II. PROBLEM STATEMENT

Given a set of samples of the transfer function $H(j\omega_k) \in \mathbb{C}$ with $k = 1 \dots K$ of a long distributed structure at the angular frequencies $\omega_k \in \mathbb{R}$, we are looking for a DRM in the following form:

$$H(j\omega_k) \approx \tilde{H}(j\omega_k) = r_0 + \sum_{i=1}^{n_\tau} \tilde{H}_i(j\omega_k) e^{-j\omega_k \tau_i}, \tag{1}$$

with,

$$\tilde{H}_i(j\omega_k) = \sum_{j=1}^{n_{p,i}} \frac{r_{ij}}{j\omega_k - p_{ij}}. \tag{2}$$

where each of the n_τ terms $\tilde{H}_i(j\omega)$ corresponds to a rational model with $n_{p,i}$ poles $p_{ij} = p'_{ij} + jp''_{ij} \in \mathbb{C}$, residues $r_{ij} = r'_{ij} + jr''_{ij} \in \mathbb{C}$ and a constant term $r_0 \in \mathbb{R}$. Each of the rational terms is associated to its corresponding time-delay term $\tau_i \in \mathbb{R}^+$.

By assuming that each delay term is associated to the same set of poles, the DRM model $\tilde{H}(j\omega)$ of (1) can be simplified as follows[1]:

$$\tilde{H}(j\omega; \mathbf{r}, \mathbf{p}, \boldsymbol{\tau}) = r_0 + \sum_{i=1}^{n_\tau} \sum_{j=1}^{n_p} \frac{r_{ij}}{j\omega - p_j} e^{-j\omega\tau_i}. \tag{3}$$

The model parameters to be estimated by the regression problem (i.e., to fit the data) are: the residues $\mathbf{r} = [r_0, r_{11}, \dots, r_{n_\tau n_p}]$, the poles $\mathbf{p} = [p_1, \dots, p_{n_p}]$ and the delays $\boldsymbol{\tau} = [\tau_1, \dots, \tau_{n_\tau}]$, for an overall number of $n_p n_\tau + n_p + n_\tau + 1$ unknowns.

The DRM is linear with respect to the residues $\mathbf{r}$, but it is nonlinear with respect to the poles $\mathbf{p}$ and the delays $\boldsymbol{\tau}$. Unfortunately, it is unfeasible to estimate all the parameters (i.e., the poles, the delays and the residues) together. A wise strategy is to rely on a two-step identification algorithm, where the delays are identified independently and afterwards the poles are estimated by considering such delays. Naturally, once the delays τ_i and the poles p_j of the models are known, the residues can be easily estimated by solving a simple linear regression. However, how can we find an accurate estimation of the delays knowing only the samples $\tilde{H}(j\omega_k)$? Two available schemes are briefly described below, based on the Hilbert transform and the Gabor transform, respectively.

A. HILBERT TRANSFORM METHOD

This procedure finds the dominant propagation delay by decomposing the transfer function as the product of a minimum-phase component $H_{min}(j\omega)$ and an all-pass function that accounts for the propagation delay τ:

$$H(j\omega) = H_{min}(j\omega) e^{-j\omega\tau}. \tag{4}$$

This decomposition is possible because, for the specific case of minimum-phase systems, the Hilbert Transform [24] (or equivalently, the Kramers-Kronig relations) establishes an unique relation between the magnitude and phase such that:

$$\arg\left(H_{min}(j\omega)\right) = -\frac{1}{2\pi} \fint_{-\pi}^{\pi} \log|H_{min}(j\theta)| \cot\left(\frac{\omega - \theta}{2}\right) d\theta, \tag{5}$$

where the barred integral symbol $\fint$ represents the Cauchy principal value of the corresponding integral. From (4), it is evident that the magnitude of $H_{min}(j\omega)$ can be easily determined as:

$$|H_{min}(j\omega)| = |H(j\omega)|. \tag{6}$$

The relation in (5) can therefore be applied to estimate the phase of $H_{min}(j\omega)$. After obtaining the complex value of $H_{min}(j\omega)$, the delay τ is estimated as [5], [6]

$$\tau = -\frac{\arg\left(e^{-j\omega\tau}\right)}{\omega} = -\frac{\arg\left(H(j\omega)/H_{min}(j\omega)\right)}{\omega}, \tag{7}$$

computed as the average over all available frequency points. The Hilbert transform method has good performance for systems with a single delay in the frequency response [6]. When multiple delays are present, the system delays can be identified by using the Gabor transform.

B. GABOR TRANSFORM METHOD

The normalized frequency domain Gabor transform is defined as follows [17]:

$$\mathcal{G}(\omega, \tau) = \int_{-\infty}^{+\infty} H(\xi) W^*_{\omega,\tau}(\xi) d\xi, \tag{8}$$

where

$$W_{\omega,\tau}(\xi) = W(\xi - \omega) e^{-j\xi\tau} \tag{9}$$

is a frequency-shifted window function modulated by the complex exponential $e^{-j\xi\tau}$, with the number of oscillations proportional to τ and * represents the complex-conjugation operation. The Gabor transform uses a normalized Gaussian window [25], defined as:

$$W(\xi) = (a^2\pi)^{-1/4} e^{-\frac{\xi^2}{2a^2}}, \tag{10}$$

such that $||W||_2 = 1$.

The shape of the window $W(\xi)$ depends on the parameter a. Such parameter provides a trade-off between the time and frequency domain resolution of the transform. A small value for a produces a narrow window, for which the resolution of transform $\mathcal{G}(\omega, \tau)$ is high in frequency-domain (i.e., for the variable ω) and poor in time-domain (i.e., for the variable τ). Contrarily, with a large a, the transform resolution is poor in frequency-domain and high in time-domain.

The spectrogram $|\mathcal{G}(\omega, \tau)|^2$ represents the time-frequency energy decomposition of $H(\xi)$ [17]. The propagation delays τ_i can be obtained from this by averaging the energy decomposition over ω via:

$$G(\tau) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} |\mathcal{G}(\omega, \tau)|^2 d\omega. \tag{11}$$

[1] Such simplified scenario will be used only to estimate the dominant delays of the system. The final DRM will be constructed by using the generic formulation in (1).

Once computed, the maxima of $G(\tau)$ provide a good estimate for the system delays τ_i [7], [17]. If $G(\tau)$ has multiple local maxima, this method is able to approximate multiple time-delays, thus overcoming the limitations of the Hilbert transform method. However, the time-frequency trade-off given by the window width still exists, and so the necessity of tuning such width.

III. A LEAST-SQUARE SUPPORT VECTOR MACHINE PRIMER

This section summarizes the basic concepts of the LS-SVM regression for complex-valued functions. The LS-SVM is a flexible machine learning technique for both classification and regression [26], [27]. Usually, the LS-SVM regression is derived for real valued inputs $\mathbf{x}$ and output y. However, for this work, the inputs and output are required to be complex, since we will deal with transfer functions in the frequency domain, as specified in (1). The modification of the LS-SVM problem to consider complex input-output relationship is straightforward [28], [29] and outlined in the Appendix.

Starting from a set of complex training pairs $\{(\mathbf{x}_k, y_k)\}_{k=1}^{K}$ with the scalar output $y_k \in \mathbb{C}$ and possibly multiple input parameters arranged in a vector $\mathbf{x}_k \in \mathbb{C}^d$, we are looking for the following *primal space* formulation of the LS-SVM regression [30], which writes:

$$\tilde{y}(\mathbf{x}) = \langle \mathbf{w}^*, \boldsymbol{\varphi}(\mathbf{x}) \rangle + b, \tag{12}$$

where $\mathbf{w} \in \mathbb{C}^D$ are the regression coefficients, $\boldsymbol{\varphi}(\cdot) : \mathbb{C}^d \to \mathbb{C}^D$ are the considered basis functions and $b \in \mathbb{C}$ is a bias term. Equivalently, the model in (12) can also be represented by its *dual space* formulation [30], which writes:

$$\tilde{y}(\mathbf{x}) = \sum_{k=1}^{K} \alpha_k \mathrm{k}(\mathbf{x}_k, \mathbf{x}) + b, \tag{13}$$

where $\alpha_k = \alpha_{k,r} + j\alpha_{k,i}$ are the model coefficients of this representation and the kernel function $\mathrm{k}(\mathbf{x}_i, \mathbf{x}_j)$ is defined as:

$$\mathrm{k}(\mathbf{x}_i, \mathbf{x}_j) = \langle \boldsymbol{\varphi}(\mathbf{x}_i), \boldsymbol{\varphi}(\mathbf{x}_j) \rangle. \tag{14}$$

The vector of model error $\boldsymbol{e} = \{e_k\}_{k=1}^{K}$ (see (43) in the Appendix) is equal to the true value y_k minus the model output of (12) computed at $\mathbf{x}_k$, for all training points. The LS-SVM minimizes the L2-norm of the primal space model coefficients $\mathbf{w}$ plus the sum of squared errors, weighted by the regularization parameter γ. Such parameter, commonly referred to as hyperparameter, is tuned during the model training, usually via cross-validation. A large value for γ will lead to a model with smaller error, but that may present overfitting.

The LS-SVM model is determined in the dual space by solving the following linear system of equations:

$$\begin{bmatrix} \boldsymbol{\Omega} + \mathbf{I}^{2K}/\gamma & \mathbf{1} & \mathbf{0} \\ & \mathbf{0} & \mathbf{1} \\ \mathbf{1}^T \quad \mathbf{0}^T & 0 & 0 \\ \mathbf{0}^T \quad \mathbf{1}^T & 0 & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{\alpha}_r \\ \boldsymbol{\alpha}_i \\ b_r \\ b_i \end{bmatrix} = \begin{bmatrix} \mathbf{y}_r \\ \mathbf{y}_i \\ 0 \\ 0 \end{bmatrix}, \tag{15}$$

where $\mathbf{1} = [1, \ldots, 1]^T$ and $\mathbf{0} = [0, \ldots, 0]^T$ are vectors containing K equal elements, $\mathbf{I}^{2K}$ is an identity matrix with size $2K$, and the $2K \times 2K$ kernel matrix $\boldsymbol{\Omega}(\mathbf{x}_k, \mathbf{x}_k)$ is defined as

$$\boldsymbol{\Omega} = \begin{bmatrix} \boldsymbol{\Omega}^{(1,1)} & \boldsymbol{\Omega}^{(1,2)} \\ \boldsymbol{\Omega}^{(2,1)} & \boldsymbol{\Omega}^{(2,2)} \end{bmatrix}. \tag{16}$$

In the $K \times K$ square submatrices that compose $\boldsymbol{\Omega}$, an element $\Omega_{j,i}^{(\cdot,\cdot)}$ in the j-th row and i-th column is defined as follows:

$$\Omega_{j,i}^{(1,1)} = \Re\{\mathrm{k}(\mathbf{x}_i, \mathbf{x}_j)\}, \tag{17a}$$
$$\Omega_{j,i}^{(1,2)} = -\Im\{\mathrm{k}(\mathbf{x}_i, \mathbf{x}_j)\}, \tag{17b}$$
$$\Omega_{j,i}^{(2,1)} = \Im\{\mathrm{k}(\mathbf{x}_i, \mathbf{x}_j)\}, \tag{17c}$$
$$\Omega_{j,i}^{(2,2)} = \Re\{\mathrm{k}(\mathbf{x}_i, \mathbf{x}_j)\}, \tag{17d}$$

for $i, j = 1, \ldots, K$.

The unknowns $\boldsymbol{\alpha} = \boldsymbol{\alpha}_r + j\boldsymbol{\alpha}_i = [\alpha_{1,r} + j\alpha_{1,i}, \ldots, \alpha_{K,r} + j\alpha_{K,i}]$ and $b = b_r + jb_i$ are computed by solving (15). Once the dual space model is determined for known basis, its equivalent primal space formulation can be obtained through

$$\mathbf{w} = \sum_{k=1}^{K} \alpha_k \boldsymbol{\varphi}^*(\mathbf{x}_k). \tag{18}$$

The dimensions of primal and dual formulations are different: the primal space formulation has D terms, which is the number of basis of the primal space, i.e., the dimension of $\mathbf{w}$ and $\boldsymbol{\varphi}(\mathbf{x})$, while the dual space formulation is a non-parametric model with a number of terms equal to the number of training samples, K. Finally, the constant bias term b is present in both models.

IV. DELAY EXTRACTION VIA LS-SVM REGRESSION

This Section deals with the application of the LS-SVM regression presented in Sec. III for the modeling of complex distributed structures. For the sake of clarity, the proposed methodology is first derived step-by-step to define the LS-SVM formulation in the case of delayed-rational transfer functions introduced in Sec. II. The practical application for the identification of delays and their use in a transfer function such as the one in (1) is provided in the next section.

The discussion starts by recasting the DRM in (3) in terms of the primal-space LS-SVM regression in (12) which writes:

$$\tilde{H}(j\omega) = \sum_{i=1,j=1}^{n_p,n_\tau} \mathrm{w}_{ij}\varphi_{ij}(\omega; p_j, \tau_i) + b \tag{19}$$
$$= \langle \mathbf{w}^*, \boldsymbol{\varphi}(\omega; \mathbf{p}, \boldsymbol{\tau}) \rangle + b, \tag{20}$$

where $\mathbf{w} = [\mathrm{w}_{11}, \ldots, \mathrm{w}_{n_\tau n_p}]^T \in \mathbb{C}^{(n_\tau n_p)}$ is a vector collecting the regression unknowns, the generic parameter $\mathbf{x}$ in (12) has been replaced by ω (i.e., $d = 1$) and $\boldsymbol{\varphi}(\omega; \mathbf{p}, \boldsymbol{\tau}) \in \mathbb{C}^{(n_\tau n_p)}$ is a vector collecting the basis functions

$$\boldsymbol{\varphi}(\omega; \mathbf{p}, \boldsymbol{\tau}) = \left[\varphi_{11}(\omega; p_1, \tau_1), \ldots, \varphi_{n_\tau n_p}(\omega; p_{n_p}, \tau_{n_\tau})\right]^T. \tag{21}$$

This formulation can be seen as a discretization in the p-τ space from which each node $\{\tau_i, p_j\}$ leads to a basis of the DRM, and in which all poles are paired with the same combination of delays.

Comparing (19) and (3), the basis functions $\varphi_{ij}(\omega; p_j, \tau_i)$ are defined by the following expression

$$\varphi_{ij}(\omega; p_j, \tau_i) = \frac{c_{ij}}{j\omega - p_j} e^{-j\omega\tau_i} \tag{22}$$

$$= c_{\tau_i} e^{-j\omega\tau_i} \cdot \frac{c_{p_j}}{j\omega - p_j} \tag{23}$$

$$= \varphi_i^{\tau}(\omega; \tau_i) \cdot \varphi_j^{p}(\omega; p_j), \tag{24}$$

where $c_{ij} = c_{\tau_i} c_{p_j}$ is a constant introduced to increase the model flexibility. The overall number of features (or bases) is $D = n_p n_\tau$, which is equivalent to the number of nodes in the aforementioned discretization.

From the latter development, it is evident that the regression unknowns w_{ij} are proportional to the corresponding residue r_{ij} by the scalar constant $c_{ij} \in \mathbb{R}$:

$$r_{ij} = c_{ij} \mathrm{w}_{ij}. \tag{25}$$

It is also clear that the bias term b is equal to the constant term r_0 of the DRM. Since the LS-SVM formulation is equivalent to the original one in (3), using a finer discretization will increase the dimensions of the feature space provided by the transformation function $\boldsymbol{\varphi}$ (i.e., the number of bases φ_{ij}).

Aside from the above expansion, a parallel dual space formulation for the LS-SVM problem, defined by (13), takes the form

$$\tilde{H}(j\omega) = \sum_{k=1}^{K} \alpha_k \mathrm{k}(\omega_k, \omega; \mathbf{p}, \boldsymbol{\tau}) + b, \tag{26}$$

where the kernel $\mathrm{k}(\cdot, \cdot) : \mathbb{C} \times \mathbb{C} \to \mathbb{C}$ is defined according to the basis functions collected in $\boldsymbol{\varphi}$ as:

$$\begin{aligned}
&\mathrm{k}(\omega_k, \omega_l; \mathbf{p}, \boldsymbol{\tau}) \\
&= \langle \boldsymbol{\varphi}(\omega_k; \mathbf{p}, \boldsymbol{\tau}), \boldsymbol{\varphi}(\omega_l; \mathbf{p}, \boldsymbol{\tau}) \rangle \\
&= \sum_{i=1}^{n_\tau} \sum_{j=1}^{n_p} \frac{c_{ij} e^{j\omega_k \tau_i}}{(-j\omega_k - p_j^*)} \frac{c_{ij} e^{-j\omega_l \tau_i}}{(j\omega_l - p_j)} \\
&= \left\langle \boldsymbol{\varphi}^{\tau}(\omega_k; \boldsymbol{\tau}), \boldsymbol{\varphi}^{\tau}(\omega_l; \boldsymbol{\tau}) \right\rangle \left\langle \boldsymbol{\varphi}^{p}(\omega_k; \mathbf{p}), \boldsymbol{\varphi}^{p}(\omega_l; \mathbf{p}) \right\rangle \\
&= \underbrace{\sum_{i=1}^{n_\tau} c_{\tau_i}^2 e^{-j(\omega_l - \omega_k)\tau_i}}_{\mathrm{k}_\tau(\omega_k, \omega_l; \boldsymbol{\tau})} \underbrace{\sum_{j=1}^{n_p} \frac{c_{p_j}^2}{(-j\omega_k - p_j^*)(j\omega_l - p_j)}}_{\mathrm{k}_\mathrm{p}(\omega_k, \omega_l; \mathbf{p})}.
\end{aligned} \tag{27}$$

where p_j and τ_i span all the poles and delays values considered in their discretization. The resulting kernel can be interpreted as the product of two kernels, one generated by the chosen delays and the other by the chosen poles. Such two terms, $\mathrm{k}_\tau(\omega_k, \omega_l; \boldsymbol{\tau})$ and $\mathrm{k}_p(\omega_k, \omega_l; \mathbf{p})$, are then named the delay and the rational kernels, respectively.

Equation (26) is a non-parametric model (i.e., the number of the regression unknowns is independent from the number of basis functions in $\boldsymbol{\varphi}$) and, no matter how finely the $p - \tau$ plane is discretized, it can always be determined by means of the solution of the linear system in (15). Indeed, the discretization is embedded within the kernel function. The key advantage of working in the dual space is that the dual space formulation does not require an explicit definition of the basis functions $\boldsymbol{\varphi}$ and its formulation can even be applied to the extreme case in which the dimension of $\boldsymbol{\varphi}$ grows to infinity.

According to this fact, let us consider the case in which the $\tau-$axis is infinitely discretized and any τ value between τ_m and τ_M is included in the kernel. This discretization is achieved by making $c_{\tau_i}^2 = c_\tau^2 = \Delta_\tau = (\tau_M - \tau_m)/n_\tau$ while $n_\tau \to \infty$. Hence, the inner product $\langle \boldsymbol{\varphi}(\omega_k; \mathbf{p}, \boldsymbol{\tau}), \boldsymbol{\varphi}(\omega_l; \mathbf{p}, \boldsymbol{\tau}) \rangle$ is computed through an integral over τ and a sum over $\mathbf{p}$. With reference to (27), the rational kernel remains the same, but the delay kernel becomes the following definite integral [31]:

$$\mathrm{k}_\tau(\omega_k, \omega_l) = \int_{\tau_m}^{\tau_M} e^{-j(\omega_l - \omega_k)\tau} d\tau. \tag{28}$$

The above delay kernel represents a space with an infinite dimension due to the continuous $\tau-$axis. The discretization along the $p-$axis is given by the other part of the kernel, $\mathrm{k}_p(\omega_k, \omega_l)$.

The integral in (28) can be evaluated to a closed expression, which writes:

$$\begin{aligned}
&\mathrm{k}_\tau(\omega_k, \omega_l) \\
&= \begin{cases} \dfrac{\left(e^{-j\tau_M(\omega_l - \omega_k)} - e^{-j\tau_m(\omega_l - \omega_k)}\right)}{-j(\omega_l - \omega_k)}, & \omega_k - \omega_l \neq 0; \\ \tau_M - \tau_m, & \omega_k - \omega_l = 0. \end{cases}
\end{aligned} \tag{29}$$

The rational kernel in (27) remains a sum over a limited set of poles $\mathcal{P} = \{p_1, \ldots, p_{n_p}\}$. This set of poles will provide the general trend that must be followed by the rational part of the model, and thus do not need to contain the exact poles of the system, while the infinite number of delays will guarantee that its exact delays are replicated in the LS-SVM model. An approach including an infinite number of poles as done for the delay kernel could also be possible. However, the resulting double integration leads to complicated equations, which makes the subsequent evaluation of the kernel slow, limiting the still undetermined benefits that it could bring.

The constants c_{p_j} in (23) are set to $c_{p_j} = |p_j'|^{1/2}$. In this way, all the terms $\varphi_j^p(\omega; p_j)$ have the same 2-norm for any value of p_j for which $p_j' \neq 0$. The 2-norm of each of the $\varphi_j^p(\omega; p_j)$ basis amounts to:

$$\left\langle \varphi_j^p(\omega; p_j), \varphi_j^p(\omega; p_j) \right\rangle = \int_{-\infty}^{\infty} \frac{|p_j'|}{(j\omega - p_j)(-j\omega - p_j^*)} d\omega = \pi. \tag{30}$$

This definition of c_{p_j} amplifies the LS-SVM weights w_{ij} associated with the dominant poles of the system (i.e., the poles with a smaller real part). Considering the same residues in the final model, a smaller value for c_{ij} means that the weight w_{ij} should increase, as can be seen in (25).

Separating the rational part of (27), and with the considerations stated above, the rational kernel $\mathrm{k}_p(\omega_k, \omega_l)$ writes:

$$\mathrm{k}_p(\omega_k, \omega_l) = \sum_{j=1}^{n_p} \frac{|p'_j|}{(j\omega_k - p_j)\left(-j\omega_l - p_j^*\right)}. \tag{31}$$

According to (27), the delay-rational kernel results from the multiplication of the delay kernel in (29) by the rational kernel in (31). It contains the delayed rational basis of (22) with the set of poles $\mathcal{P} = \{p_1, \ldots, p_{n_p}\}$, each of them containing the same infinite number of delays in the range $[\tau_m, \tau_M]$.

While the dual space LS-SVM model (26) with the aforementioned kernel is non-parametric, it can still be linked to an infinite dimensional parametric model expressed by (20):

$$\begin{aligned}\tilde{H}(j\omega) &= \left\langle \mathbf{w}^*, \boldsymbol{\varphi}(\omega; \mathbf{p}, \boldsymbol{\tau}) \right\rangle + b \\ &= \sum_{j=1}^{n_p} \int_{\tau_m}^{\tau_M} \mathrm{w}'_j(\tau) e^{-j\omega\tau} \frac{c_{p_j}}{j\omega - p_j} d\tau + b,\end{aligned} \tag{32}$$

where the weights $\mathrm{w}'_j(\tau)$ refer to the continuous parameter τ. They exist for any pole p_j and take the following form, according to (18):

$$\mathrm{w}'_j(\tau) = \sum_{k=1}^{K} \alpha_k e^{j\omega_k\tau} \frac{c_{p_j}}{-j\omega_k - p_j^*}. \tag{33}$$

These weights $\mathrm{w}'_j(\tau)$ are proportional to the residue in the DRM, thus providing information on the most relevant terms for the model construction; in fact, a large residue is associated to the delay that is most significant for the part of the model corresponding to the pole p_j. Furthermore, a combined weight $W(\tau)$ that accounts for all poles of the kernel can be defined as:

$$\begin{aligned}W(\tau) &= \left(\sum_{j=1}^{n_p} |\mathrm{w}'_j(\tau)|^2 \right)^{1/2} \\ &= \left(\sum_{j=1}^{n_p} \left(\sum_{k=1}^{K} \frac{\alpha_k c_p}{-j\omega_k - p_j^*} e^{j\omega_k\tau} \right) \right. \\ &\quad \left. \times \left(\sum_{k=1}^{K} \frac{\alpha_k c_p}{-j\omega_k - p_j^*} e^{j\omega_k\tau} \right)^* \right)^{1/2}.\end{aligned} \tag{34}$$

The plot and analysis of the weight $W(\tau)$ provides information about the system delays, with peaks indicating higher "energy" content and corresponding to the dominant delays of the original system. The representativeness of these peaks to identify the system delays is assured by the fact that one of the conditions of the LS-SVM problem is the minimization of $\langle \mathbf{w}, \mathbf{w} \rangle = \sum |\mathrm{w}|^2$ (see (43) in Appendix). Hence, it is unlikely the occurrence of unnecessary high values for $\mathrm{w}'_j(\tau)$.

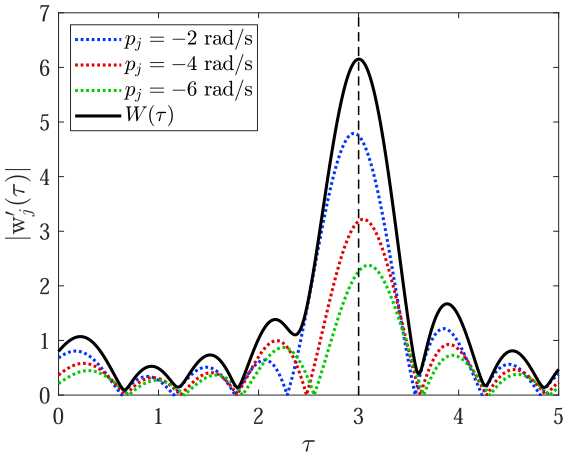


FIGURE 1. Magnitude of the LS-SVM weights as a function of τ.

A. ILLUSTRATIVE EXAMPLE

As an illustrative example, the following synthetic transfer function with a single real pole at -3 rad/s and a single delay at 3 s, is considered:

$$H(j\omega) = \frac{1}{j\omega + 3} e^{-j\omega 3}, \tag{35}$$

in the bandwidth 0 - 5 rad/s.

We can build the LS-SVM model from (26) with $\tau_m = 0$, $\tau_M = 5$ s and the poles $\mathcal{P} = \{-2, -4, -6\}$ rad/s. It is important to remark that the above set of poles does not match the exact pole in the transfer function, but are within the same region of space. The plot of the magnitude of $\mathrm{w}'_j(\tau)$ as a function of τ gives us an interesting insight. Figure 1 shows these curves for each of the poles included in the kernel and with $c_{p_j} = 1$. The curves clearly have their peak near the true delay of the original system, $\tau = 3$ s. However, if the weights are observed for one individual pole, the peak is not exactly at the expected value: in fact, it is at $\tau = 3.09$ s for $p_j = -6$ rad/s, $\tau = 3.05$ s for $p_j = -4$ rad/s and $\tau = 2.95$ s for $p_j = -2$ rad/s. Nonetheless, by looking at the combined weight $W(\tau)$, the peak appears at $\tau = 3.00$ s, showing that the use of a higher number of poles compensate the fact that the exact pole of the system is unknown. The delay is shown to be the most important element for obtaining an accurate model, and the use of infinite possible delays by means of the kernel function guarantees that its correct value is considered. In the next section, this procedure for delay identification is described for a more generic case, when the original system is unknown.

V. PRACTICAL PROCEDURE FOR DRM ESTIMATION

This Section generalizes the procedure illustrated in the previous Section to the case of a system with multiple delays and reflections like, for example, an electric interconnect. The target model to approximate this system is the DRM in (1), with *complex* and *unknown poles*.

A. TIME-DELAY INTERVALS IN THE DELAY KERNEL

The delay extraction approach outlined in Sec. IV and better detailed in this section requires a delay-rational kernel

function containing all delays within a certain interval. However, the span of this interval was not discussed yet. For causal systems, the time-delay has to be positive, and therefore, $\tau_m \geq 0$. The determination of the maximum delay τ_M can rely on the frequency discretization of input data. In fact, assuming that the original frequency response of the system has equally-spaced frequency points, with Δ_f being the separation between two consecutive samples, the following empirical relationship holds: $\tau_M - \tau_m < 1/\Delta_f$. With $\tau_m = 0$, the maximum time-delay then becomes $\tau_M < 1/\Delta_f$, which is the value we use for our examples presented in section VI.

B. THE RATIONAL KERNEL

The example in Sec. IV-A showed that the delays can be identified without knowing the exact poles that compose the rational part of the system. A sufficiently large number of poles, chosen in a reasonable region of space where the poles of the original system are expected to be, was able to replace the effect of the exact pole in the estimation of the LS-SVM model. Hence, we use the set $\mathcal{P} = \{p_1, \ldots, p_{n_p}\}$ in which the poles are randomly chosen with independent real and imaginary parts, both from a Gaussian distribution with mean zero and standard deviation equal to ω_K. The real part of the poles is forced to be negative by discarding samples with positive value, in order to ensure that the obtained system is stable. Although this is not the optimal set of poles for the rational kernel, this choice is adopted because it provides a good delay estimation using a relatively small number of poles. Since the rational part of the system (with its corresponding delay removed) should be very simple, usually no more than 10 poles, a n_p on the order of 20 to 30 poles is sufficient to identify the delay values.

C. DELAY IDENTIFICATION ALGORITHM

After considering the details of the previous subsections, we can devise a procedure to identify the relevant delays of a generic distributed system based on the weights estimated by a LS-SVM model. The procedure is simply an extension of what was described at the end of Sec. IV and illustrated by the black line of Fig. 1. The complete procedure for time-delay identification can be summarized by Algorithm 1. Once identified by the peak positions, the delays can be used to build a DRM of little complexity.

Admittedly, the peaks still need to be found visually, and some discernment is required to distinguish the delay peaks from the side lobes that appear in the plot, as observed in Fig. 1. Nonetheless, the main delay peak is more pronounced than its corresponding side lobes, and hence the delay peaks are identified as being higher than its near surroundings.

D. DELAYED RATIONAL FITTING

It is presented here a procedure to progressively identify a DRM of (1) starting from the delay terms obtained via the procedure proposed in V-C. The DRM model $\check{H}$ is built in n_τ steps, being n_τ the number of identified delays, by considering a different set of poles for each delay term, i.e.,

$$H(j\omega_k) \approx \check{H}(j\omega_k) = r_0 + \sum_{i=1}^{n_\tau} \check{H}_i(j\omega_k) e^{-j\omega_k \tau_i}, \qquad (36)$$

for $k = 1, \ldots, K$, where, similar to (1), $\check{H}_i$ indicates the approximated rational terms associated to the delay τ_i.

The proposed procedure progressively identified the terms $\check{H}_i$ of the DRM in (36) by adding a new rational term at each iteration. Specifically, at a generic step l of the procedure, a standard rational fitting algorithm [4] is applied to get an approximated version $\check{H}_l(\omega_k)$ of the rational term associated to the l-th delay τ_l.

The approximation of each rational term $\check{H}_l(j\omega_k)$ is computed as the difference between the target transfer function H and the DRM approximation at the current iteration $\check{H}^{(l)}$, compensated for the time-delay associated to the current step $\check{H}^{(l)}$, and writes:

$$\check{H}_l(j\omega_k) = \left[H(j\omega_k) - \check{H}^{(l-1)}(j\omega_k) \right] e^{j\omega_k \tau_l} \qquad (37)$$

for $l = 1, \ldots, n_\tau$.

Whilst, the overall DRM approximation $\check{H}^{(l)}$ at a generic iteration l writes

$$\check{H}^{(l)}(j\omega_k) = \sum_{i=1}^{l} \check{H}_i(j\omega_k) \exp(-j\omega_k \tau_i). \qquad (38)$$

At the first step, the procedure is initialized as $\check{H}^{(0)}(j\omega_k) = 0$. After n_τ steps, the algorithm ends and provides as a results the DRM $\check{H}(j\omega_k) = \check{H}^{(n_\tau)}(j\omega_k)$.

The rational fitting adopted at each iteration considers a different number of poles for each of rational term $\check{H}_i$ of the DRM. For the rational term, the number of poles is selected by looking at the marginal improvement of the model error (i.e., we check if the model accuracy increases significantly when the number of poles is increased by one). The above procedure is summarized in Alg. 2.

The procedure does not lead to the optimal selection of the poles and residues for the DRM, but it is simple and effective

Algorithm 1 Delay Identification Algorithm

1: Group the frequency response samples in the vectors $\boldsymbol{\omega} = [\omega_1, \ldots, \omega_K]$ and $H(j\boldsymbol{\omega}) = H_r(j\boldsymbol{\omega}) + jH_i(j\boldsymbol{\omega}) = [H_r(j\omega_1) + jH_i(j\omega_1), \ldots, H_r(j\omega_K) + jH_i(j\omega_K)]$;
2: Draw a random set of complex poles $\mathcal{P} \in \mathbb{C}^{n_p}$ in a reasonable region of the complex plane as discussed in Sec. V-B;
3: Define the minimum and maximum considered delays τ_m and τ_M;
4: Compute the kernel for all frequency pairs $\{\omega_i, \omega_j\}_{i,j=1}^{K}$ using (27), and assembly the linear system in (15);
5: Tune the hyperparameter γ such that the model error is acceptable and solve (15);
6: Compute $W(\tau)$ according to (34);
7: Find the peaks in $W(\tau)$, which are the set of relevant delays $\boldsymbol{\tau} = \{\tau_1, \ldots, \tau_{n_\tau}\}$ for the modeled system.

in providing a DRM with a small number of poles, which are different for each delay term. A more robust and automated method for a delayed rational fitting using the same poles for all delay terms is provided by [7]. The passivity of the resulting model is not guaranteed by the method. However, it can be checked and enforced afterwards by using methods available in the literature [9].

Algorithm 2 Incremental Rational Fitting Algorithm

1: Group the frequency response samples in the vectors $\boldsymbol{\omega} = [\omega_1, \ldots, \omega_K]$ and initialize the algorithm with the time-delay vector $\boldsymbol{\tau} = [\tau_1, \ldots, \tau_{n_\tau}]$, the target transfer function $H(j\boldsymbol{\omega})$ and the approximate model $\tilde{H}^{(0)}(j\boldsymbol{\omega}) = 0$;
2: **for** $l = 1, \ldots, n_\tau$ **do**
3: Compute $\check{H}_l(j\boldsymbol{\omega})$ by using (37);
4: Use VF to get a rational model of $\check{H}_l(j\boldsymbol{\omega})$ with a maximum number of poles from 1 to a small $n_{p,\max}$;
5: Select the appropriate number of poles as explained in V-D;
6: Use (38) to update the partial DRM $\check{H}^{(l)}(j\boldsymbol{\omega})$;
7: **end for**
8: $\check{H}(j\boldsymbol{\omega}) = \check{H}^{(n_\tau)}(j\boldsymbol{\omega})$ is the final model containing all delays.

VI. APPLICATION EXAMPLES

The main purpose of this work was to devise a procedure to extract the delays and poles of a distributed system that can be applied to extract a DRM. In order to exemplify the use of the advocated procedure, three examples are presented in this section. Throughout the examples, comparisons with the state-of-the-art VF technique for the estimation of rational models [4], Hilbert transform [5] approach and Gabor transform [7] method for delay identification are presented.

A. EXAMPLE 1: SYNTHETIC TRANSFER FUNCTION

The proposed model is first investigated on a synthetic example with known poles and delays. Initially, the following transfer function sampled at 1001 evenly spaced points from $\omega = 0$ to $\omega = 2000\pi$ rad/s is considered:

$$H_1(j\omega) = \frac{20\pi}{j\omega + 20\pi} e^{-0.15j\omega}. \tag{39}$$

For the delay identification, we adopted a kernel with the 20 random poles shown in Fig. 2-a, and a maximum delay value $\tau_M = 1$ s determined according to subsection V-A. Then, Algorithm 1 produces the weight $W(\tau)$, as displayed in Fig. 2-b with normalization to its maximum value. The peak identifies exactly the delay of our synthetic transfer function, i.e. $\tau = 0.15$ s. The figure also compares the results of the present method with the time-delays obtained by means of two other methods: the Hilbert transform (gray line) and the Gabor transform (dashed lines). The former method identifies by design only one delay value, at $\tau = 0.1494$ s, which is slightly to the left of the actual delay value. On the other hand, the Gabor transform, after the tuning of the

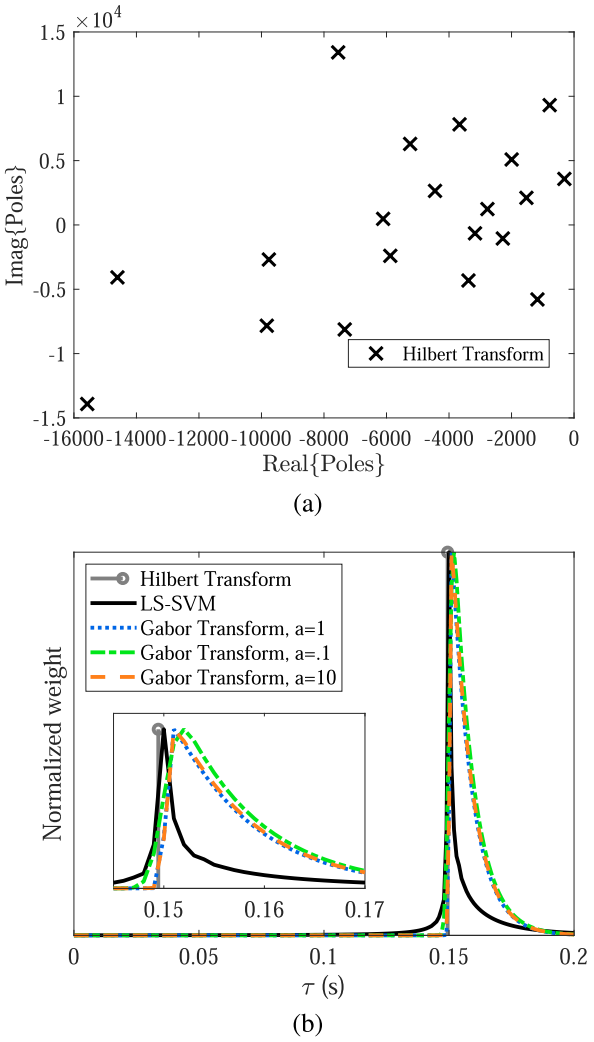


FIGURE 2. First synthetic example: Panel (a) shows the 20 random poles used within the LS-SVM kernel. Panel (b) plots $W(\tau)$ for delay identification; comparison with Hilbert and Gabor transforms is also displayed.

window width provides a pattern which is similar to the one provided by the proposed method. However, the peak in the Gabor transform plot is shifted slightly to the right of the LS-SVM peak, and it can be pushed further if smaller values for the window width a are chosen.

As a further comparison among the three methods, and to stress their performances for a system with multiple delays, let us consider a second synthetic transfer function:

$$H_2(j\omega) = \left(\frac{20\sqrt{10}\pi}{j\omega + 20\pi + 60\pi j} + \frac{20\sqrt{10}\pi}{j\omega + 20\pi - 60\pi j} \right) e^{-0.15j\omega} + \frac{4\pi}{j\omega + 20\pi} e^{-0.3j\omega} \tag{40}$$

sampled at the same frequency points as $H_1(j\omega)$. There are two main differences from the previous example: $H_2(j\omega)$ is composed by terms with two different delays $\tau = \{0.15, 0.3\}$ s, and one of this terms is an underdamped system.

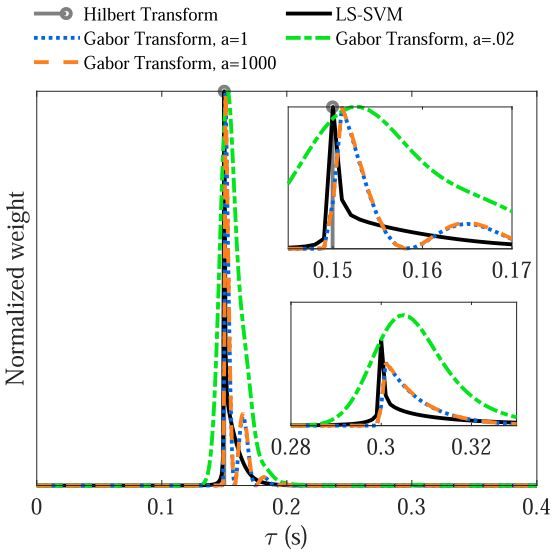


FIGURE 3. Plots of the normalized weight used for delay identification; comparison with Hilbert and Gabor transforms is also displayed, for the second synthetic example.

TABLE 1. Comparison of the computational time between the three applied delay identification methods for different number of frequency response samples, for the two examples of synthetic transfer functions.

Delay identification method:		LS-SVM	Gabor transform	Hilbert transform
$H_1(j\omega)$:	1001 samples	17.57	12.96	276.47
Computational	501 samples	7.00	6.47	113.06
time [s]	334 samples	4.15	5.24	71.62
$H_2(j\omega)$	1001 samples	16.45	9.88	220.83
Computational	501 samples	7.35	7.52	110.94
time [s]	334 samples	4.50	5.65	82.65

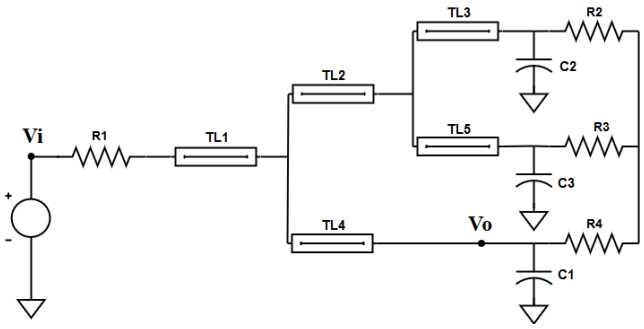


FIGURE 4. Schematic of the circuit used in example 2.

The normalized $W(\tau)$ is displayed in Fig. 3, together with the Hilbert and Gabor transform curves. The main LS-SVM weight peak identifies exactly the delays of our synthetic transfer function, i.e. $\tau = 0.15$ s, and a smaller peak is present at the second transfer function delay at $\tau = 0.3$ s. The Hilbert transform also identifies exactly the first and most important delay peak, but misses the second one. The Gabor transform peaks are again shifted slightly to the right, but this time additional side peaks appear when the frequency domain window is wide. They disappear by using a narrower window, as shown in the green curve, but these leads to a worse time-domain resolution of the transform, which makes both the first and second peaks appear in more inaccurate values of τ.

The computational cost of the LS-SVM method might be worrisome, as it depends on the inversion of matrices, which might be computationally slow if the involved matrices are large. However, Table 1 shows that the impact on the performance was not so large in these examples. Using those $K = 1001$ frequency response samples, the LS-SVM required a computational time of 17.57 and 16.45 s, compared to 12.96 and 9.88 s for the Gabor transform. For smaller number of samples, the difference is reduced, up to a point where the LS-SVM may be faster than the Gabor transform, e.g., with 334 samples. The Hilbert transformed always required a larger computational time than both other methods.

The proposed method presents a trade-off between the maximum frequency of the system (or bandwidth), the maximum identifiable delay (related to the length of the interconnect) and computational performance (that depends on the number of frequency samples). For a fixed computational time, if the maximum frequency is increased, Δ_f increases, and therefore the maximum identifiable delay τ_M is smaller. On the other hand, channels that operate at higher frequencies would be naturally shorter, thus the delays that need to be identified in such channels are smaller, compensating the shrink of τ_M.

B. EXAMPLE 2: CIRCUIT WITH MULTIPLE TRANSMISSION LINE PATHS

The transfer function $H(j\omega) = V_o(j\omega)/V_i(j\omega)$ of the circuit in Fig. 4 presents a larger number of delays, due to multiple propagation paths. The circuit is composed by 5 transmission lines with different lengths (TL1: 0.47 m; TL2: 1.38 m; TL3: 0.31 m; TL4: 1.19 m; TL5: 0.28 m), resulting in 3 different paths from the input to the output, and additional paths generated by reflections at the various splitting points of the circuit. In addition to the transmission lines, there are also a few lumped resistors and capacitors, creating line mismatches and adding corrugations to the voltage/current waveforms.

The above circuit is simulated in SPICE, and $H(j\omega)$ is obtained at 1229 equally-spaced frequency points between 0 and 2 GHz. The proposed method is applied in the same sequence as described in example 1, but this time 25 kernel poles are adopted, due to the higher complexity of the transfer function. The obtained time-delay curve is shown in Fig. 5. In the same figure, the delays from the Hilbert and Gabor transform methods are also shown, for comparison. The detail of the first peak (shown in an inset of Fig. 5) is interesting: it reveals that the peak shape is almost coincident with the Gabor transform curve, but identifies a delay slightly higher than the Hilbert transform. Also for the additional peaks, the time-delay curve computed via the proposed approach presents a remarkable resemblance to the

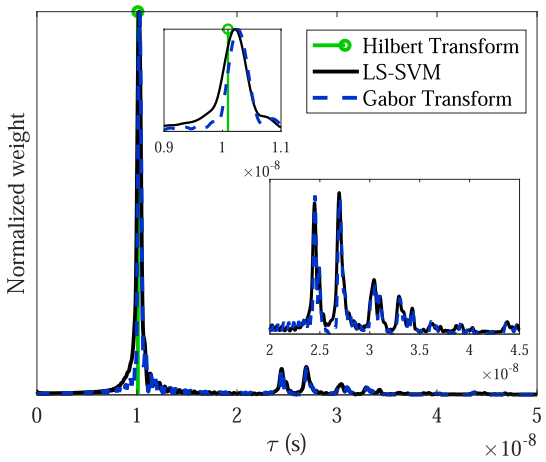


FIGURE 5. Example 2: plots of $w_p(\tau)$ for delay identification; comparison with Hilbert (green line) and Gabor (dashed blue curve) transforms is also displayed.

TABLE 2. Computational time required for delay identification in example 2 as a function of the number of considered points in the τ-axis.

Number of points in the τ-axis	Computational time [s]		
	LS-SVM	Gabor transform	Hilbert transform
1,000	27.0	5.36	261.8
3,333	65.3	55.8	
5,000	89.0	135.2	
10,000	165.9	566.0	

one obtained through the Gabor transform, although all peaks are slightly to the left of the Gabor transform ones.

The curves on the previous figure require the discretization of the τ-axis. Together with the number of frequency samples, this discretization affect the computational time of the curves. Table 2 shows how this time changes according to the number of τ points considered, in this example. It is seem that, for a small number of points, the Gabor transform performs better, but as the number of points increases, the LS-SVM becomes more efficient. The model estimation does not depend on the τ-axis discretization, and thus, it is computationally easy to compute the weights at a new point, with (34). The computational time of the Hilbert transform is higher, and it does not require any discretization.

Five significant delays are identify from the curve of Fig. 5: $\tau_1 = 10.199$ ns, $\tau_2 = 24.453$ ns, $\tau_3 = 26.941$ ns, $\tau_4 = 30.443$ ns, $\tau_5 = 32.931$ ns. Smaller peaks or peaks very close to larger ones are ignored. By using the five identified delays with Algorithm 2, the original transfer function $H(j\omega)$ is fitted using 12 poles in total. This approximation is shown in Fig. 6, indicating an excellent phase reconstruction over the entire bandwidth (lower panel).

In table 3, we compare the DRMs obtained using Algorithm 2 and the delays obtained through the proposed LS-SVM method, with three alternatives: a DRM obtained considering the delays from the Gabor transform and Algorithm 2; a single delay DRM that uses the Hilbert trans-

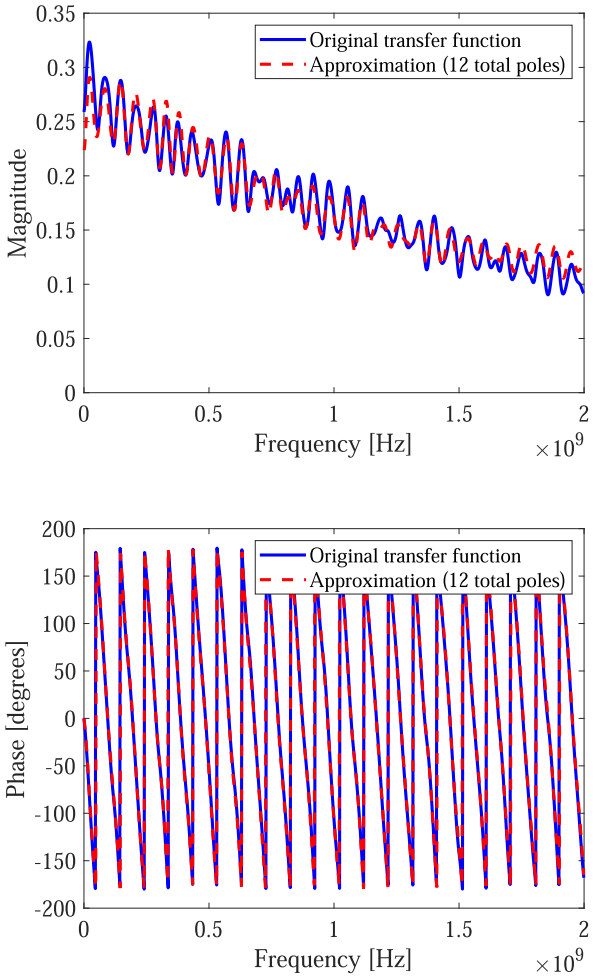


FIGURE 6. Magnitude and phase plots of the reconstructed transfer function of example 2.

form delay and VF to obtain the rational part; and a direct VF approximation. It is shown that a VF approximation with similar accuracy would require at least 40 poles, if the model accounts for the Hilbert transform delay, or 52 poles, if no delay is considered. This is more than three times the number of poles adopted by the proposed model. Moreover, with a very small change in the delays to $\tau_1 = 10.260$ ns, $\tau_2 = 24.514$ ns, $\tau_3 = 27.002$ ns, $\tau_4 = 30.566$ ns, $\tau_5 = 33.023$ ns, the model obtained using the delays from the Gabor transform does not achieve a good accuracy. The issue seems to rest on τ_1: by replacing it with the LS-SVM value, which is only 0.061 ns smaller, a good accuracy is restored. It is believed by the authors that $\check{H}_1(j\omega_k)$ in (37) might become non-causal when $H(j\omega_k)$ has a delay larger than its intrinsic delay compensated (i.e., if τ_1 is larger than the true delay in $H(j\omega_k)$), which hinders the rational approximation. Overall, the obtained results demonstrates that the identified delays help reducing the complexity of the model.

As a final comparison, we can look at the total computational time required to obtain these DRM or VF models, which is presented in Table 4. That data corresponds to the

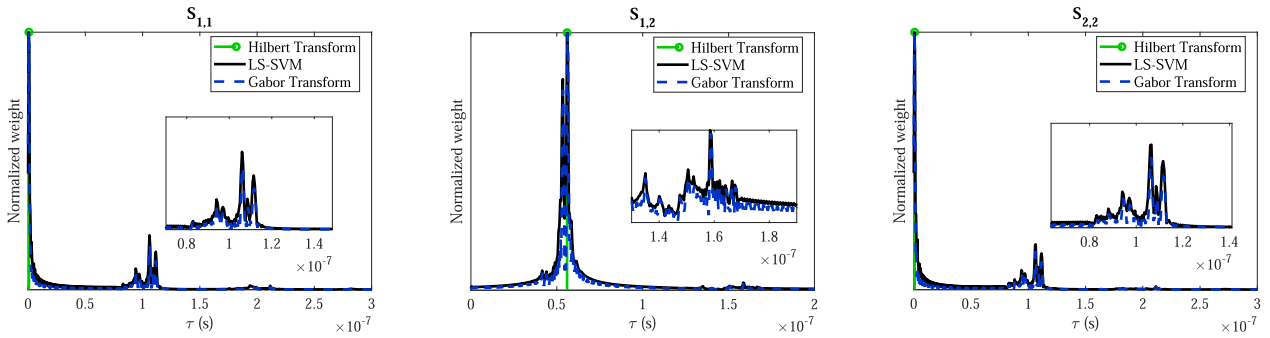


FIGURE 7. Example 3: plots of $W(\tau)$ for delay identification; comparison with Hilbert (green line) and Gabor (dashed blue curve) transforms is also displayed.

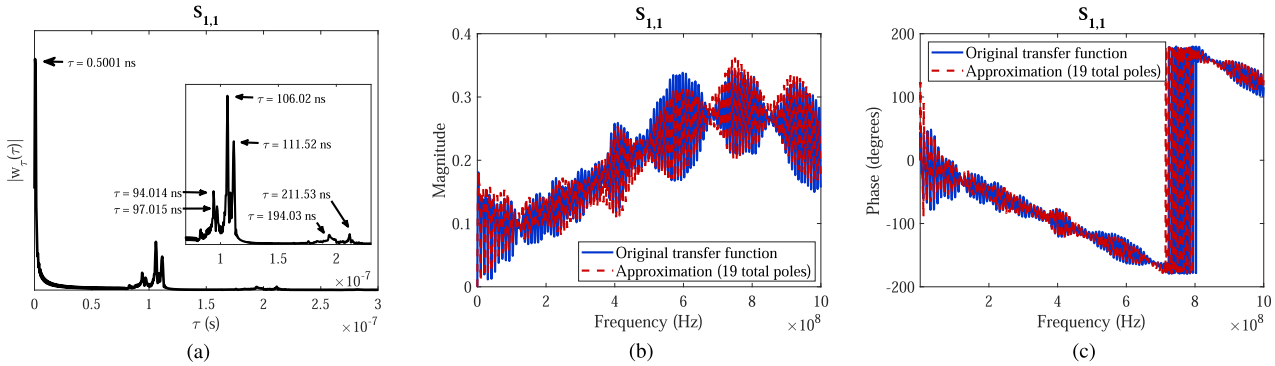


FIGURE 8. Example 3. (a) – Plot of $W(\tau)$ of $S_{1,1}$ for delay identification. (b) and (c) – Comparison of the model with original data for the magnitude and phase of $S_{1,1}(j\omega)$.

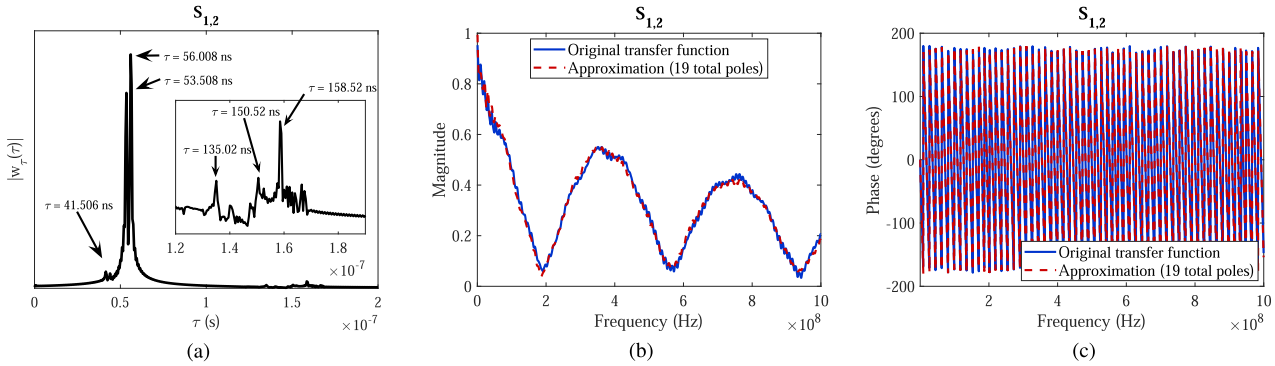


FIGURE 9. Example 3. (a) – Plot of $W(\tau)$ of $S_{1,2}$ for delay identification. (b) and (c) – Comparison of the model with original data for the magnitude and phase of $S_{1,2}(j\omega)$.

TABLE 3. Summary of the error between system response $H(j\omega)$ in example 2 and models $\tilde{H}(j\omega)$ used to approximate it.

	LS-SVM + Alg. 2	Gabor tr. + Alg. 2	Hilbert tr. + VF	VF
Error - L_2−norm	0.462	1.940	0.457	0.587
Error - L_∞-norm	0.040	0.138	0.033	0.037
Total poles	12	13	40	52

cases where 10,000 τ-values were evaluated to obtain the plot in Fig. 5. It is shown that VF is faster, but this was also the method with the highest number of poles. The proposed method took around five times longer than VF, but it requires around 4 times less poles, and had a better performance than the other DRM alternatives.

C. EXAMPLE 3: SpaceWire CABLE

Simulated data of a SpaceWire (SpW) link, containing a 10 m long SpaceWire cable with compatible connectors and PCB adapters [32] is considered as a final application example of the presented method. The structure of SpaceWire cable is rather complicated (4 twisted-shielded pairs surrounded by an external shield), thus requiring an 18 × 18 scattering

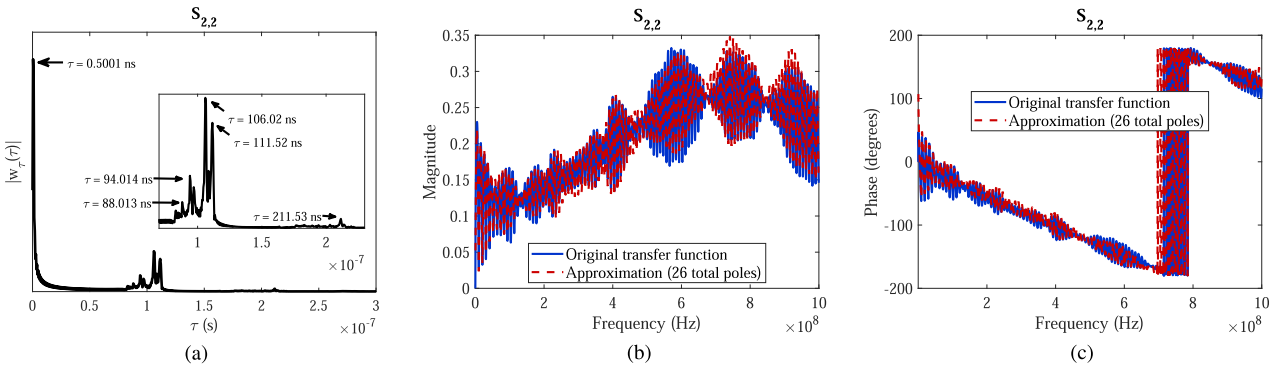


FIGURE 10. Example 3. (a) – Plot of $W(\tau)$ of $S_{2,2}$ for delay identification. (b) and (c) – Comparison of the model with original data for the magnitude and phase of $S_{2,2}(j\omega)$.

TABLE 4. Comparison of the computational time required for delay identification and rational fitting through the analyzed methods in example 2.

Method	Computational time [s]		
	Delay id.	Rational fit.	Total
LS-SVM + Alg. 2	165.9	0.898	166.8
Gabor tr. + Alg. 2	566.0	0.719	566.7
Hilbert tr. + VF	261.8	0.283	262.1
VF	-	32.31	32.31

matrix to fully describe the electromagnetic behaviour of the link. The simulation accounted for the entire structure of the SpaceWire cable (full details are given in [32]), but, for simplicity, we model here only the scattering parameters of a single wire, from end to end. The 1-wire link is then represented by the following scattering matrix $\mathbf{S}(j\omega)$:

$$S(j\omega) = \begin{bmatrix} S_{1,1}(j\omega) & S_{1,2}(j\omega) \\ S_{2,1}(j\omega) & S_{2,2}(j\omega) \end{bmatrix}, \tag{41}$$

where $S_{1,2}(j\omega)$ and $S_{2,1}(j\omega)$ are equal due to the reciprocity of the passive link.

The proposed method is applied to each of the elements of the matrix individually, and then a multiport model is achieved by combining the individual results [7]. A remarkable simplification when dealing with multiport systems comes from the fact that the kernel matrix $\mathbf{\Omega}$ depends only on the set of poles $\mathcal{P}$, τ_m, τ_M and the sampled frequency points $\boldsymbol{\omega}_k$, which are the same for all elements of the scattering matrix, i.e., the $k = 1, \ldots, 2000$ equally-spaced points between 0 and 1 GHz. The kernel does not depend on the frequency response itself. Therefore, the kernel matrix used to solve (50) can be computed only once, simplifying the computational process.

The proposed method is applied to $S_{1,1}(j\omega)$, $S_{1,2}(j\omega)$ and $S_{2,2}(j\omega)$ in the same manner as in example 2, with 25 kernel poles, $\tau_m = 0$ and $\tau_M = 1/\Delta_f = 2\ \mu$ s. In parallel and for the sake of comparison, the Hilbert and Gabor transform methods are applied to the scattering matrix of the SpW link. Figure 7 shows the results of the three methods, normalized so that their highest peak has the same value for all three plots. The plots document a very good agreement of the three techniques.

For $S_{1,1}(j\omega)$, Fig. 8-a shows the identified delays, of which the first one happens at $\tau = 0.5$ ns and represents a reflection occurring a few centimeters into the propagation path, where the PCB and connector transitions into the SpaceWire cable. Another 6 delays were identified with order of magnitude in the hundreds of ns. Furthermore, $\tau = 0$ is a candidate delay, representing the port reflection, due to a possible mismatches between the port and the input impedance of the structure; however, its identification is compromised by the very close first peak. Figures 8-b and 8-c show the magnitude and phase of the $S_{1,1}$ approximation, performed with 8 delays and 19 total poles.

For $S_{1,2}(j\omega)$, Fig. 9-a shows the identified delays. Since $S_{1,2}$ represents the end-to-end transmission of the link, the first peak appears after a considerable delay, contrary to the reflection term $S_{1,1}$. In total, 6 main delays were identified. Figures 9-b and 9-c show the approximation of the $S_{1,2}$ transfer function performed with the identified 6 delays and 19 total poles: the good quality of this approximation is evident.

The time-delays of $S_{2,2}$ are identified from the curve displayed in Fig. 10-a. They are similar to the ones of $S_{1,1}$, although not exactly the same, because the link is non-symmetrical. In total, 7 delays terms, including $\tau = 0$ due to direct reflection, are identified. In addition, 26 poles in total are needed for a good quality approximation, as shown in Figs. 10-b and 10-c.

The three alternative models presented in VI-B are built for comparison. A summary of the L_2 and L_∞-norm error between the original data and the constructed models is shown in Table 5. It is evident that the proposed DRM model presents a much lower complexity when compared with the mere rational VF model. In this example, it achieved a similar performance than the model that uses the Gabor transform delays. The Hilbert transform delay also led to an accurate and low order model, but only for the transmission scattering parameter $S_{1,2}$. For the reflections, this method is not applicable, and thus it would require the high order VF models to build the complete multiport system.

TABLE 5. Summary of the error between available data for the link in Example 3 and models used to approximate it.

	LS-SVM + Alg. 2	Gabor tr. + Alg. 2	Hilbert tr. + VF	VF
$S_{1,1}$ error - L_2	0.841	1.052	-	0.894
$S_{1,1}$ error - L_∞	0.098	0.070	-	0.090
$S_{1,1}$ order	19	24	-	159
$S_{1,2}$ error - L_2	0.907	0.905	0.871	0.735
$S_{1,2}$ error - L_∞	0.052	0.052	0.132	0.152
$S_{1,2}$ - order	19	20	35	122
$S_{2,2}$ error - L_2	0.849	1.122	-	0.823
$S_{2,2}$ error - L_∞	0.074	0.074	-	0.083
$S_{2,2}$ - order	26	26	-	162

TABLE 6. Computational time required for delay identification and rational fitting of the multiport system from example 3.

Cumputational time [s]		LS-SVM + Alg. 1	Gabor tr. + Alg. 1	Hilbert tr. + VF	VF
Delay estimation	Kernel computation	43.9	-	-	-
	$S_{1,1}$	103.1	126.3	-	-
	$S_{1,2}$	106.0	123.4	443.1	-
	$S_{2,2}$	104.4	123.6	-	-
Rat. fitting	$S_{1,1}$	1.86	1.99	44.06	44.06
	$S_{1,2}$	1.63	1.56	0.16	53.58
	$S_{2,2}$	1.67	2.30	44.06	53.76
Total:		362.6	379.2	531.4	151.4

The total computational time required to build the four aforementioned models is also relevant. It is compared in Table 6, for both the delay identification and the rational fitting. We see that the faster complete procedure is VF (which could be even faster using techniques specific for multiport systems [3]). Nonetheless, the total time spent by the DRM techniques is reasonable, given the remarkable reduction in the number of required poles. Moreover, we see that the fact of computing the kernel matrix only once provides an advantage for the LS-SVM in terms of delay estimation time.

VII. CONCLUSION

DRM models are one of the best alternatives to reduce the complexity of models of distributed systems. However, the extended use of such a clever implementation is hindered by the difficulty in estimating accurately both the poles and delays needed for their representation. The present work proposed a new method to assist on the construction of a DRM. The approach consists in approximating the frequency response of the system by means of the dual space formulation of the LS-SVM with an ad-hoc kernel based on an infinite number of delayed-rational basis; and the analysis of the weights of this approximation which allows us to identify the dominant time-delays present in the distributed system, that represent the most critical elements of the model. Once those terms are identified, only the rational part of the DRM needs to be estimated, and this can be done through the more conventional VF procedure.

The lower complexity of the resulting DRM implies that an equivalent circuit for the distributed system would require less dynamic components, and thus would be processed more efficiently by time-domain simulators. A clear task that can benefit from this speed-up is the assessment of the integrity of signal propagating in a system with electric interconnects. The DRM has also the advantage of being causal, as positive delays are used in the transfer function; furthermore, its phase is very accurate at all frequencies, while VF models could loose accuracy for smaller amplitudes.

Comparing the results of this work with similar existing delay-identification methods, it can be readily observed that the present technique, like the Gabor transform, has the ability to identify multiple propagation delays. However, the Gabor transform suffers from the necessity of tunning the width of its window function, which can push the identified delays to values larger than the actual value. Meanwhile the Hilbert transform is limited to a single delay. The distinctive feature of our approach is its DRM structure, inspired by the physical structure of electronic systems of large dimensions, where the signal propagation is subject to time-delays. In addition, the flexibility provided by the kernel is unique; in fact, the considered poles and time-delay interval can be adapted according to the knowledge about the system, and a strategy of handling multiple time-delay intervals deserves future investigations, together with the possible extension to a large number of parameters.

APPENDIX
EXTENSION OF THE LS-SVM TO COMPLEX FUNCTIONS

Starting from a set of complex training pairs $\{(\mathbf{x}_k, y_k)\}_{k=1}^K$ with the scalar output $y_k \in \mathbb{C}$ and possibly multiple input parameters arranged in a vector $\mathbf{x}_k \in \mathbb{C}^d$, we are looking for the following *primal* space formulation of the LS-SVM regression, which writes:

$$\tilde{y}(\mathbf{x}) = \langle \mathbf{w}^*, \boldsymbol{\varphi}(\mathbf{x}) \rangle + b, \tag{42}$$

where $\mathbf{w} \in \mathbb{C}^D$ are the regression coefficients, $\boldsymbol{\varphi}(\cdot) : \mathbb{C}^d \to \mathbb{C}^D$ are the considered basis functions, $b \in \mathbb{C}$ is a bias term and * represents the element-wise complex-conjugation operation.

The LS-SVM is derived by solving the following optimization problem:

$$\begin{aligned} \min_{\mathbf{w},\boldsymbol{e}} J_p(\mathbf{w}, \boldsymbol{e}) &= \frac{1}{2}\mathbf{w}^H\mathbf{w} + \frac{\gamma}{2}\sum_{k=1}^{K} e_k e_k^* \\ s.t.\ \Re\{\mathbf{w}^T\boldsymbol{\varphi}(\mathbf{x}_k) + b + e_k\} &= y_{k,r} \\ \Im\{\mathbf{w}^T\boldsymbol{\varphi}(\mathbf{x}_k) + b + e_k\} &= y_{k,i} \end{aligned} \tag{43}$$

for $k = 1, \ldots, K$. In the above equation, the superscript ^{H} denotes the conjugate-transpose operation and the vector of model error $\boldsymbol{e} = \{e_k\}_{k=1}^K$ is equal to the true value y_k minus the model output of (42) computed at $\mathbf{x}_k$, for all training points. Problem (43) minimizes the L2-norm of the primal space model coefficients $\mathbf{w}$ plus the sum of squared errors, weighted by the regularization parameter γ. Such parameter, commonly referred to as hyperparameter, is tuned during

the model training usually via cross-validation. The optimal model is found by means of the Lagrangian, which for this problem writes:

$$\begin{aligned}\mathcal{L}(\mathbf{w}, b, \mathbf{e}; \boldsymbol{\alpha}) &= \frac{1}{2}\left(\mathbf{w}_r^T\mathbf{w}_r + \mathbf{w}_i^T\mathbf{w}_i\right) \\ &+ \gamma\sum_{k=1}^{K}\left(e_{k,r}^2 + e_{k,i}^2\right) \\ &- \sum_{k=1}^{K}\alpha_{k,r}\left(\mathbf{w}_r^T\boldsymbol{\varphi}_r(\mathbf{x}_k) - \mathbf{w}_i^T\boldsymbol{\varphi}_i(\mathbf{x}_k) + b_r + e_{k,r} - \mathrm{y}_{k,r}\right) \\ &- \sum_{k=1}^{K}\alpha_{k,i}\left(\mathbf{w}_r^T\boldsymbol{\varphi}_i(\mathbf{x}_k) + \mathbf{w}_i^T\boldsymbol{\varphi}_r(\mathbf{x}_k) + b_i + e_{k,i} - \mathrm{y}_{k,i}\right),\end{aligned} \tag{44}$$

where $\alpha_k = \alpha_{k,r} + j\,\alpha_{k,i}$ are Lagrange multipliers, $\mathbf{w} = \mathbf{w}_r + j\,\mathbf{w}_i$, $e_k = e_{k,r} + j\,e_{k,i}$, $\mathrm{y}_k = \mathrm{y}_{k,r} + j\,\mathrm{y}_{k,i}$ and $\boldsymbol{\varphi}(\mathbf{x}_k) = \boldsymbol{\varphi}_r(\mathbf{x}_k) + j\,\boldsymbol{\varphi}_i(\mathbf{x}_k)$. The derivatives of the Lagrangian are equal to:

$$\frac{\partial\mathcal{L}}{\partial\mathbf{w}_r} = \frac{1}{2}(2\mathbf{w}_r) - \sum_{k=1}^{K}\alpha_{k,r}\boldsymbol{\varphi}_r(\mathbf{x}_k) - \sum_{k=1}^{K}\alpha_{k,i}\boldsymbol{\varphi}_i(\mathbf{x}_k), \tag{45a}$$

$$\frac{\partial\mathcal{L}}{\partial\mathbf{w}_i} = \frac{1}{2}(2\mathbf{w}_i) + \sum_{k=1}^{K}\alpha_{k,r}\boldsymbol{\varphi}_i(\mathbf{x}_k) - \sum_{k=1}^{K}\alpha_{k,i}\boldsymbol{\varphi}_r(\mathbf{x}_k), \tag{45b}$$

$$\frac{\partial\mathcal{L}}{\partial b_r} = -\sum_{k=1}^{K}\alpha_{k,r}, \tag{45c}$$

$$\frac{\partial\mathcal{L}}{\partial b_i} = -\sum_{k=1}^{K}\alpha_{k,i}, \tag{45d}$$

$$\frac{\partial\mathcal{L}}{\partial e_{k,r}} = -\alpha_{k,r} + 2\frac{\gamma}{2}e_{k,r}, \tag{45e}$$

$$\frac{\partial\mathcal{L}}{\partial e_{k,i}} = -\alpha_{k,i} + 2\frac{\gamma}{2}e_{k,i}, \tag{45f}$$

$$\frac{\partial\mathcal{L}}{\partial\alpha_{k,r}} = -(\mathbf{w}_r^T\boldsymbol{\varphi}_r(\mathbf{x}_k) - \mathbf{w}_i^T\boldsymbol{\varphi}_i(\mathbf{x}_k) + b_r + e_{k,r} - \mathrm{y}_{k,r}), \tag{45g}$$

$$\frac{\partial\mathcal{L}}{\partial\alpha_{k,i}} = -(\mathbf{w}_r^T\boldsymbol{\varphi}_i(\mathbf{x}_k) + \mathbf{w}_i^T\boldsymbol{\varphi}_r(\mathbf{x}_k) + b_i + e_{k,i} - \mathrm{y}_{k,i}). \tag{45h}$$

Making the derivatives equal to zero results in the definitions below:

$$\mathbf{w} = \sum_{k=1}^{K}\alpha_k\boldsymbol{\varphi}^*(\mathbf{x}_k), \tag{46}$$

$$\sum_{k=1}^{K}\alpha_k = 0, \tag{47}$$

$$e_k = \frac{\alpha_k}{\gamma}, \tag{48}$$

$$\mathbf{w}^T\boldsymbol{\varphi}(\mathbf{x}_k) + b + e_k - \mathrm{y}_k = 0. \tag{49}$$

By combining the equations in (49) and (47) and separating its real and imaginary parts, and substituting in (49) the value for $\mathbf{w}$ and e_k in (46) and (48), one can build the following linear system of equations:

$$\begin{bmatrix} \boldsymbol{\Omega} + \mathbf{I}^{2K}/\gamma & \mathbf{1} & \mathbf{0} \\ & \mathbf{0} & \mathbf{1} \\ \mathbf{1}^T \quad \mathbf{0}^T & 0 & 0 \\ \mathbf{0}^T \quad \mathbf{1}^T & 0 & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{\alpha}_r \\ \boldsymbol{\alpha}_i \\ b_r \\ b_i \end{bmatrix} = \begin{bmatrix} \mathbf{y}_r \\ \mathbf{y}_i \\ 0 \\ 0 \end{bmatrix}, \tag{50}$$

where $\mathbf{1} = [1, \ldots, 1]^T$ and $\mathbf{0} = [0, \ldots, 0]^T$ are vectors containing K equal elements, $\mathbf{I}^{2K}$ is an identity matrix with size $2K$, and the $2K \times 2K$ kernel matrix $\boldsymbol{\Omega}(\mathbf{x}_k, \mathbf{x}_k)$ is defined as

$$\boldsymbol{\Omega} = \begin{bmatrix} \boldsymbol{\Omega}^{(1,1)} & \boldsymbol{\Omega}^{(1,2)} \\ \boldsymbol{\Omega}^{(2,1)} & \boldsymbol{\Omega}^{(2,2)} \end{bmatrix}. \tag{51}$$

In the $K \times K$ square submatrices that compose $\boldsymbol{\Omega}$, an element $\Omega_{j,i}^{(\cdot,\cdot)}$ in the j-th row and i-th column is defined as follows:

$$\Omega_{j,i}^{(1,1)} = \Re\{\mathrm{k}(\mathbf{x}_i, \mathbf{x}_j)\}, \tag{52a}$$

$$\Omega_{j,i}^{(1,2)} = -\Im\{\mathrm{k}(\mathbf{x}_i, \mathbf{x}_j)\}, \tag{52b}$$

$$\Omega_{j,i}^{(2,1)} = \Im\{\mathrm{k}(\mathbf{x}_i, \mathbf{x}_j)\}, \tag{52c}$$

$$\Omega_{j,i}^{(2,2)} = \Re\{\mathrm{k}(\mathbf{x}_i, \mathbf{x}_j)\}, \tag{52d}$$

for $i, j = 1, \ldots, K$. The kernel function $\mathrm{k}(\mathbf{x}_i, \mathbf{x}_j)$ is defined as:

$$\mathrm{k}(\mathbf{x}_i, \mathbf{x}_j) = \left\langle\boldsymbol{\varphi}(\mathbf{x}_i), \boldsymbol{\varphi}(\mathbf{x}_j)\right\rangle. \tag{53}$$

The unknowns $\boldsymbol{\alpha} = \boldsymbol{\alpha}_r + j\,\boldsymbol{\alpha}_i = [\alpha_{1,r} + j\,\alpha_{1,i}, \ldots, \alpha_{K,r} + j\,\alpha_{K,i}]$ and $b = b_r + j\,b_i$ are computed by solving (50). The determination of the previous parameters leads to the *dual* space representation of the regression, written as

$$\tilde{\mathrm{y}}(\mathbf{x}) = \sum_{k=1}^{K}\alpha_k\mathrm{k}(\mathbf{x}_k, \mathbf{x}) + b, \tag{54}$$

where the Lagrange multipliers α_k become the unknowns for the model.

REFERENCES

[1] R. Achar and M. S. Nakhla, "Simulation of high-speed interconnects," *Proc. IEEE*, vol. 89, no. 5, pp. 693–728, May 2001.
[2] R. C. Paul, *Analysis of Multiconductor Transmission Lines*, 2nd ed. Hoboken, NJ, USA: Wiley, 2008.
[3] S. Grivet-Talocia and B. Gustavsen, *Passive Macromodeling: Theory and Applications*. Hoboken, NJ, USA: Wiley, 2016.
[4] B. Gustavsen and A. Semlyen, "Rational approximation of frequency domain responses by vector fitting," *IEEE Trans. Power Del.*, vol. 14, no. 3, pp. 1052–1061, Jul. 1999.
[5] R. Mandrekar and M. Swaminathan, "Causality enforcement in transient simulation of passive networks through delay extraction," in *Proc. 9th IEEE Workshop Signal Propag. Interconnects*, May 2005, pp. 25–28.
[6] S. N. Lalgudi, E. Engin, G. Casinovi, and M. Swaminathan, "Accurate transient simulation of interconnects characterized by band-limited data with propagation delay enforcement in a modified nodal analysis framework," *IEEE Trans. Electromagn. Compat.*, vol. 50, no. 3, pp. 715–729, Aug. 2008.
[7] A. Chinea, P. Triverio, and S. Grivet-Talocia, "Delay-based macromodeling of long interconnects from frequency-domain terminal responses," *IEEE Trans. Adv. Packag.*, vol. 33, no. 1, pp. 246–256, Feb. 2010.
[8] A. Charest, M. S. Nakhla, R. Achar, D. Saraswat, N. Soveiko, and I. Erdin, "Time domain delay extraction-based macromodeling algorithm for long-delay networks," *IEEE Trans. Adv. Packag.*, vol. 33, no. 1, pp. 219–235, Feb. 2010.

[9] A. Chinea, S. Grivet-Talocia, H. Hu, P. Triverio, D. Kaller, C. Siviero, and M. Kindscher, "Signal integrity verification of multichip links using passive channel macromodels," *IEEE Trans. Compon., Packag., Manuf. Technol.*, vol. 1, no. 6, pp. 920–933, Jun. 2011.

[10] M. Luo and K.-M. Huang, "An extended delay-rational macromodel for electromagnetic interference analysis of mixed signal circuits," *Prog. Electromagn. Res.*, vol. 127, pp. 189–210, 2012.

[11] M. Sahouli and A. Dounavis, "Delay extraction-based modeling using loewner matrix framework," *IEEE Trans. Compon., Packag., Manuf. Technol.*, vol. 7, no. 3, pp. 424–433, Mar. 2017.

[12] B. Gustavsen and J. Nordstrom, "Pole identification for the universal line model based on trace fitting," *IEEE Trans. Power Del.*, vol. 23, no. 1, pp. 472–479, Jan. 2008.

[13] B. Gustavsen, "Optimal time delay extraction for transmission line modeling," *IEEE Trans. Power Del.*, vol. 32, no. 1, pp. 45–54, Feb. 2017, doi: 10.1109/TPWRD.2016.2609039.

[14] B. Gustavsen, "Time delay identification for transmission line modeling," in *Proc. 45th Annu. IEEE Symp. Found. Comput. Sci.*, May 2004, pp. 103–106.

[15] J. Cho, J. Ahn, J. Kim, J. Park, Y. Shin, K. Kim, J. Choi, and S. Ahn, "Low- and high-frequency extrapolation of band-limited frequency responses to extract delay causal time responses," *IEEE Trans. Electromagn. Compat.*, early access, Oct. 6, 2020, doi: 10.1109/TEMC.2020.3025132.

[16] L. Barannyk, H. Tran, A. Elshabini, and F. Barlow, "Time delay extraction from frequency domain data using causal Fourier continuations for high-speed interconnects," *Electronics*, vol. 4, no. 4, pp. 799–826, Oct. 2015.

[17] S. Grivet-Talocia, "Delay-based macromodels for long interconnects via time-frequency decompositions," in *Proc. IEEE Electr. Performane Electron. Packag.*, Oct. 2006, pp. 199–202.

[18] S. Roy and A. Dounavis, "Transient simulation of distributed networks using delay extraction based numerical convolution," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 30, no. 3, pp. 364–373, Mar. 2011.

[19] A. Charest, D. Saraswat, M. Nakhla, R. Achar, and N. Soveiko, "Compact macromodeling of high-speed circuits via delayed rational functions," *IEEE Microw. Wireless Compon. Lett.*, vol. 17, no. 12, pp. 828–830, Dec. 2007.

[20] P. Triverio, S. Grivet-Talocia, and A. Chinea, "Identification of highly efficient delay-rational macromodels of long interconnects from tabulated frequency data," *IEEE Trans. Microw. Theory Techn.*, vol. 58, no. 3, pp. 566–577, Mar. 2010.

[21] M. Sgueglia, A. Sorrentino, M. de Magistris, D. Spina, D. Deschrijver, and T. Dhaene, "A novel parametric macromodeling technique for electromagnetic structures with propagation delays," in *Proc. IEEE 21st Workshop Signal Power Integrity (SPI)*, May 2017, pp. 1–4.

[22] F. Treviso, R. Trinchero, and F. G. Canavero, "Machine learning applied to the blind identification of multiple delays in distributed systems," in *Proc. 33rd Gen. Assem. Sci. Symp. Int. Union Radio Sci.*, Aug. 2020, pp. 1–4, doi: 10.23919/URSIGASS49373.2020.9232325.

[23] R. Trinchero, M. Larbi, H. M. Torun, F. G. Canavero, and M. Swaminathan, "Machine learning and uncertainty quantification for surrogate models of integrated devices with a large number of parameters," *IEEE Access*, vol. 7, pp. 4056–4066, 2019.

[24] A. V. Oppenhein, R. W. Schaffer, and J. R. Buck, *Discrete-Time Signal Processing*, 2nd ed. Upper Saddle River, NJ, USA: Prentice-Hall, 1998.

[25] A. Mertins, *Signal Analysis: Wavelets, Filter Banks, Time-Frequency Transforms and Applications—Revised Edition*. Chichester, U.K.: Wiley, , 1999.

[26] N. Cristianini and J. Shawne-Taylor, *An Introduction to Support Vector Machines and Other Kernel Based Learning Methods*. Cambridge, U.K.: Cambridge Univ. Press, 2000.

[27] H. Wang and D. Hu, "Comparison SVM LS-SVM for regression," in *Proc. Int. Conf. Neural Netw. Brain*, Beijing, China, Oct. 2005, pp. 279–283.

[28] P. Bouboulis, S. Theodoridis, C. Mavroforakis, and L. Evaggelatou-Dalla, "Complex support vector machines for regression and quaternary classification," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 6, pp. 1260–1274, Jun. 2015.

[29] R. Boloix-Tortosa, J. J. Murillo-Fuentes, I. Santos, and F. Perez-Cruz, "Widely linear complex-valued kernel methods for regression," *IEEE Trans. Signal Process.*, vol. 65, no. 19, pp. 5240–5248, Oct. 2017.

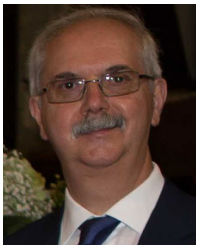
[30] J. A. K. Suykens, T. Van Gestel, J. De Brabanter, B. De Moor, and J. Vandewalle, *Least Squares Support Vector Machines*. Singapore: World Scientific, 2002.

[31] D. Hughes-Hallett, A. M. Gleason, W. G. McCallum, D. E. Flath, P. F. Lock, and T. W. Tucker, *Calculus: Single and Multivariable*, 7th ed. Hoboken, NJ, USA: Wiley.

[32] F. Treviso, R. Trinchero, and F. G. Canavero, "Validation of a physical-based model for a spacewire cable," in *Proc. ESA Workshop Aerosp. EMC (Aerosp. EMC)*, May 2019, pp. 1–6.

FELIPE TREVISO (Student Member, IEEE) was born in Guaporé, Brazil. He received the bachelor's and master's degrees in electrical engineering from the Universidade Federal do Rio Grande do Sul, Porto Alegre, Brazil, in 2014 and 2016, respectively. He is currently pursuing the Ph.D. degree with the Electronics Department, Politecnico di Torino, Torino, Italy, working on the modeling of electrical interconnects. He was awarded an honorable mention in the URSI Student Paper Competition at the URSI General Assembly and Scientific Symposium 2020.

RICCARDO TRINCHERO (Member, IEEE) received the M.Sc. and Ph.D. degrees in electronics and communication engineering from the Politecnico di Torino, Torino, Italy, in 2011 and 2015, respectively. He is currently an Assistant Professor with the EMC Group, Department of Electronics and Telecommunications, 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.



FLAVIO G. CANAVERO (Fellow, IEEE) received the degree in electronic engineering from the Politecnico (Technical University) of Torino, Italy, and the Ph.D. degree from the Georgia Institute of Technology, Atlanta, GA, USA, in 1986. He is currently a Professor of Circuit Theory with the Department of Electronics and Telecommunications, Politecnico di Torino. His research interests include signal integrity and EMC design issues, interconnect modeling, black-box characterization of digital integrated circuits, EMI, and statistics in EMC. 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. He has been the Editor-in-Chief of IEEE TRANSACTIONS ON ELECTROMAGNETIC COMPATIBILITY, the Vice President (VP) for Communication Services of the EMC Society, and the Chair of URSI Commission E.