Sparse Flexible Reduced-Volterra Model for Power Amplifier Digital Pre-Distortion

CYRO S. HEMSI AND CRISTIANO M. PANAZIO, (Member, IEEE)
Department of Telecommunication and Control (PTC), University of São Paulo (USP), São Paulo 05508-900, Brazil
Corresponding author: Cyro S. Hemsi (cyro.hemsi@alumni.usp.br)

This work was supported in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES), Finance Code 001.

ABSTRACT The Volterra framework plays an important role in digital pre-distortion (DPD) for power amplifier (PA) linearization. In practice, the full-Volterra (FV) model is avoided due to the so-called curse of dimensionality, which has motivated several Volterra pruning techniques in the literature. However, choosing the pruned-Volterra structure with a given complexity that most accurately describes a PA or DPD with unknown characteristics often requires trial and error. The motivation of this article is to propose a reduced-Volterra model that extends traditional memory polynomials, e.g., the generalized memory polynomial (GMP), by including higher-dimensional terms in a flexible and controlled way, avoiding the curse of dimensionality of the Volterra model and resulting in a model that can apply to a wide range of PA and DPD. Furthermore, the article investigates the parsimonious sizing of the proposed model using least absolute shrinkage and selection operator (LASSO) and sparse-group LASSO (SGL) convex optimization, significantly reducing the model’s running cost, while preserving its performance. In our experimental validation, the sparse versions of the proposed model achieved 3 dB better adjacent channel power ratio (ACPR) than the conventional GMP model for the same or lower DPD running cost, at the expense of a more costly estimation, which is performed off-line.

INDEX TERMS Baseband Volterra model, behavioural modelling, digital pre-distortion, LASSO, memory polynomial, power amplifier, sparse-group LASSO.

I. INTRODUCTION The Volterra framework [1], [2] plays an important role in digital pre-distortion (DPD) and baseband modelling of high-power and wideband nonlinear (NL) power amplifiers (PA) [3], [4], [5], [6], although, in practice, the full-Volterra (FV) model is avoided due to the large number of coefficients to be estimated, the length of the corresponding filter and its on-line running cost. The so-called FV curse of dimensionality has motivated several Volterra pruning techniques in the literature, the most prominent being the memory polynomial (MP) [7], the generalized memory polynomial (GMP) [8], the dynamic deviation reduction (DDR) [9], [10] and block-oriented [11], [12] behavioural models. Moreover, several specialized models have also been proposed, each of which taking into account additional nonlinearities, e.g., [13], [14], [15], [16], [17], [18], [19], [20], [21], [22], and [23]. Nevertheless, choosing among several candidates with a given complexity the most accurate pruned structure to describe a PA or DPD with unknown characteristics often requires trial and error. It is not a priori guaranteed that one such memory polynomial or block-oriented model adequately fits a NL system, without any physical knowledge available. This is one major challenge with pruned-Volterra behavioural models, such as the MP, GMP and DDR, compared to models developed from physical knowledge of the systems, such as PA operating/environmental conditions. The latter, in contrast are difficult to derive, typically more complex and less general [12]. Another challenge with behavioural modelling concerns to sizing, or dimensioning, i.e. setting appropriate values for the model parameters. Typically, the search for the best features (or regressors) to be added to a model of a given complexity is a combinatorial task, performed by exhaustive search [24], becoming

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/
a non-deterministic polynomial time (NP-hard) problem [25]. Indeed, although heuristic methods, such as hill-climbing [26], particle swarm optimization (PSO) [27], [28] and evolutionary algorithms [29] have been proposed, they still suffer from scalability [24], [30].

Furthermore, PA/DPD models usually admit parsimonious/sparse approximations [31], [32], [33], [34], [35], i.e., only a few of the PA/DPD model coefficients are relevant, either because the physical system is sparse or the original model is over-parametrized. The search for the optimum sparse set of model coefficients based on the \( \ell_0 \) pseudo-norm penalty is a known NP-hard problem, this way, among several techniques [36], most of the attention has been focused on the convex \( \ell_1 \) norm relaxation, i.e., the least absolute shrinkage and selection operator (LASSO) [37], [38] and on the \( \ell_q \) approximations, such as the orthogonal matching pursuit (OMP) [39], [40]. In some PA/DPD papers, sparse techniques have been applied to the FV model, such as LASSO in [31] and [32] and greedy OMP in [34], however, using the FV model is more costly than if a pruned model was considered. In [33], LASSO is applied to the parallel Hammerstein (PH) model; in this case, the obtained performance may suffer from limitations of the initial model. Finally, in [35], the sparse compressed-sampling matching pursuit (CoSaMP) technique is considered with pruned Volterra models, being advantageous in comparison to principal component analysis (PCA). In our paper, we choose LASSO-based techniques, specially the sparse-group LASSO (SGL) [41], due to their group-sparse inducing properties, convexity, robustness and strong theoretical guarantees [39], [40]. Moreover, we also investigate LASSO and SGL as a convex optimization approach for the sizing of the proposed behavioural model. For the SGL technique, we first represent the model in terms of higher dimension regressors to be included and is also economic, by allowing higher order model parameters to be controlled; (b) Furthermore, we propose a block-wise modular structure for the proposed model and apply SGL estimation to select the subset of the most relevant blocks/regressors, which leads to model sizing and reduction.

This paper is organized as follows. Section II provides background on the existing pruned-Volterra models and sparse estimation techniques; in Section III, the proposed DPD model is derived. Section IV proposes the modular structure of the proposed model and its SGL estimation. Section V provides the simulation and experimental results, followed by Section VI, where final conclusions are drawn.

II. SYSTEM MODEL

The baseband full-Volterra (FV) series model [42], [43], truncated to memory depth \( M \) and NL order \( K \), is given by:

\[
\hat{x}_{FV}(n) = \sum_{k=1, m_1=0}^{K} \sum_{m_2=0}^{M} \tilde{h}_k(m_{1:k}) \prod_{i=1}^{k+1} \tilde{u}(n - m_i) \times \prod_{j=k+2}^{k+1} \tilde{u}^*(n - m_i),
\]

where \( \tilde{u}(n) \) and \( \hat{x}_{FV}(n) \) are, respectively, the DPD complex-valued, discrete-time, baseband input and modelled output signals, \( \cdot^* \) is the conjugate operator, and \( \tilde{h}_k(m_{1:k}) \) are the \( k^{th} \) order FV kernels, where \( m_{1:k} = m_1, \ldots, m_k \). In Equation (1), the baseband Volterra kernels for \( k \geq 3 \) present a doubly-symmetry, i.e., all permutations of the indices within each of the products \( \prod_{i=1}^{k+1} \tilde{u}(n - m_i) \) and \( \prod_{j=k+2}^{k+1} \tilde{u}^*(n - m_i) \) result in the same components. Therefore, the FV model can be equivalently replaced by the triangular baseband FV model, more economical in the number of coefficients [43]. Moreover, in [44], a modified triangular Volterra \((MV) \) model is proposed, in which the memory depths of each NL order are independent. This way, shorter memory spans can be assigned to higher NL orders, thus reducing the model size, e.g., see [32]. The \( MV \) model is given by:

\[
\hat{x}_{MV}(n) = \sum_{k=1, m_1=0}^{K} \sum_{m_2=m_1}^{M} \sum_{m_3=m_2}^{M} \cdots \sum_{m_{k+1}=m_{k-1}}^{M} \tilde{h}_k(m_{1:k}) \prod_{i=1}^{k+1} \tilde{u}(n - m_i) \times \prod_{i=k+2}^{k+1} \tilde{u}^*(n - m_i),
\]
where $M_k$ is the memory depth for the $k^{th}$-order polynomials. This way, its required number of coefficients is:

$$N_{MV\Delta} = \sum_{k=1}^{K} \left( M_k + \frac{k+1}{2} \right) \left( M_k + \frac{k-1}{2} \right). \tag{3}$$

Although reducing the FV model size, the doubly-combinatorial in Equation (2) still grows very fast with the NL order and memory depths. This has motivated several other pruned-Volterra models in the literature; among the most prominent are:

- The MP model [7] that is limited to one-dimensional polynomials, i.e., built using a single input sample.
- The GMP model [8] that includes some two-dimensional polynomials, i.e., allows cross-sample products. The GMP model is given by:

$$\hat{x}_{GMP} (n) = \sum_{m=0}^{M_1} \sum_{k=1}^{K_1} \beta_{m,k}^{(1)} \tilde{u} (n - m) |\tilde{u} (n - m)|^{k-1} + \sum_{m=0}^{M_2} \sum_{k=1}^{K_2} \sum_{j=1}^{J_2} \beta_{m,k,j}^{(2)} \tilde{u} (n - m) \times |\tilde{u} (n - m - j)|^{k-1} + \sum_{m=0}^{M_3} \sum_{k=1}^{K_3} \sum_{j=1}^{J_3} \beta_{m,k,j}^{(3)} \tilde{u} (n - m) \times |\tilde{u} (n - m + j)|^{k-1}, \tag{4}$$

where parameters $M_1$ and $K_1$ are, respectively, the memory depth and NL order of the terms of aligned signal and powered envelope, $M_2$, $J_2$ and $K_2$ are, respectively, the memory depth, envelope time-shift and NL order of the terms between signal and lagging powered envelope and $M_3$, $J_3$ and $K_3$ are, respectively, the memory depth, envelope time-shift and NL order of the terms between signal and leading powered envelope.

- The DDR model [9] that controls the power orders of the delayed samples in the FV model, as described in [10], instead of limiting the polynomial dimension. For example, the polynomials $\tilde{u}(n - m_1)\tilde{u}(n)$ and $\tilde{u}(n - m_1)\tilde{u}(n - m_2)\tilde{u}(n)\tilde{u}(n)$ belong, respectively, to the first- and second-order DDR models.

Even more sophisticated models have also been derived in recent papers from the FV model, such as in [15], [16], [17], [18], and [19]; among them, the complexity reduced Volterra (CRV) model proposed in [18] is given by:

$$\hat{x}_{CRV} (n) = \sum_{k=1}^{K} \frac{K-2}{2} \beta_{k+1}^{(1)} \tilde{u}^{k+1} (n) |\tilde{u}^{k+1} (n)|^{k-1} + \sum_{k=1}^{K} \sum_{m=1}^{M} \sum_{q=2}^{Q} \beta_{k,m,q}^{(2)} \tilde{u}^{k+1} (n) |\tilde{u}^{k+1} (n)|^{k-1} \times |\tilde{u}^{k+1} (n - m)\tilde{u}^{k+1} (n - m)|^{k-1}, \tag{5}$$

where $\cdot^*$ is the conjugate operator and $K$, $Q$, and $M$ are, respectively, the NL order of instantaneous samples, NL order of lagging samples and the memory depth.

### A. ESTIMATION TECHNIQUES

During the DPD training phase and following the indirect learning architecture (ILA) [45], we measure, down-convert and process the length-$N$ signals $\hat{x}$ and $\hat{y}$, respectively, the PA undistorted input and NL output. In the ILA, the DPD training postdistorter, placed after the PA, estimates directly the inverse model, by interchanging the roles of the signals above, i.e., using $\hat{u} = \hat{y} / G_{PA}$ as the post-inverse normalized input ($G_{PA}$ is the PA linear gain) and minimizing the least squares error between $\hat{x}$ and the modelled post-inverse output $\hat{\tilde{x}}$, i.e., $\|\tilde{x} - \hat{\tilde{x}}\|^2_2$, where $\|\cdot\|^2_2$ is the Euclidean norm [46]. After the training phase is completed, the postdistorter inverse model is copied to the DPD predistorter [1]. Using the $MV_{\Delta}$ model from Equation (2), which is linear in the coefficients, the post-inverse output signal can be modelled as $\hat{x} = \Phi \hat{\beta}$, where $\Phi$ is the regression matrix and $\hat{\beta}$ are coefficients to be estimated by ordinary least squares (OLS), as follows:

$$\hat{\beta}_{OLS} = \arg \min_{\beta \in \mathbb{C}^R \times 1} \frac{1}{2N} \|\tilde{x} - \Phi \beta\|^2_2, \tag{6}$$

where $N$ is the number of samples, $R$ is the number of regressors, $\Phi$ is the $N \times R$ model-specific regression matrix and $\beta$ is the $R \times 1$ coefficient vector to be estimated. Finally, the least squares solution is given by $\hat{\beta}_{OLS} = (\Phi^H \Phi)^{-1} \Phi^H \tilde{x}$, assuming $\Phi^H \Phi$ is non-singular. However, as the model dimensions increase, frequently $\Phi$ becomes ill-conditioned (due to high correlation among regressors) and/or the OLS overfits the training data; in both cases, Equation (6) solution is compromised in terms of accuracy and stability [47] and ridge regression regularization [48] leads to better model prediction. Moreover, usually PA/DPD models admit parsimonious/sparse approximations [31], i.e., only a few of the PA/DPD model coefficients are relevant, either because the physical system is sparse or the original model is over-parametrized. In this work, we use LASSO-based techniques to select the subset of the most relevant regressors, while avoiding ill-conditioning and over-fitting. LASSO-based techniques are chosen due to group-sparse inducing properties, convexity, robustness and strong theoretical guarantees [39], [40], compared to greedy techniques, such as OMP. The LASSO batch estimator solves the following convex optimization problem:

$$\hat{\beta}_{LASSO}^{(l)} = \arg \min_{\beta \in \mathbb{C}^R \times 1} \left( \frac{1}{2N} \|\tilde{x} - \Phi \beta\|^2_2 + \lambda \|\beta\|_1 \right), \tag{7}$$
where $\|\beta\|_1$ is the $\ell_1$ norm, given by $\sum_{i=1}^R |\beta_i| = \sum_{i=1}^R \sqrt{r_i^2 + \Re(\beta_i)^2}$, in terms of the real and imaginary parts of the coefficients, and $\lambda$ is the sparsity parameter. One typically finds the LASSO solution by solving Equation (7) over a pre-defined range of $\lambda$ values and obtaining the best $\lambda$ by cross-validation [38] or more economic methods, such as the Akaike’s information criterion (AIC) [49]. Without loss of generality, we assume that the matrix $\Phi$ is standardized, such that each column $\phi_i$ is centered, i.e., $\frac{1}{N} \sum_{n=1}^N (\phi_i) = 0$ and normalized, i.e., $\frac{1}{N} \phi_i^H \phi_i = 1$, for all $r = 1, \ldots, R$. Examples in the literature that have applied LASSO to reduce the dimensionality of the FV model are [31], [32], and [33].

Furthermore, when it is possible to organize the model into blocks, the regression matrix becomes block-modular, i.e., $\Phi$ is divided into $G$ disjoint blocks and group-sparse LASSO extensions, such as group-LASSO [50], [51], [52] and SGL [41], can be applied. In this paper, we apply the bi-level SGL selection to explore the sparsity of the model both at the individual and block levels. The SGL cost function is given by:

$$\frac{1}{2N} \|\tilde{x} - \sum_{g=1}^G \Phi_g \beta_g\|_2^2 + \alpha \lambda \sum_{g=1}^G \|\beta_g\|_1$$

$$+ (1 - \alpha) \lambda \sum_{g=1}^G w_g \|\beta_g\|_2,$$  \tag{8}

where $\Phi_g$ is the $g^{th}$-group of regressors from $\Phi$, $\beta_g$ is the corresponding coefficient vector, $w_g$ is the group weight, often equal to $\sqrt{d_g}$ (where $d_g$ is the group length), $\lambda$ is the sparsity parameter and $\alpha \in [0, 1]$ is the additional SGL parameter. Equivalently, one can use $\lambda_1 = \alpha \lambda$ and $\lambda_2 = (1 - \alpha) \lambda$.

### III. PROPOSED FLEXIBLE VOLterra-BASED MODEL

This section presents the proposed reduced-Volterra model, whose parameters are defined independently for each NL order. This allows to increase the model parameters, such as the NL order, polynomial dimensions and memory depths in a controlled way, generalizing existing memory polynomials, such as the MP and GMP, and avoiding the curse of dimensionality. Moreover, by exploring higher polynomial dimensions, not assessed by the GMP model, the proposed model is expected to achieve better normalized mean square error (NMSE) and adjacent channel power ratio (ACPR). Our pruning strategy is described below. The fading memory behaviour of physical systems [44] allows to reduce the memory depth of higher NL orders of the model. Starting from the $\text{MV}_\Delta$ model in Equation (2), let, for each NL order $k$, the collection of input sample multi-indices, drawn from the finite set of integers $\{0, \ldots, m_k\}$, be represented as follows:

$$S^{(k)} = \{(m_1, m_2, \ldots, m_{k+1}) \mid \frac{m_{k+1}}{2}, \frac{m_{k+5}}{2}, \ldots, m_k\},$$

$$M^{(k)} = \{(m_1, m_2, \ldots, m_{k+1}) \mid m_1 + 1, m_2 + 1, \ldots, m_k + 1\}.$$  \tag{9}

in which $m_1$ is the first non-conjugated sample delay index. $M^{(k)}$ and $L^{(k)}_x$ are multiset, i.e., unordered collections of elements in which repetitions are allowed [53], [54], each with cardinality $k_1^\frac{k-1}{2}$, and containing, respectively, the delay multi-indices of the remaining non-conjugated and conjugated input samples. Additionally, the triangular model imposes that $m_{k+1} \geq m_{k+1} \geq \cdots \geq m_3$ and $m_1 \geq m_1 \geq \cdots \geq m_{k+3}$.

The pruning strategy is described in two steps, as follows. In step 1, the doubly-combinatorial structure in Equation (2) is replaced by a single-combinatorial one, inspired by memory polynomials, such as the MP and GMP. This way, following the MP/GMP structure, we only retain the polynomials from Equation (2) in which non-conjugated and conjugated indices appear in pairs of the same delays. This corresponds to imposing in Equation (9) that, for each $k$:

$$M^{(k)}_x = M^{(k)}_x,$$  \tag{10}

i.e., the two multisets are equal. This way, the vector of multi-indices reduces to $S^{(k)} = \{m_1, M^{(k)}_x\}$, where $M^{(k)}_x = (m_2, m_3, \ldots, m_{k+1})$ now contains the sample delay multi-indices of squared envelope terms. Thus, the reduced-Volterra model $\text{RV}_1$, after step 1, is given by:

$$\hat{\lambda}_{\text{RV}_1}(n) = \sum_{k=1}^K \sum_{m_1=0}^{M_1} \sum_{m_2=0}^{M_2} \cdots \sum_{m_{k+1}=m_{k+1}}^{M_{k+1}} \hat{h}_k(m_1,m_{k+1}) \times \tilde{u}(n-m_1) \prod_{i=2}^{k+1} |\tilde{u}(n-m_i)|^2.$$  \tag{11}

This way, the $k^{th}$-order polynomials have up to $k+1$ squared envelope terms with distinct delays (cross-samples) and their maximum polynomial dimension is $k+1$. The required number of coefficients in Equation (11) is:

$$N_{\text{RV}_1} = \sum_{k=1}^K (M_k + 1) \left(\frac{M_k + 1}{k + 1}\right),$$  \tag{12}

where $\left(\frac{M_k + 1}{k + 1}\right)$ is the multiset binomial coefficient and is equal to $\left(\frac{M_k + 1}{k + 1}\right)$.

Next, in step 2, for each NL order $k$ greater than or equal to three, the $k^{th}$-order polynomials from Equation (11) are limited to have polynomial dimension up to $L_k + 1$, or, equivalently, $L_k$ distinct powered envelope terms, where the parameter $L_k$ is user-defined in the range from 1 to $k+1$. Thus, once $L_k$ is chosen, the model only keeps polynomials with up to $L_k$ distinct powered envelope terms. For example, let us consider the $7^{th}$-order polynomials $u(n) |u(n)|^6$, $u(n) |u(n-1)|^6$, $u(n) |u(n-1)|^4 |u(n-2)|^2$ and $u(n) |u(n-1)|^2 |u(n-2)|^2 |u(n-3)|^2$ that have, respectively, polynomial dimensions one, two, three and four. Now, if $L_k$ (with $k=7$) is set to two, the last polynomial is discarded from the model.

Let the multiset $M^{(k,l)}_x = (m_2, m_3, \ldots, m_{l+1})$ contain the sample delay indices of the polynomials $(k, l)$, i.e., the
$k$th-order polynomials with exactly $l$ distinct powered envelope terms. In this case, the $l$ sample delays in $M^{(k,l)}$ are not repeated, that is, $m_{l+1} > m_l > \cdots > m_3 > m_2$. Furthermore, let the vector $v^{(k,l)}$ contain the exponents of the powered envelope terms in the polynomials $(k, l)$. Table 1 shows the exponents $v^{(k,l)}$ for $k$ up to 13 and $l = 1, \ldots, L_k$ ($L_k$ maximum equals $\frac{k+1}{2}$), as well as the corresponding polynomial terms. In practice, to avoid redundancy in the model, we do not consider all exponent vectors from Table 1, nor their permutations. In fact, we only consider the first exponent vector of each $(k, l)$ tuple from the table in the model, that is, we apply only the highest possible exponents to the least delayed samples, inspired by the DDR [9] that also controls the power orders of the delayed samples. Finally, the proposed reduced-Volterra model $RV_2$ is given by:

$$
\hat{x}_{RV_2}(n) = \sum_{k=1}^{K} \sum_{\substack{M_k \geq 0 \atop L_k \geq 0}} L_k \sum_{m_{l+1}} \sum_{m_l} \cdots \sum_{m_2} \sum_{m_1} h_k(m_{l+1}) 
\times \hat{u}(n-m_l) \prod_{i=2}^{l+1} |\hat{u}(n-m_i)|^{v^{(k,l)}(i-1)},
$$

(13)

where $v^{(k,l)}$ are exponent vectors given in Table 1 (as discussed above) and the required total number of coefficients is:

$$
N_{RV_2} = \sum_{k=1}^{K} (M_k + 1) \left(\left(\frac{M_k + 1}{L_k}\right)\right),
$$

(14)

where $L_1 = 0$ and $\left(\frac{M_k + 1}{L_k}\right) = \left(\frac{M_l + L_k}{L_k}\right)$.

## IV. PROPOSED SPARSE-GROUP LASSO SOLUTION

This section proposes to generate the RV$_2$ model with a block-wise modular structure, so that group-sparse estimation techniques can be applied for model sizing and reduction. This section is motivated by the reduction in running cost that is expected by the use of a block-modular structure and group-sparse estimation, compared to LASSO, due to the reuse of previously computed blocks, as verified numerically in Section V.

Let us consider the RV$_2$ model in Equation (13), whose parameters are $K$ and the sets $\{M_k\}$ and $\{L_k\}$ whose values are for $k = 1, \ldots, K$ (k odd). Given $N$ samples of the measured training signals, the model’s regression matrix $\Phi$ can be assembled modularly by concatenating sub-matrices $\Phi^{(k)}$ that contain the $k$th-order model regressors. In turn, each sub-matrix $\Phi^{(k)}$ is assembled by concatenating the disjoint blocks $\Phi^{(k,l)}$, for $l = 1, \ldots, L_k$, that is:

$$
\Phi = \begin{bmatrix}
\Phi^{(1)} & \Phi^{(3)} & \cdots & \Phi^{(k)} & \cdots & \Phi^{(K)}
\end{bmatrix},
$$

(15)

where each $k$th-order sub-matrix is given by:

$$
\Phi^{(k)} = \begin{bmatrix}
\Phi^{(k,1)} & \cdots & \Phi^{(k,l)} & \cdots & \Phi^{(k,L_k)}
\end{bmatrix},
$$

(16)

as illustrated in Figure 1. Each block $(k, l)$ is composed of row vectors $\Phi^{(k,l)}(n)$ of the regressors at instant $n$, i.e.,

$$
\Phi^{(k,l)} = \begin{bmatrix}
\Phi^{(k,l)}(0) \\
\vdots \\
\Phi^{(k,l)}(N-1)
\end{bmatrix}.
$$

Note that the size of $\Phi^{(k)}$ is $N \times (M_k + 1) \left(\frac{M_k + 1}{L_k}\right)$, whereas the size of $\Phi^{(k,l)}$ is $N \times (M_k + 1) \left(\frac{M_k + 1}{L_k}\right)$.

A truncated Kronecker operator is defined in [55] and [56] that allows an efficient implementation of triangular Volterra models, such as in Equation (2). Likewise, we next propose the modified truncated Kronecker operator that allows to separately implement each regression block $\Phi^{(k,l)}$ of our model efficiently and avoid redundancy due to sample delay permutations. Let us define the modified truncated Kronecker operator of an $m \times 1$ vector $a = [a_1 a_2 \ldots a_m]^T$ by itself as:

$$
\otimes^q a = a \otimes a = \begin{bmatrix}
a_1 a_2 \\
 a_2 a_3 \\
 \vdots \\
 a_{m-1} a_m
\end{bmatrix},
$$

(17)

where $a_i = [a_i a_{i+1} \ldots a_m]^T$ and $\otimes$ is the modified truncated Kronecker operator. Recursively, the $q^{th}$-order modified truncated Kronecker operator $\otimes^q a \equiv a \otimes \cdots \otimes a$ is given by:

$$
\otimes^q a = \begin{bmatrix}
a_1 \otimes^{q-1} a_2 \\
 a_2 \otimes^{q-1} a_3 \\
 \vdots \\
 a_{m-q+1} \otimes^{q-1} a_{m-q+2}
\end{bmatrix},
$$

(18)

where $\otimes^0 a = 1$ and $\otimes^1 a = a$.

For the RV$_2$ model, we first define the input squared envelope vector, adjusted to the length $M_k + 1$ for each NL order $k$, as follows:

$$
\bar{u}^{(2)}(n) = \begin{bmatrix} |\hat{u}(n)|^2 & |\hat{u}(n-1)|^2 & \cdots & |\hat{u}(n-M_k)|^2 \end{bmatrix}^T,
$$

(19)

then, the modified truncated Kronecker product of $\bar{u}^{(2)}(n)$ by itself is given by:

$$
\otimes^2 \bar{u}^{(2)}(n) = \begin{bmatrix} |\hat{u}(n)|^2 \bar{u}^{(2)}(n-1) \\
|\hat{u}(n-1)|^2 \bar{u}^{(2)}(n-2) \\
\vdots \\
|\hat{u}(n-M_k+1)|^2 \bar{u}^{(2)}(n-M_k)
\end{bmatrix},
$$

(20)

where:

$$
\bar{u}^{(2)}(n-i) = \begin{bmatrix} |\hat{u}(n-i)|^2 & |\hat{u}(n-i-1)|^2 & \cdots & |\hat{u}(n-M_k)|^2 \end{bmatrix}^T
$$

(21)
and, recursively, the corresponding $q^{th}$-order modified truncated Kronecker product results:

\[
\bigotimes_q \tilde{u}^{(2)}(n) = \begin{bmatrix}
|\tilde{u}(n)|^2 \bigotimes_{q-1} \tilde{u}^{(2)}(n-1) \\
|\tilde{u}(n-1)|^2 \bigotimes_{q-1} \tilde{u}^{(2)}(n-2) \\
\ddots \\
|\tilde{u}(n-M_k+q-1)|^2 \bigotimes_{q-1} \tilde{u}^{(2)}(n-M_k+q-2)
\end{bmatrix}.
\] (22)

Therefore, the row vectors $\phi^{(k,l)}(n)$ of blocks $\Phi^{(k,l)}$ $(k \geq 3)$ of the proposed model can be efficiently computed as:

\[
\phi^{(k,l)}(n) = \tilde{u}(n) \otimes \left[ \tilde{u}^{(k+1-2l)}(n) \bigotimes_{q-1} \tilde{u}^{(2)}(n) \right],
\] (23)

where $\otimes$ is the conventional Kronecker product. In Equation (23), since the exponent $(k+1-2l)$ is even, $\tilde{u}^{(k+1-2l)}(n)$ is an element-wise powered version of $\tilde{u}^{(2)}(n)$, thus:

\[
\tilde{u}^{(k+1-2l)}(n) \bigotimes_{q-1} \tilde{u}^{(2)}(n-1) = \begin{bmatrix}
|\tilde{u}(n)|^{k+1-2l} \bigotimes_{q-1} \tilde{u}^{(2)}(n) \\
|\tilde{u}(n-1)|^{k+1-2l} \bigotimes_{q-1} \tilde{u}^{(2)}(n-1) \\
\vdots \\
|\tilde{u}(n-M_k+q-1)|^{k+1-2l} \bigotimes_{q-1} \tilde{u}^{(2)}(n-M_k+q-2)
\end{bmatrix}. \tag{24}
\]

Note that $\bigotimes_0 \tilde{u}^{(2)}(n) = 1$, $\bigotimes_1 \tilde{u}^{(2)}(n) = \tilde{u}^{(2)}(n)$ and $\phi^{(1,0)}(n) = \tilde{u}(n)$.

Finally, before applying group-sparse techniques, each $RV_2$ model block $\Phi^{(k,l)}$ is further sub-divided into $M_k + 1$ non-overlapping sub-blocks, through the sample delay index $m = m_1$ in Equation (13), i.e., the elements of $\tilde{u}(n)$ in Equation (23). The length of each sub-block of $\Phi^{(k,l)}$ is given by:

\[
d_{sb}^{(k,l)} = \left( M_k + 1 \right) \tag{25}
\]

and corresponds to $d_g$ values in Equation (7).

### TABLE 1. Exponent vectors for the proposed $RV_2$ model.

| $k$ | $l$ | $u^{(k,l)}$ | Polynomials |
|-----|-----|-------------|-------------|
| 1   | 0   | -           | $u(n-m_1)$  |
| 3   | 1   | [2]         | $u(n-m_1)^2$ |
| 5   | 1   | [4]         | $u(n-m_2)^4$ |
| 2   |     | [2,2]       | $u(n-m_2)^2 u(n-m_3)^2$ |
| 7   | 1   | [6]         | $u(n-m_1)^6$ |
| 2   |     | [4,2]       | $u(n-m_2)^4 u(n-m_3)^2$ |
| 3   |     | [2,2,2]     | $u(n-m_1) u(n-m_2)^2 \ldots u(n-m_4)^2$ |
| 9   | 1   | [8]         | $u(n-m_1)^8$ |
| 2   |     | [6,2]       | $u(n-m_2)^6 u(n-m_3)^4$ |
| 3   |     | [4,2,2]     | $u(n-m_1) u(n-m_2)^4 \ldots u(n-m_5)^2$ |
| 4   |     | [2,2,2,2]   | $u(n) u(n-m_2)^2 \ldots u(n-m_6)^2$ |
| 11  | 1   | [10]        | $u(n-m_1)^{10}$ |
| 2   |     | [8,2]       | $u(n-m_2)^8 u(n-m_3)^2$ |
| 3   |     | [6,2,2]     | $u(n-m_2)^6 u(n-m_3)^4 \ldots u(n-m_5)^2$ |
| 4   |     | [4,2,2,2]   | $u(n-m_1) u(n-m_2)^4 \ldots u(n-m_6)^2$ |
| 5   |     | [2,2,2,2,2] | $u(n-m_1) u(n-m_2)^2 \ldots u(n-m_7)^2$ |
This way, using the bi-level SGL selection we can perform the sizing of the model in a systematic way, by convex optimization, obtaining the active set of sub-blocks \((k, l, m)\) and the indexes of the most relevant regressors within the selected sub-blocks. In the next section, we discuss the numerical/experimental results.

V. NUMERICAL VALIDATION

The numerical validation of the proposed \(RV_2\) DPD model in Equation (13) is performed by comparison to well-known models, such as the triangular Volterra, MP, GMP and others. This task consists of an initial proof of concept performed through computer simulations, followed by an experimental test using hardware that validates the performance of the model.

In both steps, the PA input and output signals are generated, simulated/measured and stored in separate datasets for training and validation. The ability of the proposed DPD technique to compensate for in-band and out-of-band PA distortions and its performance are evaluated using the validation dataset. The DPD training algorithms are implemented offline using Matlab and, additionally, LASSO and extensions use the package Sparse Learning with Efficient Projections (SLEP) [57], also in Matlab, although there are other implementations available, e.g., [58] and [59]. Moreover, during sparse estimation, the LASSO algorithm is run for a range of pre-defined values of the sparsity parameter \(\lambda\), using the SLEP package with up to 5,000 iterations per run and tolerance \(1e^{-7}\). The best \(\lambda\) value is obtained using the Akaike's information criterion (AIC) [49]:

\[
AIC^{(\lambda_j)} = N \ln \left( \frac{\| \hat{\mathbf{y}} - \mathbf{\Phi} (\lambda_j) \|_2}{\lambda} \right) + 2 \| \hat{\mathbf{\beta}} (\lambda_j) \|_0,
\]

where \(\| \hat{\mathbf{\beta}} (\lambda_j) \|_0\) is the \(\ell_0\) pseudo-norm of the coefficient vector from LASSO with candidate \(\lambda_j\). The AIC is a widely used selection criterion for choosing the best among several competing models, trading-off their fitting capabilities and required running complexity [60]. Thus, we run the optimization in Equation (7) for a predefined range of \(\lambda\) values and choose the \(\lambda\) that minimizes the AIC.

As is discussed in the next subsections, the main advantages of the proposed model, comparing to conventional models from the literature, are: flexibility, i.e., parameter values can be independently defined for each NL order; generality, i.e., higher polynomial dimensions can be added to the model as required; modularity, i.e., its block-modular structure allows group-wise sparse estimation (further reducing the running cost, as in Subsection V-C) and parsimony, i.e., the curse of dimensionality of Volterra-based models is avoided.

A. SIMULATION RESULTS

In this subsection, we carry out simple computer simulations to evaluate the performance of the proposed DPD model. For this, we describe the PA in the complex baseband domain by the well-known Wiener-Hammerstein (WH) model [11], [12]. As in Figure 2, the WH is a block-oriented model consisting of the series interconnections of a linear time-invariant (LTI) filter, a memoryless nonlinear (NL) block and another LTI filter. In the literature, the NL block is described by memoryless models, such as complex polynomials, nonlinear functions, the Saleh [61] and Rapp [62] models, or directly by amplitude-to-amplitude modulation (AM/AM) and amplitude-to-phase modulation (AM/PM) look-up tables. In our simulations, we account for the NL block through three different options, as follows:

- The 5th order memoryless polynomial given in [47], with coefficients: \(\beta_1 = 14.9740 + j0.0519\), \(\beta_2 = -23.0954 + j4.9680\) and \(\beta_3 = 21.3936 + j0.4305\), extracted from a Class AB PA.
- The arctan model in [63], as follows:

\[
\hat{v}(n) = (\gamma_1 \arctan(\xi_1|\tilde{v}(n)|^2) + \gamma_2 \arctan(\xi_2|\tilde{v}(n)|^2))
\]

\[
e^{\phi|\tilde{v}(n)|^2},
\]

where \(\gamma_1 = 8.00335 + j4.61157\), \(\gamma_2 = -3.77167 + j12.03758\), \(\xi_1 = 2.26895\) and \(\xi_2 = 0.8234\).
- The Saleh model [61], with parameter values from [63]: \(a_a = 2\), \(b_a = 2.2\), \(a_\phi = 2\) and \(b_\phi = 1\).

The LTI blocks in the WH model can be either finite impulse response (FIR) or infinite impulse response (IIR) filters. We adopt the transfer functions from [3], i.e., \(H_1(z) = (1 + 0.25z^{-2})/(1 + 0.4z^{-1})\) for the input filter and \(H_0(z) = (1 - 0.1z^{-1})/(1 - 0.2z^{-1})\) for the output filter, to be used in combination with any of the NL block options above. In the simulations, the input signal is a single-carrier 64-quadrature amplitude modulation (64-QAM) waveform. In each case, the input power level is adjusted to drive the PA model into

\[
\begin{array}{cccccc}
\Phi^{(1)} & \Phi^{(3)} & \Phi^{(5)} & \Phi^{(7)} & \Phi^{(9)} \\
\Phi^{(5,1)} & \Phi^{(5,2)} & \Phi^{(7,1)} & \Phi^{(7,2)} & \Phi^{(7,3)} \\
\Phi^{(7,3,1)} & \Phi^{(7,3,2)} & \Phi^{(9,1)} & \Phi^{(9,2)} & \Phi^{(9,3)} \\
\Phi^{(9,3)} & \Phi^{(9,4)} & \Phi^{(11)} & \Phi^{(13)} \\
\Phi^{(11,1)} & \Phi^{(11,2)} & \Phi^{(11,3)} & \Phi^{(11,4)} & \Phi^{(11,5)} \\
\Phi^{(11,5)} & \Phi^{(13,1)} & \Phi^{(13,2)} & \Phi^{(13,3)} & \Phi^{(13,4)} \\
\Phi^{(13,4)} & \Phi^{(13,5)} & \Phi^{(13,6)} & \Phi^{(13,6)} & \\
\end{array}
\]

FIGURE 1. Proposed block structure of the \(RV_2\) model (up to \(k=13\)).
the nonlinear range. Tables 2, 3 and 4 present the simulation results for the distinct NL options of the WH PA model, respectively, the polynomial, arctan and Saleh, using the DPD models: MP, GMP, dense \( RV_2 \) and sparse \( RV_2 \) LASSO. The DPD performance evaluation is given in terms of the required numbers of coefficients, achieved NMSE and ACPR. The DPD model sizing is performed by exhaustive search with the GMP model, choosing the model parameters based on the NMSE and ACPR achieved at each search grid point defined by \( K, M \) and \( J \) in the ranges \( K = 5, 7, 9; M = 4, 5, 6 \) and \( J = 2, 3, 4, 5 \) and assuming, for simplicity, in Equation (4), that \( M = M_1 = M_2 = M_3, K = K_1 = K_2 = K_3 \) and \( J = J_2 = J_3 \). The same model sizing procedure is performed in Subsection V-B, where it is presented in more details. The \( RV_2 \) parameters follow the ones for the GMP, i.e. same \( K \) value and maximum \( M_k \) equal to \( M + 2J \) (which is the GMP memory span), with lower \( M_k \) values for higher NL orders. The \( RV_2 \) model is run for two sets of \( L_k \) parameters: \( \{0, 1, \ldots , 1\} \) and \( \{0, 1, 2, \ldots , 2\} \), as indicated in each case in Tables 2, 3 and 4. The \( RV_2 \) LASSO estimation is run for \( L_k = \{0, 1, 2, \ldots , 2\} \) and a pre-defined range of the sparsity parameter \( \lambda \), so the best \( \lambda \) value is obtained in each case using the AIC score in Equation (26). This procedure is also discussed in more details in Subsection V-B. From our simulations, we can verify that the proposed model is able to outperform the conventional models under study and, in addition, the \( RV_2 \) LASSO outperforms the GMP model with approximately the same numbers of coefficients. Figures 3, 4 and 5 show the normalized power spectral densities (PSD) of the compensated output signals obtained using the MP, GMP, \( RV_2 \) (\( L_k = \{0, 1, 2, \ldots , 2\} \)) and \( RV_2 \) LASSO DPD models for, respectively, the polynomial, arctan and Saleh NL models with \( P_{\text{avg}} = -14 \text{ dBm} \) (polynomial and Saleh) and \( P_{\text{avg}} = -9 \text{ dBm} \) (arctan). As we can see in the figures, the proposed model can effectively linearise the PA, further reducing the spectral regrowth compared to the conventional models.

### B. EXPERIMENTAL RESULTS

Next, we validate the performance of the proposed DPD model through experimental tests, using a 10 W Gallium Arsenide (GaAs) metal–semiconductor field-effect transistor (MESFET) RF PA with 40 dB of nominal (linear) gain, designed at Laboratory of Microelectronics (LME) of the University of São Paulo (USP), measured in Class AB with the test set-up shown in Figure 6. As detailed in the block diagram, an N5182B MXG vector signal generator from Keysight Technologies is used to synthesize, at the carrier frequency of 5.9 GHz, the PA input signal for this experiment, an orthogonal frequency division multiplexing (OFDM) waveform with 1,024 sub-carriers modulated with 64-QAM and 20 MHz of bandwidth. This range of bandwidth complies with the orthogonal frequency division multiple access (OFDMA) downlink of the long term evolution (LTE) 3GPP standard [64] and is also considered in recent papers, e.g., [2], [15], [16], [20], [65], and [66]. Furthermore, note that memory polynomials in general, of which the proposed model is an expression, are also applicable for DPD of configurations with wider bandwidths, as for example [35]. An N9020B MXA vector signal analyzer from Keysight Technologies is used to digitize and capture the complex
I/Q data streams of the PA input and output waveforms. Both the MXG and MXA are connected to a PC via LAN. The PA input $\tilde{x}$ and output $\tilde{y}$ complex envelope datasets are time aligned by cross-correlation in Matlab and split into the disjoint training and validation partitions, each of which with $N = 40,000$ samples. In the next items, the $RV_2$ model and its sparse versions are compared, in terms of numbers of required coefficients and DPD performance, to well-known behavioural models, such as the GMP and $MV_{\Delta}$, all estimated using the same training dataset. Additionally, we compare the proposed model to the MP, CRV (Equation (5)) and $RV_1$ (Equation (11)). Finally, the $RV_2$ model with block-wise structure and SGL estimation is validated.

1) INPUT POWER LEVEL
Initially, in order to decide an appropriate PA input power level, the PA is driven by the modulated input signal from the MXG with three average power levels: $P_{\text{avg}} = -17$, $-12$ and $-8$ dBm. As the PA input power increases, the amplifier reaches its NL region and shows gain compression, as in Figure 7, where the PA gain is plotted versus the instantaneous power $P_{\text{in}}$ of the input signal. As expected, NL behaviour and memory effects become more pronounced as the PA average power level increases, reaching our region of interest, where spectral regrowth, modulation distortion and nonlinear memory effects are visible. Based on this analysis, $P_{\text{avg}} = -12$ and $-8$ dBm are chosen for the DPD model comparison.

2) GMP MODEL SIZING
In this item, the GMP model sizing is performed by exhaustive search, as follows. A search grid is defined by varying the model parameters $K$, $M$ and $J$ in the ranges $K = 1, \ldots, 11$ ($K$ odd); $M = 1, \ldots, 6$ and $J = 1, \ldots, 3$ and also assuming in Equation (4) that $M = M_1 = M_2 = M_3$, $K = K_1 = K_2 = K_3$ and $J = J_2 = J_3$. Using the measured data for $P_{\text{avg}} = -12$ and $-8$ dBm, the model is estimated at each grid point by least-squares and the corresponding NMSE and ACPR performances are captured. As shown in Figure 8(a) and (b), as the model size increases, the ACPR gradually decreases, until a plateau region is reached, the same being observed for NMSE (not shown). From the plateau onwards, the GMP model cannot provide further significant improvements. By inspection, the model parameters for $P_{\text{avg}} = -12$ and $-8$ dBm are chosen as, respectively, $K = 7, M = 5, J = 2$ and $K = 9, M = 5, J = 3$. 
3) DENSE RV\textsubscript{2} RESULTS

Next, we discuss the RV\textsubscript{2} model results. In this item, the dense RV\textsubscript{2} model is considered, i.e., not exploring sparsity yet, so the RV\textsubscript{2} LASSO and RV\textsubscript{2} SGL models are discussed in the next items. Figures 9 and 10 show, respectively, for $P_{\text{avg}} = -12$ and $-8$ dBm, the residual ACPR after compensation versus the numbers of coefficients obtained with the RV\textsubscript{2} model, having the GMP model as a benchmark. In this analysis, the GMP parameters are run in the ranges $K = 1, \ldots, 11$ (K odd); $M = 1, \ldots, 6$ and $J = 0, \ldots, 3$. The RV\textsubscript{2} model parameters follow the ones for the GMP, i.e. same $K$ value, maximum $M_k$ equal to $M + 2J$ (which is the GMP memory span) and lower $M_k$ values for higher NL orders. The RV\textsubscript{2} model is run with parameters $L_k = \{0, 1, 2, \ldots, 2\}$ and the first exponent vectors from Table 1. For each combination of parameters, the GMP and RV\textsubscript{2} models are estimated by least-squares. In addition, ridge regression is also applied when it is necessary to avoid overfitting the training data. The figures include, for each model size (number of coefficients), the combination of parameters that gives the best performance, i.e., the Pareto front. Comparing the residual performance provided by the models, the RV\textsubscript{2} (dense) model is capable of achieving lower ACPR levels, which is very relevant, since wireless transmitters must comply with spectral emission masks. This is also verified for in-band performance (NMSE) (not shown). However, the performance improvements required much larger model sizes, which impacts the on-line DPD running cost. Tables 5 and 6 summarize, respectively, for $P_{\text{avg}} = -12$ and $-8$ dBm, the DPD performance of the proposed RV\textsubscript{2} model compared to the $MV_\Delta$, $RV_1$, MP, GMP and CRV models. Model parameters adopted in Table 5 are $K = 7$, $M = 5$ and $J = 2$ (GMP), $M_k = \{9, 6, 5\}$ and $L_k = \{0, 1, 2, 2\}$ (RV\textsubscript{2}), whereas in Table 6 the model parameters are $K = 9$, $M = 5$ and $J = 3$ (GMP), $M_k = \{10, 10, 7, 6, 6\}$ and $L_k = \{0, 1, 2, 2, 2\}$ (RV\textsubscript{2}), as indicated therein. The parameters for the MP, CRV and RV\textsubscript{2} models are obtained from the GMP model sizing. For adjusting the RV\textsubscript{2} model sizing, $L_k$ starts at $\{0, 1, \ldots, 1\}$ and is increased until performance improvements are achieved. Please note that we can flexibly include higher dimension and more delayed terms in each NL order of the RV\textsubscript{2} model by adjusting $M_k$ and $L_k$. In Table 6, the size of the $MV_\Delta$ model grows very fast, so we were only able to estimate the model up to $K = 9$ and $M_k = \{10, 10, 6, 4, 3\}$ or 7,499 coefficients. The RV\textsubscript{2} model replaces the $MV_\Delta$ doubly-combinatorial in Equation (2) by Equation (13), reducing substantially the model size, estimation and running costs. In addition, the RV\textsubscript{2} model allows the maximum polynomial dimension to be flexibly defined for each NL order. As a conclusion, the proposed RV\textsubscript{2} model approximates the DPD performance of the $MV_\Delta$, while is more affordable for practical
sparse Flexible Reduced-Volterra Model for Power Amplifier Digital Pre-Distortion

C. S. Hemsi, C. M. Panazio: Sparse Flexible Reduced-Volterra Model for Power Amplifier Digital Pre-Distortion

FIGURE 8. MP/GMP model sizing (search grid): (a) $P_{\text{avg}} = -12$ dBm and (b) $P_{\text{avg}} = -8$ dBm.

imputation. The MP model is a subset of the GMP, whereas the CRV is an extended version, in which a larger amount of two-dimensional polynomials is allowed. Comparing to the MP, GMP and CRV models, the dense $RV_2$ model offers 2 to 3 dB superior performance at the cost of higher complexity, which is addressed by the sparse techniques considered next.

FIGURE 9. ACPR versus model size ($P_{\text{avg}} = -12$ dBm).

4) $RV_2$ LASSO RESULTS

In this item, the LASSO sparse estimation from Equation (7) is applied to the dense $RV_2$ model, in order to reduce its complexity and DPD running cost. For $P_{\text{avg}} = -12$ and $-8$ dBm, the previously assembled $RV_2$ regression matrix $\Phi$ is standardized. In Figures 9 and 10, we run $RV_2$ LASSO with a predefined range of the sparsity parameter $\lambda$ and visualize the ACPR values achieved versus the number of coefficients. In the figures, the ACPR performance with the $RV_2$ LASSO is very close to that of the dense $RV_2$, with the advantage of requiring a reduced complexity. Moreover, Figure 11 shows, for $P_{\text{avg}} = -12$ dBm, the NMSE and AIC values achieved with the $RV_2$ LASSO best fit solutions for a predefined range of $\lambda$ values, indicated in the figure. The best $\lambda$ value, where a good trade-off between accuracy and complexity is achieved, corresponds to the minimum AIC score in the curve. Note that by making $\lambda$ too large, LASSO is not able to capture the underlying behaviour of the system. As $\lambda$ is swept, the elbow in the NMSE curve indicates a range of economic solutions for which very small degradations of the model performance are observed. Tables 5 and 6 also present sparse estimation results, as follows. We compare the $RV_2$ LASSO and $RV_2$ SGL run for two sets of sparsity parameters, the one that corresponds to the best AIC scores and another that implies a more reduced model complexity. The $RV_2$ LASSO and $RV_2$ SGL models outperform the GMP and CRV models, as well as the sparse selection of regressors based on the greedy OMP [34], [35], [67], as in the tables. The LASSO sparse estimation technique preserves the model’s performance, while significantly reducing its running cost. In the next item, we discuss the $RV_2$ SGL model results.

FIGURE 10. ACPR versus model size ($P_{\text{avg}} = -8$ dBm).
TABLE 5. DPD performance with measured PA and $P_{\text{avg}} = -12$ dBm.

| Model (Parameters) | Num. of coeffs. | Max. ACPR (dBc) | NMSE (dB) |
|--------------------|-----------------|-----------------|-----------|
| No DPD             |                 | -33.5           | -1.7      |
| MP ($K = 7, M = 7$) | 32              | -47.3           | -37.7     |
| GMP ($K = 7, M = 7, J = 5$) | 96         | -48.6           | -38.1     |
| CRV ($K = 7, M = 7, Q = 5$) | 148        | -49.8           | -38.2     |
| $M_{\text{VAR}}$ ($K = 7, M_k = \{9, 9, 6, 4\}$) | 5,362 | -52.8           | -38.9     |
| $RV_1$ ($K = 7, M_k = \{9, 9, 6, 5\}$, $L_d = \{0, 1, 2, 3\}$) | 552   | -52.6           | -38.7     |
| $RV_2$ ($K = 7, M_k = \{9, 9, 6, 5\}$, $L_d = \{0, 1, 2, 2\}$) | 432   | -52.0           | -38.6     |
| $RV_2$ OMP ($N_{\text{OMP}} = 82$) | 82    | -50.2           | -38.3     |
| $RV_2$ OMP ($N_{\text{OMP}} = 38$) | 38    | -46.0           | -37.4     |
| $RV_2$ LASSO ($\lambda_1 = 4.5$) | 82    | -51.3           | -38.4     |
| $RV_2$ SGL ($\lambda_1 = 6, \lambda_2 = 7$) | 77    | -51.3           | -38.6     |
| $RV_2$ LASSO ($\lambda_1 = 9$) | 38    | -49.3           | -38.1     |
| $RV_2$ SGL ($\lambda_1 = 12, \lambda_2 = 16$) | 44    | -49.3           | -38.1     |

TABLE 6. DPD performance with measured PA and $P_{\text{avg}} = -8$ dBm.

| Model (Parameters) | Num. of coeffs. | Max. ACPR (dBc) | NMSE (dB) |
|--------------------|-----------------|-----------------|-----------|
| No DPD             |                 | -39.4           | -1.4      |
| MP ($K = 9, M = 8$) | 45              | -47.8           | -37.6     |
| GMP ($K = 9, M = 5, J = 3$) | 174        | -49.2           | -38.2     |
| CRV ($K = 9, M = 8, Q = 5$) | 212        | -50.5           | -38.4     |
| $M_{\text{VAR}}$ ($K = 9, M_k = \{10, 10, 7, 6, 6\}$, $L_d = \{0, 1, 2, 3, 4\}$) | 7,499  | -53.3           | -38.9     |
| $RV_1$ ($K = 9, M_k = \{10, 10, 7, 6, 6\}$, $L_d = \{0, 1, 2, 2, 2\}$) | 951   | -52.8           | -38.9     |
| $RV_2$ OMP ($N_{\text{OMP}} = 132$) | 621  | -52.7           | -38.8     |
| $RV_2$ OMP ($N_{\text{OMP}} = 62$) | 132  | -52.7           | -38.8     |
| $RV_2$ LASSO ($\lambda_1 = 4$) | 132  | -51.8           | -38.6     |
| $RV_2$ SGL ($\lambda_1 = 14, \lambda_2 = 22$) | 142  | -51.8           | -38.7     |
| $RV_2$ LASSO ($\lambda_1 = 13$) | 62   | -49.3           | -38.2     |
| $RV_2$ SGL ($\lambda_1 = 40, \lambda_2 = 34$) | 60   | -48.8           | -38.0     |

FIGURE 11. NMSE and AIC versus model size ($RV_2$ LASSO).

5) $RV_2$ SGL RESULTS

Next, the bi-level SGL sparse estimation from Equation (8) is applied to the original $RV_2$ model. The SGL estimation is run for a pre-defined search grid of the sparsity parameters ($\lambda_1, \lambda_2$) using the SLEP package [57] with tolerance $1e^{-6}$ and the best values of ($\lambda_1, \lambda_2$) are found through the AIC score. Note that this task becomes more complex compared to LASSO, due to the additional SGL parameter. As already mentioned, the SGL criteria performs the sizing of the original model, selecting the sparse set of the most relevant subblocks, as well as the dominant regressors. Tables 5 and 6 show that both the $RV_2$ LASSO and $RV_2$ SGL models outperform the ACPR of the GMP model by 3 dB for similar DPD running costs (at the expense of a more costly estimation). For $P_{\text{avg}} = -12$ and $-8$ dBm, the active sets selected by SGL resulted in, respectively, 11 out of 37 and 18 out of 53 sub-blocks. Figure 12 illustrates for $P_{\text{avg}} = -12$ dBm the normalized PSD of the compensated output signals using the GMP, dense $RV_2$, $RV_2$ LASSO and $RV_2$ SGL models. From Table 5 and Figure 12, the sparse versions of the proposed model achieved better ACPR and NMSE performance than the GMP model for the same or lower numbers of coefficients. Moreover, Figure 13 (a) and (b) presents, respectively, the AM-AM gain (in dB) and AM-PM plots and we can see that the techniques are able to compensate for the nonlinearities. This way, the large size of the dense $RV_2$ model is addressed using sparse techniques, leading to a more affordable model for practical implementation. Note that the model extraction using LASSO and SGL is more costly than with ordinary least squares, but is performed offline. In the next subsection we compare the running complexity of the GMP, $RV_2$ LASSO and $RV_2$ SGL models.

C. COMPLEXITY ANALYSIS

In this subsection, we consider the DPD computational complexity of the models, particularly, the running cost. As in [68], the total DPD complexity can be classified into estimation, adaptation and running costs. The estimation and adaptation costs refer, respectively, to DPD training and updating tasks, both performed offline in a digital signal processor (DSP). The running cost refers to the real-time
tasks of regression matrix generation and DPD filtering, performed in hardware. In [68], expressions for the running cost in floating point operations (FLOPs) per sample are developed for several DPD behavioural models, based on FLOPs required per operation, such as amplitude squared, complex-complex and complex-real multiplications, complex addition, and so on. This way, the GMP model running cost expressions (in FLOPs per sample) are:

(a) regressors generation: $3 + (2J + 1)(K - 1)$;
(b) DPD filtering: $8N_{gmp} - 2$, where $N_{gmp}$ is the number of GMP coefficients. Next, we derive running cost expressions in FLOPs per sample for the RV2 LASSO and RV2 SGL models, as follows:

- In both cases, DPD filtering cost: $8N_{model} - 2$.
- RV2 LASSO regressors generation cost:
  \[
  3 + \sum_{k=1}^{K} \frac{k + 1}{2} N_{lasso}^{(k)},
  \]
  where $N_{lasso}^{(k)}$ is the number of $k^{th}$-order polynomials. Note that each $k^{th}$-order regressor involves $\frac{k-3}{2}$ real multiplications plus one complex-real multiplication.
- RV2 SGL regressors generation cost:
  \[
  3 + \sum_{k=1}^{K} \sum_{l=1}^{L_k} ((2N_{sb}^{(k,l)} + 1)d_{sb}^{(k,l-1)} + 2d_{sb}^{(k,l)}),
  \]
  where $N_{sb}^{(k,l)}$ is the number of selected sub-blocks of a block $(k,l)$ and $d_{sb}^{(k,l)}$ is the length of each of these sub-blocks. In the proposed modular structure, all sub-blocks of a given block can reuse the same computations stored in memory and the computation of higher order sub-blocks partially reuse previously calculated ones.

Table 7 compares the running complexities in FLOPs per sample of the GMP, RV2 LASSO and RV2 SGL models. We can conclude that even the models RV2 LASSO and RV2 SGL having approximately the same number of coefficients, the proposed block-modular structure is more advantageous for on-line DPD implementation, further reducing the required running cost.

VI. CONCLUSION
This paper proposed a flexible reduced-Volterra polynomial model that extends existing memory polynomial models, such as the MP and GMP, by including higher dimension terms in a controlled way, being applicable to a wide range of PA and DPD. In our experimental validation, the sparse versions of the proposed model delivered better in-band and out-of-band performance than conventional models for the same or lower DPD running cost. The main advantages of the proposed model, compared to conventional models from the literature, are: flexibility, i.e., parameter values can be independently defined for each NL order; generality, i.e., higher polynomial dimensions can be added to the model as required; modularity, i.e., its block-modular structure allows group-wise sparse estimation (further reducing the running cost, as in Subsection V-C) and parsimony, i.e., the curse of dimensionality of Volterra-based models is avoided. Future works consider a generalization of the proposed model for the joint compensation of NL PA and in-phase and quadrature (I/Q) modulator imbalance and for beamforming and phased-array.
The authors would like thank the Laboratory of Microelectronics (LME), University of São Paulo (USP), for supporting the experiments.

REFERENCES

[1] M. Schetzen, The Volterra and Wiener Theories of Nonlinear Systems, Melbourne, FL, USA: Krieger 2006.
[2] C. Crespo-Cadenas, M. J. Madero-Ayora, J. Reina-Tosina, and J. A. Becerra-González, “Formal deduction of a Volterra series model for complex-valued systems,” Signal Process., vol. 131, pp. 245–248, Feb. 2017.
[3] L. Ding, G. T. Zhou, D. R. Morgan, Z. Ma, J. S. Kenney, J. Kim, and C. R. Giardina, “A robust digital baseband predistorter constructed using memory polynomials,” IEEE Trans. Commun., vol. 52, no. 1, pp. 159–165, Jan. 2004.
[4] B. M. Lee and R. J. P. de Figueiredo, “Adaptive predistorters for linearization of high-power amplifiers in OFDM wireless communications,” Circuits, Syst. Signal Process., vol. 25, no. 1, pp. 59–80, Feb. 2006.
[5] C. D. Presti, D. F. Kimball, and P. M. Asbeck, “Closed-loop digital predistortion system with fast real-time adaptation applied to a handset WCDMA PA module,” IEEE Trans. Microw. Theory Techn., vol. 60, no. 3, pp. 604–618, Mar. 2012.
[6] G. Norris, J. Staudinger, J.-H. Chen, C. Rey, P. Pratt, R. Sherman, and H. Fraiz, “Application of digital adaptive pre-distortion to mobile wire- less devices,” in Proc. IEEE Radio Freq. Integr. Circuits (RFIC) Symp., Jun. 2007, pp. 247–250.
[7] J. Kim and K. Konstantinou, “Digital predistortion of wideband signals based on power amplifier model with memory,” Electron. Lett., vol. 37, no. 23, pp. 1417–1418, Nov. 2001.
[8] D. R. Morgan, Z. Ma, J. Kim, M. G. Zierdt, and J. Pastalan, “A gener- alized memory polynomial model for digital predistortion of RF power amplifiers,” IEEE Trans. Signal Process., vol. 54, no. 10, pp. 3852–3860, Oct. 2006.
[9] A. Zha, J. Dooley, and T. Brazil, “Simplified Volterra series based behav- ioral modeling of RF power amplifiers using deviation-reduction,” in IEEE MTT-S Int. Microw. Symp. Dig., Jun. 2006, pp. 1113–1116.
[10] A. Zha, J. C. Pedro, and T. J. Brazil, “Dynamic deviation reduction- based Volterra behavioral modeling of RF power amplifiers,” IEEE Trans. Microw. Theory Techn., vol. 54, no. 12, pp. 4323–4332, Dec. 2006.
[11] J. C. Pedro and S. A. Maas, “A comparative overview of microwave and wireless power- amplifier behavioral modeling approaches,” IEEE Trans. Microw. Theory Techn., vol. 53, no. 4, pp. 1150–1163, Apr. 2005.
[12] M. Isaksen, D. Witzig, and D. Rönnow, “A comparative analysis of behavioral models for RF power amplifiers,” IEEE Trans. Microw. Theory Techn., vol. 54, no. 1, pp. 348–359, Jan. 2006.
[13] F. M. Ghanouchi and O. Hammi, “Behavioral modeling and predistor- tion,” IEEE Microw. Mag., vol. 10, no. 7, pp. 52–64, Dec. 2009.
[14] C. Crespo-Cadenas, J. Reina-Tosina, M. J. Madero-Ayora, and J. Muñoz-Cruzado, “A new approach to pruning Volterra models for power amplifiers,” IEEE Trans. Signal Process., vol. 58, no. 4, pp. 2113–2120, Apr. 2010.
[15] C. Crespo-Cadenas, M. J. Madero-Ayora, and J. A. Becerra, “Upgrading behavioral models for the design of digital predistorters,” Sensors, vol. 21, no. 16, p. 5350, Aug. 2021.
[16] L. Schuartz, E. L. Santos, B. Leite, A. A. Mariano, and E. G. Lima, “Reduced-complexity polynomials with memory applied to the lineariza- tion of power amplifiers with real-time discrete gain control,” Circuits, Syst., Signal Process., vol. 38, pp. 3901–3930, Feb. 2019.
[17] C.-F. Cheang, P.-I. Mak, and R. P. Martins, “A hardware-efficient feedback polynomial topology for DPD linearization of power amplifiers: Theory and FPGA validation,” IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 65, no. 9, pp. 2889–2902, Sep. 2018.
[18] F. Mkadem, M. C. Fares, S. Boumaiza, and J. Wood, “Complexity- reduced Volterra series model for power amplifier digital predistor- tion,” Anal. Integ. Circuits Signal Process., vol. 79, no. 2, pp. 331–343, May 2014.
[19] M. Li, Z. Yang, Z. Zhang, R. Li, Q. Dong, and S. Nakatake, “Sparsity adaptive estimation of memory polynomial based models for power ampli- fier behavioral modeling,” IEEE Microw. Wireless Compon. Lett., vol. 26, no. 5, pp. 370–372, May 2016.
Cyro S. Hemi received the B.S. degree in electrical engineering from the State University of Campinas, Campinas, Brazil, in 1995, and the M.Sc. and Ph.D. degrees in telecommunications from the Polytechnic School, University of São Paulo, São Paulo, Brazil, in 2001 and 2021, respectively. He has worked for many years as an RF/Microwave Application Engineer at Keysight Technologies and previously as a Telecom Engineer at Ericsson, ZTE, and Promon. His research interests include signal processing, wireless communications, machine learning, RF power amplifiers, and test and measurement.

Cristiano M. Panazio (Member, IEEE) received the B.S. and M.Sc. degrees in electrical engineering from the State University of Campinas, Campinas, Brazil, in 1999 and 2001, respectively, and the Ph.D. degree in telecommunications from Conservatoire National des Arts et Métiers, Paris, France, in 2005. He is currently an Assistant Professor with the Laboratory of Communications and Signals, Polytechnic School, University of São Paulo, São Paulo, Brazil. He has served as Associate Editor for the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—I: REGULAR PAPERS, from 2016 to 2018. His research interests include signal processing, wireless communications, multichannel modulation, and massive MIMO.

* * *

[42] S. Benedetto, E. Biglieri, and R. Daffara, “Modeling and performance evaluation of nonlinear satellite links—A Volterra series approach,” IEEE Trans. Aerosp. Electron. Syst., vol. AES-15, no. 4, pp. 494–507, Jul. 1979.

[43] R. Raich and G. T. Zhou, “On the modeling of memory nonlinear effects of power amplifiers for communication applications,” in Proc. IEEE 10th Digit. Signal Process. Workshop, 2nd Signal Process. Educ. Workshop, Oct. 2002, pp. 7–10.

[44] J. Staudinger, J.-C. Nanan, and J. Wood, “Memory fading Volterra series model for high power infrastructure amplifiers,” in Proc. IEEE Radio Wireless Symp. (RWS), Jan. 2010, pp. 184–187.

[45] C. Eun and E. J. Powers, “A new Volterra predistorter based on the indirect learning architecture,” IEEE Trans. Signal Process., vol. 45, no. 1, pp. 223–227, Jan. 1997.

[46] E. Ngoya and S. Mons, “Progress for behavioral challenges: A summary of time-domain behavioral modeling of RF and microwave subsystems,” IEEE Microw. Mag., vol. 15, no. 6, pp. 91–105, Sep. 2014.

[47] J. N. Franklin, “On Tikhonov's method for ill-posed problems,” Math. Comput., vol. 28, no. 128, pp. 889–907, 1974.

[48] H. Akaike, “A new look at the statistical model identification,” IEEE Trans. Aerosp. Electron. Syst., vol. AES-15, no. 4, pp. 494–507, Jul. 1979.

[49] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani, “Least angle regression,” Ann. Statist., vol. 32, no. 2, pp. 407–499, 2004.

[50] J. Liu, S. Ji, and J. Ye, SLEP: Sparse Learning With Efficient Projections, IEEE Trans. Veh. Technol., vol. 53, no. 5, pp. 1468–1479, Sep. 2004.

[51] J. Yuan and Y. Lin, “Model selection and estimation in regression with grouped variables,” J. Roy. Statist. Soc. B, vol. 68, no. 1, pp. 49–67, Feb. 2006.

[52] K. P. Girish and S. J. John, “Multiset topologies induced by multiset relations,” Inf. Sci., vol. 188, pp. 298–313, Apr. 2012.

[53] D. Singh, I. A. Musa, Y. Tellia, and J. Singh, “An overview of the applications of multisets,” Novi Sad J. Math., vol. 37, pp. 73–92, Jan. 2007.

[54] C. A. R. Fernandes, J. C. M. Mota, and G. Favier, “MIMO Volterra modeling for nonlinear communication channels,” Learn. Nonlinear Models, vol. 8, no. 2, pp. 71–92, 2010.

[55] J. Liu, S. J. Ji, and J. Ye, SLEP: Sparse Learning With Efficient Projections, Tempe, AZ, USA: Arizona State Univ., 2009.

[56] CRISTIANO M. PANAZIO (Member, IEEE) received the B.S. and M.Sc. degrees in electrical engineering from the State University of Campinas, Campinas, Brazil, in 1999 and 2001, respectively, and the Ph.D. degree in telecommunications from Conservatoire National des Arts et Métiers, Paris, France, in 2005. He is currently an Assistant Professor with the Laboratory of Communications and Signals, Polytechnic School, University of São Paulo, São Paulo, Brazil. He has served as Associate Editor for the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—I: REGULAR PAPERS, from 2016 to 2018. His research interests include signal processing, wireless communications, multichannel modulation, and massive MIMO.