electronics

Article

Methods for Model Complexity Reduction for the Nonlinear Calibration of Amplifiers Using Volterra Kernels

Francesco Centurelli, Pietro Monsurrò *, Giuseppe Scotti, Pasquale Tommasino and Alessandro Trifiletti

Department of Information, Electronics and Telecommunications Engineering, Sapienza University, 00184 Roma, Italy

* Correspondence: pietro.monsurro@uniroma1.it

Abstract: Volterra models allow modeling nonlinear dynamical systems, even though they require the estimation of a large number of parameters and have, consequently, potentially large computational costs. The pruning of Volterra models is thus of fundamental importance to reduce the computational costs of nonlinear calibration, and improve stability and speed, while preserving accuracy. Several techniques (LASSO, DOMP and OBS) and their variants (WLASSO and OBD) are compared in this paper for the experimental calibration of an IF amplifier. The results show that Volterra models can be simplified, yielding models that are 4–5 times sparser, with a limited impact on accuracy. About 6 dB of improved Error Vector Magnitude (EVM) is obtained, improving the dynamic range of the amplifiers. The Symbol Error Rate (SER) is greatly reduced by calibration at a large input power, and pruning reduces the model complexity without hindering SER. Hence, pruning allows improving the dynamic range of the amplifier, with almost an order of magnitude reduction in model complexity. We propose the OBS technique, used in the neural network field, in conjunction with the better known DOMP technique, to prune the model with the best accuracy. The simulations show, in fact, that the OBS and DOMP techniques outperform the others, and OBD, LASSO and WLASSO are, in turn, less efficient. A methodology for pruning in the complex domain is described, based on the Frisch–Waugh–Lovell (FWL) theorem, to separate the linear and nonlinear sections of the model. This is essential because linear models are used for equalization and cannot be pruned to preserve model generality vis-a-vis channel variations, whereas nonlinear models must be pruned as much as possible to minimize the computational overhead. This methodology can be extended to models other than the Volterra one, as the only conditions we impose on the nonlinear model are that it is feedforward and linear in the parameters.

Keywords: digital calibration; nonlinear models; complexity reduction; amplifiers; analog circuits; optimal brain surgeon; LASSO; orthogonal matching pursuit

Citation: Centurelli, F.; Monsurrò, P.; Scotti, G.; Tommasino, P.; Trifiletti, A. Methods for Model Complexity Reduction for the Nonlinear Calibration of Amplifiers Using Volterra Kernels. *Electronics* **2022**, *11*, 3067. https://doi.org/10.3390/electronics11193067

Academic Editor: Riccardo Bernardini

Received: 30 August 2022
Accepted: 21 September 2022
Published: 26 September 2022

1. Introduction

Modern communication systems are increasingly relying on digital signal processing, and the availability of the signal in digital form allows calibration in the digital domain, enabling the so-called digitally assisted analog electronics [1,2], where digital calibration is used to improve the performance of analog blocks. For instance, power amplifiers [3–6], IQ mixers [7,8] and ADCs [9–13] can be digitally enhanced, correcting nonlinear errors, I/Q mismatches, channel mismatches, etc. In particular, calibration can be used to reduce the nonlinearity of system components, and even of the entire system [14–17], by correcting nonlinear errors and improve the dynamic range of the system. For this goal, nonlinear models are required.

Volterra models [18] are often used to describe weakly nonlinear systems where memory effects are relevant. Depending on the required maximum nonlinearity order and memory depth, Volterra models may require a large number of parameters, resulting in high computational complexity and estimation problems. However, many of these parameters are often of little significance, and can be neglected with a limited loss of precision and a net reduction in computational complexity and resource cost.

Several complexity-reducing techniques have been presented in the literature to find sparse solutions without impairing model accuracy based on different principles. Sparsity, i.e., forcing parameters to 0, allows the reduction in the computational cost of the model by removing ("pruning") as many nonlinear terms as possible, in a manner that is compatible with the required accuracy. The Optimal Brain Surgeon (OBS) algorithm [19–23] starts from the largest model and iteratively removes the least significant coefficient until the simplest (and least accurate) model is found. It is mainly used in the neural network field and apparently has never been used for pruning Volterra models, which can be interpreted as a special case of neural networks [24]. The authors of [22] drew a parallel between neural and Volterra networks, proposed an analytical method to identify Volterra coefficients from the coefficients of a neural network and successfully compared their proposed method with the OBS and OBD [19,21] techniques. Their target application was face recognition. On the other hand, Orthogonal Matching Pursuit (OMP) techniques [25–27] start from the simplest model, finding the most correlated regressor, and iteratively add complexity to the model to improve accuracy. They are the techniques of choice in most of the literature on nonlinear calibration. Finally, Least Absolute Shrinkage and Selection Operator (LASSO) techniques [28] use weighting and regularization to find sparse solutions. These techniques use the L1-norm regularization term, which enforces sparsity, and depend on a regularization parameter, which indirectly determines the trade-off between sparsity and accuracy.

Variants of these techniques also exist. The Optimal Brain Damage (OBD) approach is a simplified version of OBS, which assumes orthogonal regressors [19,21]. Doubly OMP (DOMP) [27] is an improvement of the conventional OMP that employs the orthogonalization of the residual regressors. A variant of the LASSO technique [28] is the weighted LASSO (WLASSO) algorithm, which uses the least squares coefficients of the L2-norm regularized regression as weighting in the L1-norm regularizer.

Once a pruned model is obtained, estimation can be performed in real time with techniques such as least squares adaptive filters [29]. All these pruning techniques result in different trade-offs between complexity reduction and loss of precision, and one of the goals of this paper is to compare them to investigate their performance in optimizing the precision–complexity trade-off, with reference to the experimental calibration of an IF amplifier fed by QAM waveforms [30,31].

First, we present a methodology for pruning the nonlinear section of the Volterra model in the complex domain, while leaving the linear section (used for channel equalization) unaffected. In fact, while the channel's frequency response depends on many factors and the equalizer cannot be simplified without losing generalizability, the nonlinear response depends on the devices in the transmitter or receiver sections and is relatively time invariant. Hence, pruning of the linear section should be avoided. This is achieved by using the Frisch–Waugh–Lovell [32] theorem to separate regression in the linear and nonlinear steps, and performing pruning only on the latter. Pruning is performed in the complex domain, to make it compatible with the conventional linear equalization techniques used in communication systems [30,31].

This allows stating important guidelines for the choice of the most appropriate pruning technique to simplify the calibration model for RF systems, minimizing the computational complexity of the models without impacting accuracy. We prove that, in our dataset, the DOMP and OBS techniques outperform the others, and their combined use creates the best approximation of the optimal complexity–accuracy trade-off.

The same methodology can be applied to any nonlinear model, even those not derived by Volterra theory. In our derivation, in fact, we only assume that the model has no feedback (it is feedforward) and is linear in the parameters. Volterra models are thus only a subset of the class of nonlinear models that can be pruned using our approach.

The paper is organized as follows. Section 2 summarizes Volterra models in the real and complex domains, and the complexity-reducing algorithms to find sparse solutions for the linear-in-the-parameters (LIP) feedforward models used in this paper. Section 3 describes the experimental setup and the model identification techniques used to find the optimal calibration coefficients from known input waveforms. Section 4 shows and discusses the experimental results. Section 5 is the conclusion.

2. Volterra Models and Pruning Techniques

Nonlinear calibration techniques attempt to correct linear and nonlinear errors arising in electronic systems due to active devices. There is no standard model for nonlinearities, especially for dynamic nonlinearities, i.e., nonlinear effects with memory.

Usually, Volterra series are used, but these models easily become unmanageable due to the large number of coefficients. Volterra models can model weak nonlinear continuous systems with good accuracy [18], as they are based on a polynomial expansion of the nonlinear behavior of a system with memory. Such distortions are commonplace in analog and RF devices operating close to their maximum output power, but they are less adept for modeling discontinuities, such as those arising in ADCs or DACs [13]. Hence, we considered an IF amplifier as testbed for our comparison of pruning techniques.

We considered a nonlinear calibration technique that assumes the knowledge of the input transmitted signal (for instance, an equalization preamble in a packet-based communication system), and we focused on linear-in-the-parameters (LIP) feedforward models, where the nonlinear output is a linear combination of linear and nonlinear functions of the input, without feedback.

Since we assumed the system is sampled, we modeled the amplifier as a discrete-time system. The input signal is $x[n]$, sampled at the rate f_S, and produced by a DAC, whose linearity and noise were assumed to be better than that of the amplifier to be calibrated. The output of the device to be calibrated was $y[n]$, which was sampled, at the same rate, by an ADC, whose accuracy was also supposed to be better than that of the device to be calibrated. IF signals of bandwidth B_W around the carrier f_C were used, with $0 < f_C - {}^{B_W}/_2 < f_C + {}^{B_W}/_2 < {}^{f_S}/_2$.

In general, the nonlinear function describing the device is not known, and may be written as:

$$y[n] = f(x[n], \dots, x[n - M + 1]) \tag{1}$$

where $M - 1$ is the maximum memory depth of the system, which in principle may be infinite. The system is causal, meaning that only present and past input samples affect the output, and feedforward, implying that the actual output does not depend on the previous values of the output.

Digital calibration consists of approximating the inverse function of $f(\cdot)$ in the digital domain, so that the calibrated output $z[n]$ is as close as possible to the input $x[n]$. In this way, all deterministic errors affecting $y[n]$ are ideally removed (assuming system (1) is invertible):

$$z[n] = g(y[n], \dots, y[n - M + 1]). \tag{2}$$

Since such a model is unworkable, we focused on models that are linear-in-the-parameters and feedforward:

$$z[n] = \sum_{0}^{L-1} \beta_l g_l\big[y[n], \dots, y[n - M + 1]\big]. \tag{3}$$

This model has L unknown parameters, β_l, which are the coefficients of the linear combination of the known nonlinear functions $g_l(\cdot)$ of the input signal, including its past values. The use of LIP models ensures that a wide array of estimation algorithms for linear models can be employed: batch least squares, recursive least squares (RLS), least mean squares (LMS), etc. [29].

As the assumption implicit in (3) is that the output only depends on the input, i.e., it does not depend on the past values of the output. Such models are called feedforward, as there is no feedback of the output toward the input, and are a generalization of finite impulse response (FIR) filters. For instance, if $g_l\big[y[n], \dots, y[n-M+1]\big] \equiv y[n-l]$, and $L = M$, the model in (2) becomes a FIR filter, which is the workhorse of linear equalization techniques in telecommunications. The Volterra models we used in this paper are an example of LIP feedforward models, and can be considered generalizations of linear equalization. This implies, of course, that nonlinear models can perform both linear and nonlinear calibration, where linear calibration is commonly referred to as equalization.

Equation (3) is more general than Volterra models, because feedforward Volterra models are a subset of feedforward LIP models. The class of models described by Equation (3) also includes other models, such as functional-link artificial neural networks (FLANN) [33], and some of the restricted Volterra models [9,10] proposed in the literature to reduce model complexity a priori. Additionally, Hammerstein models [34] are feedforward and LIP. The methodology and techniques presented in this paper can be extended to any model that is both feedforward and LIP.

2.1. Real Volterra Models

(Feedforward) Volterra models are a generalization of FIR filters that allow for nonlinear behavior with memory. They can also be considered generalizations of the simplest nonlinear model: the memoryless polynomial.

In general, the Volterra model is the sum of kernels of degree $p = 1, \dots, P$, where monomials of order p are obtained from the input $y[n]$ using lagged terms up to a delay $M(p) - 1$, where the memory length can be a function of the degree, to minimize model complexity [9,10]. For instance, $y^2[n]$ is a term of degree 2 and delays $(0{,}0)$, whereas $y[n-1]y[n-2]y[n-3]$ is a term of degree 3 and delays $(1{,}2{,}3)$. There are p possible delays for a term of degree p, each going from 0 to $M(p) - 1$. For instance, with $p = 3$ and $M(p) = 4$, all the combinations of delays are the tuples from $(0{,}0{,}0)$ to $(3{,}3{,}3)$. Since products are commutative, delays can be placed in non-decreasing order: (m_1, m_2, m_3) would yield the same monomial as any permutation of the same indexes, such as (m_3, m_1, m_2). Hence, we focused on non-decreasing tuples and wrote the output of the kernel of degree p as:

$$z^p[n] = \sum_{i_1=0}^{M_p-1} \cdots \sum_{i_p=i_{p-1}}^{M_p-1} \beta_{i_1 \dots i_p} y[n-i_1] \cdots y\big[n-i_p\big]. \tag{4}$$

Finally, the output of the Volterra model is the sum of the outputs of the Volterra kernels, up to the highest order P:

$$z[n] = \sum_{p=1}^{P} z^p[n]. \tag{5}$$

The model may include a term of order 0, which is the offset.

The main problem with Volterra models is the number of coefficients: a 5th-order model with a delay of 4 would have 126 free parameters. The computational cost of Volterra models depends on the number of free parameters that needs to be estimated, plus a setup cost (which is significantly lower than the number of parameters) to compute all the monomials in the Volterra kernels. The number of free parameters to estimate has a direct impact on the complexity of the estimation technique, estimation convergence time and the stability of the estimation algorithms [29]. Hence, model pruning is of the essence to reduce model complexity and improve convergence time, i.e., the number of known samples required to identify the system. Model pruning reduces the cost of real-time correction and makes parameter estimation easier, faster and less prone to numerical problems.

2.2. Complex Volterra Models

Usually, equalization in communication systems is performed in the complex domain [30,31], after demodulation and carrier and timing recovery, to minimize the linear and nonlinear Inter-Symbol Interference (ISI) and allow symbol decision with the lowest Symbol Error Rate (SER). For this reason, we performed nonlinear calibration in the complex domain, so that it can be performed together with linear equalization. We thus need to express the Volterra models in the complex domain. Hence, we defined the intermediate frequency (*IF*) and baseband frequency (*BF*) signals, where ω_0 is the normalized carrier frequency $\omega_0 = 2\pi f_C T_S$, with f_C the carrier frequency in Hz, and T_S the sampling period:

$$y_{IF}[n] = \mathbb{Re}\{y_{BF}[n]e^{j\omega_0 n}\}. \tag{6}$$

A FIR filter can be written as:

$$z_{IF}[n] = \sum_{l=0}^{L-1} h_l y_{IF}[n-l] = \mathbb{Re}\left\{\sum\nolimits_{l=0}^{L-1} h_l y_{BF}[n-l]e^{j\omega_0(n-l)}\right\}. \tag{7}$$

If we rewrite the second expression, we obtain:

$$z_{BF}[n] = \sum\nolimits_{l=0}^{L-1} h_l y_{BF}[n-l]e^{-j\omega_0 l}. \tag{8}$$

Hence, a linear filter in the *IF* domain is equivalent to a linear filter in the *BF* domain, with the same (real) coefficients h_l, if the BF samples are phase rotated by $e^{-j\omega_0 l}$ and delayed by l.

Similar calculations hold for Volterra kernels of higher degrees. For instance, a generic quadratic term can be written as:

$$x_{IF}[n-l]x_{IF}[n-m] = \mathbb{Re}\{x_{BF}[n-l]e^{j\omega_0(n-l)}\}\mathbb{Re}\{x_{BF}[n-m]e^{j\omega_0(n-m)}\}. \tag{9}$$

By writing $\mathbb{Re}\{x\} = \frac{1}{2}(x + x^*)$, it is possible to obtain:

$$\begin{aligned} x_{IF}[n-l]x_{IF}[n-m] = \tfrac{1}{2}\mathbb{Re}\{&x_{BF}[n-l]x_{BF}[n-m]e^{j\omega_0(2n-l-m)} + \\ &x_{BF}[n-l]x_{BF}^*[n-m]e^{j\omega_0(-l+m)}\}. \end{aligned} \tag{10}$$

Similar relations can be obtained for higher-order kernels. Hence, *IF* kernels can be expressed in terms of the *BF* components, and the Volterra coefficients remain real and have the same value.

Some of the terms are frequency-modulated around 0, ω_0 or multiples of the carrier. For instance, $x_{BF}[n-l]x_{BF}^*[n-m]e^{j\omega_0(-l+m)}$ can be rewritten as $x_{BF}[n-l]x_{BF}^*[n-m]e^{j\omega_0(-n-l+m)}e^{j\omega_0 n}$, and the BF component is evidently modulated by $-\omega_0$. Such terms arise because nonlinearities produce terms at other frequencies. In narrowband systems, it is possible to neglect all the terms but those around ω_0, but we used all the terms because our system is wideband and because terms at carrier frequencies $2\omega_0$ or $3\omega_0$ may alias at lower frequencies owing to sampling.

The above equations were verified using MATLAB. The output of the *IF* model was the same as the output of the *BF* model with the same coefficients.

2.3. Orthogonalization and Linear and Nonlinear Sub-Models

Nonlinear effects are mostly due to the active elements in the transmitter and the receiver, so that most distortions occur before and after the channel. On the other hand, the channel adds a significant amount of linear gain and phase errors, which are usually removed via linear equalization. Hence, nonlinear errors are mostly predictable because they are caused by the transmit and receive hardware, whereas linear errors are time-varying, as they depend on channel conditions. Consequently, pruning can be used to select the correct nonlinear model, which is fixed, but cannot be used to select the linear coefficients, because each of them may be relevant or not depending on the channel's frequency response, which varies with time. Hence, a technique is required to separate linear and nonlinear coefficients in order to perform pruning only on the nonlinear part of the model.

This operation can be performed via the Frisch–Waugh–Lovell (FWL) theorem [32], which states that identification of a model with L variables can be performed in two steps, using the first $L_1 < L$ variables (the linear section) in the first step and the other $L_2 = L - L_1$ (the nonlinear section) in the second step. The theorem states that estimation yields the same residual and the same coefficients for the second step (the nonlinear section of the model) if the first variables are used to regress both the output and the remaining L_2 variables.

Basically, the FWL theorem orthogonalizes the output and the nonlinear section of the model with respect to the linear section, whose effect is completely removed. The generic nonlinear model, depending on a parameter vector β, with desired response V and input matrix X, can be split in two parts with input matrices X_1 and X_2. We can then regress V and X_2 over X_1 to remove the effect of linear filtering on the output and on the nonlinear sub-model.

$$V = X\beta = X_1\beta_1 + X_2\beta_2. \tag{11}$$

where $X \equiv [X_1; X_2]$ and $\beta = [\beta_1; \beta_2]$. The linear section X_1 is used to regress the output V and the nonlinear section X_2. The resulting regression coefficients are called γ_{y1} and γ_{21} and, in general, $\gamma_{y1} \neq \beta_1$. In this way, both the desired response V and the nonlinear sub-model X_2 become orthogonal to X_1, and the regression of the residual of V over the residual of X_2 (after regression against X_1) can be performed. Thence, the nonlinear model can be estimated using the regressed nonlinear section, and the ensuing parameter vector β_2 and the final error are the same:

$$V - X_1\gamma_{y1} = (X_2 - X_1\gamma_{21})\beta_2 \equiv V - X\beta = V - X_1\beta_1 - X_2\beta_2. \tag{12}$$

In our case, X is the $N \times L$ design matrix whose columns are the nonlinear functions $g_l(\cdot)$ of the system output $y[n]$, which is divided into a linear section X_1 and a nonlinear section X_2 containing all the Volterra terms of order higher than 1. V is the desired input of the system $x[n]$, and N is the number of samples. Because the models are LIP, linear regression techniques can be used for estimation.

In a communication system, the length of the linear section (which also includes an offset term) mostly depends on the channel, with a limited impact on the transmit and receive hardware. Hence, the length of the linear model X_1 will in general be much larger in a real application than in our experimental setting, where a single IF amplifier is modeled. However, the pruned nonlinear coefficients, mostly caused by the transmit and receive hardware, will not change with the channel, though they may slowly depend on bias or temperature parameters in the active devices.

We performed pruning only on the nonlinear section, so that we could select a simple nonlinear sub-model with good accuracy and limited hardware cost. Once the model is selected, the linear and nonlinear sections may be estimated together, so that the FWL theorem is only used for (off-line) model selection and not for real-time model estimation.

2.4. Pruning Techniques

Due to the large number of unknown parameters to be estimated, Volterra models are unworkable in practice and should be simplified. Moreover, in most practical cases, many of the parameters convey little information. Thus, the models can be greatly simplified without hindering accuracy. Several techniques can be employed to reduce model complexity, and they are described in the following sub-sections.

2.4.1. Optimal Brain Surgeon (OBS)

The OBS technique [19–23] has been proposed in the neural networks field to reduce the number of neurons with as little impact as possible on accuracy. The algorithm is greedy and starts from the full model, and then removes the least important coefficients one by one.

The algorithm estimates the importance of each coefficient by computing a score that depends on the coefficient value and its Hessian. The score is the result of a greedy optimization procedure that chooses the minimum residual error of the model constrained to have one zero coefficient. For a model of L parameters, there are L possible models with $L-1$ coefficients: some terms will be important, and their removal will significantly reduce accuracy, while others may be irrelevant and have little or no impact on accuracy. The OBS algorithm evaluates the importance of each parameter in the model by removing the one that has the lowest impact on accuracy. It is the closed-form solution of the iterative pruning procedure advocated in [9,10], and yields the same result. At each iteration, a linear constrained optimization problem is solved in closed form, so the decision is iteratively (greedily) optimal.

The main cost of the OBS technique is computing the diagonal of the inverse Hessian matrix of the regression problem. By assuming a diagonal Hessian, the problem can be greatly simplified at the expense of accuracy: the optimal brain damage (OBD) algorithm [19,21] is thus an approximate low-cost version of the OBS. Its performance is usually inferior, especially for high-correlated residuals, which are common in Volterra models. We compared both the OBS and OBD in the following, and while OBS is usually the best algorithm for pruning, OBD is not and should be avoided. Both OBS and OBD start with an accurate, but complex, model and yield simpler and simpler models at each iteration, attempting to minimize the loss of accuracy.

Since the OBS has never been used before for pruning Volterra kernels, whereas OMP and LASSO are not new to the field of nonlinear calibration with Volterra models, we add more details about this technique in what follows.

The general linear model $V \approx X\beta$ has a residual error $e = V - X\beta$, whose energy can be written as $E = e'e$, where the superscript is the transpose operator.

The ideal unconstrained model β must be constrained to have a zero coefficient, so that it is perturbed by a variation δ subject to the constraint $\beta_l + \delta_l = 0$ for the coefficient l. The goal is to choose l optimally, i.e., minimizing the increase in the model error. The unconstrained model has the following error:

$$E_0 = e'e = V'V - 2e'X'\beta + V'X'XV. \tag{13}$$

The constrained model has a larger error, because the unconstrained model was estimated to have the lowest possible residual error, E_0:

$$E_\delta = E_0 + \delta'X'X\delta. \tag{14}$$

This error depends on the parameter that has been nulled, and the goal is to find the parameter whose removal produces the lowest excess error. The Lagrangian of the constrained optimization problem is:

$$\Gamma = E_\delta + \lambda 1_l'(\beta + \delta) \tag{15}$$

where 1_l is a column vector of all zeros with only one '1' in position l, so that $1_l'(\beta + \delta) \equiv \beta_l + \delta_l = 0$. Forcing to zero the partial derivatives over δ and λ yields:

$$\begin{cases} \delta = \frac{\lambda}{2}(X'X)^{-1}1_l \\ \lambda = \frac{21_l'\beta}{1_l'(X'X)^{-1}1_l} \equiv \frac{2\beta_l}{H_{ll}} \end{cases} \tag{16}$$

where $H_{ll} = 1_l'(X'X)^{-1}1_l$ is the l-th diagonal element of the inverse Hessian matrix. Hence:

$$E_\delta = E_0 + \frac{\beta_l^2}{H_{ll}^2}1_l'(X'X)^{-1}1_l = E_0 + \frac{\beta_l^2}{H_{ll}}. \tag{17}$$

The coefficient with the least effect on the error for the constrained model is the one with the minimum E_δ, which is also the one with the minimum β_l^2/H_{ll}. This term is called the score, and the coefficient with the minimum score is eliminated, to produce a sparser model with the least additional error among all the models with one zero coefficient.

In the OBD algorithm, the cumbersome computation of $H = (X'X)^{-1}$ is replaced by the inverse of the l-th diagonal element of $X'X$. However, this is exact only when the matrix is diagonal, otherwise the inverse of the diagonal is not the diagonal of the inverse. The OBD is thus numerically simpler, but less accurate for non-diagonal (i.e., correlated) input matrices.

2.4.2. Orthogonal Matching Pursuit (OMP)

The OMP technique [25–27] operates in the opposite direction: it starts from the simplest model, by selecting the regressor with the highest absolute correlation coefficient, and then adds one element at a time to improve accuracy as quickly as possible.

The OMP technique has several variants. DOMP (doubly OMP [27]), for instance, performs the orthogonalization of all the residual (unused) regressors, i.e., something similar to a Gram–Schmidt orthogonalization procedure. At each step, the regressor with the highest correlation is selected, and the output and all the residual regressors are regressed and orthogonalized. This also imply that no matrix operations are required [32], because regression is computed one variable at a time.

Because of its superior numerical performance and analytical equivalence, we used the DOMP technique throughout this paper. The traditional OMP would yield the same result using high-precision arithmetic, but worse results on finite-precision machines.

OMP algorithms start with models with low complexity and low accuracy, and attempt to improve accuracy as quickly as possible, yielding progressively more complex, but more accurate, models.

2.4.3. Least Absolute Shrinkage and Selection Operator (LASSO)

The LASSO technique [28] employs an L1-norm regularization to enforce sparsity in the parameter vector. This is a common relaxation of the intractable L0-norm optimization: while the L0-norm is the sparsity of the vector (the number of nonzero components) and yields a non-convex optimization problem, the L1-norm is the sum of the absolute values of the regressors and yields a quadratic programming problem that can be easily solved using standard techniques, such as interior-point methods.

A variant of the LASSO technique, the weighted LASSO (WLASSO), uses a diagonal weighting matrix proportional to the inverse of the absolute value of the model obtained through conventional L2-norm regression (eventually with L2-norm regularization to ensure stability in the case of a strong correlation between the regressors). Hence, coefficients with larger values have lower weight, whereas small values are amplified. This results in a better performance because sparsity is reinforced by weighting. WLASSO outperforms LASSO. There are two regularization parameters: one for the initial L2-norm regularized regression, required to compute the weights, and one for the L1-norm regularized regression, which provides the actual coefficients.

While the OBS and OMP techniques change the model size one element at a time (producing iteratively smaller models in the OBS and larger models in the OMP), regularization yields an unpredictable level of sparsity, so many regressions with different regularization parameters need to be performed to estimate the accuracy–complexity frontier. The size of the pruned model cannot be predicted a priori, and sometimes different models with the same complexity (number of parameters) are selected with different regularization parameters. In this case, the model with the highest accuracy is selected among all those with the same complexity, yielding the optimal accuracy–complexity trade-off of the LASSO and WLASSO algorithms. Both algorithms are, however, outperformed by OBS and DOMP, as shown in Section 4.

3. Experimental Setup

We performed the experiments on a MiniCircuits ZX60-100VH+ IF amplifier [35], connected to an FMC150 board containing two 250MSps 14-bit ADCs and two 250MSps 14-bit DACs, and a Virtex-7 FPGA. The results were validated using different waveforms (with QAM-64 constellation) and different acquisitions of the same waveform.

The amplifier has a gain of 36 dB and a bandwidth of 0.3–100 MHz; with an output compression point of 30 dBm, it accepts power levels up to about −6 dBm before compression. It was tested using modulated waveforms of 50 MHz bandwidth around a 50 MHz carrier.

The setup, composed of the DAC, connected through SMA cables to the amplifier to calibrate, and finally to the ADC, included several attenuators to have roughly unitary gain. An input attenuator of 3 dB and two output attenuators for a total of 30 dB were added. The attenuators, of the VAT-X+ series by Mini-Circuits, had 6GHz bandwidth, so their frequency response is almost ideal in the band of interest. The ADC was preceded by a SLP-100+ lowpass filter with 100 MHz bandwidth as anti-aliasing filter. The DAC and ADC, operating at 250MSps, were AC-coupled. The DAC was also connected to a built-in lowpass filter; hence, the usable input bandwidth was about 10–80 MHz.

Preliminarily, we tested the DAC/ADC loop alone to check the Error Vector Magnitude (EVM) of the setup without amplifiers. The DAC/ADC chain without any amplifier or attenuator has an EVM of 0.6% with a FIR filter of 9 taps, and 0.2% with a FIR filter of 21 taps. This sets the upper limit to accuracy. The accuracy was estimated as the error after linear calibration of a QAM waveform with 50MHz carrier frequency and 50 MHz bandwidth. These results show that the accuracy of the DAC and ADC chain is better than the system to be analyzed and calibrated, which implies that our calibration techniques actually improve the performance of the IF amplifier, and not of the experimental setup.

Furthermore, Volterra models cannot improve the DAC/ADC chain, whose distortions are not due to weak nonlinearities as those modeled by Volterra kernels, so that calibration is pointless for the DAC/ADC chain and can only improve the performance of the amplifier. The averaging of seven waveforms limited the impact on the accuracy of the DAC/ADC chain (−1.3 dB EVM, instead of the theoretical −8.5 dB if all the EVM were due to noise and averaging were effective in reducing errors), suggesting that almost the entirety of the error between the received and transmitted waveforms is due to non-stochastic effects.

Of course, a significant improvement in linearity was obtained after calibration when the amplifier was included, because the amplifier introduced significant distortions, which could be improved due to the use of Volterra models. However, these preliminary tests show that linearity and noise are not limited by the ADC/DAC chain, so the impairments of the amplifier dominate the EVM and SER results.

4. Experimental Results and Discussion

This section reports on the experimental results. The full-scale value of the DAC is 1 Vpp and the input waveform had a peak-to-peak swing of 900 mVpp. The full-scale value of the ADC is 2 Vpp, so the expected output swing of the received waveform was about 32% of the full swing of the ADC. This was confirmed by the measurements.

Additional measurements with 1, 2, 3 and 6 dB of additional input attenuation were made, but, after 3 dB, the SER was zero even without calibration (only linear equalization was necessary), because EVM was limited. Nonlinear ISI is significant only for a large input power.

Several waveforms were acquired, each with more than 3000 symbols and thus 15,000 samples, for different QAM-64 waveforms and different acquisitions of the same waveform to allow the averaging of stochastic effects. Averaging had a limited impact on performance, implying that linear and nonlinear deterministic errors are dominant.

Characterization of the IF Amplifier

Figure 1 shows the transmitted and received spectra. Spectral regrowth is clearly evident in the output waveform (red) with respect to the input waveform (black). The output spectrum has about 3 dB of gain loss at 80 MHz, which is compatible with the 80 MHz pulse-shaping lowpass filter after the DAC. The nonlinear spectral regrowth after the amplifier is attenuated after 100 MHz by the anti-aliasing lowpass filter before the ADC.

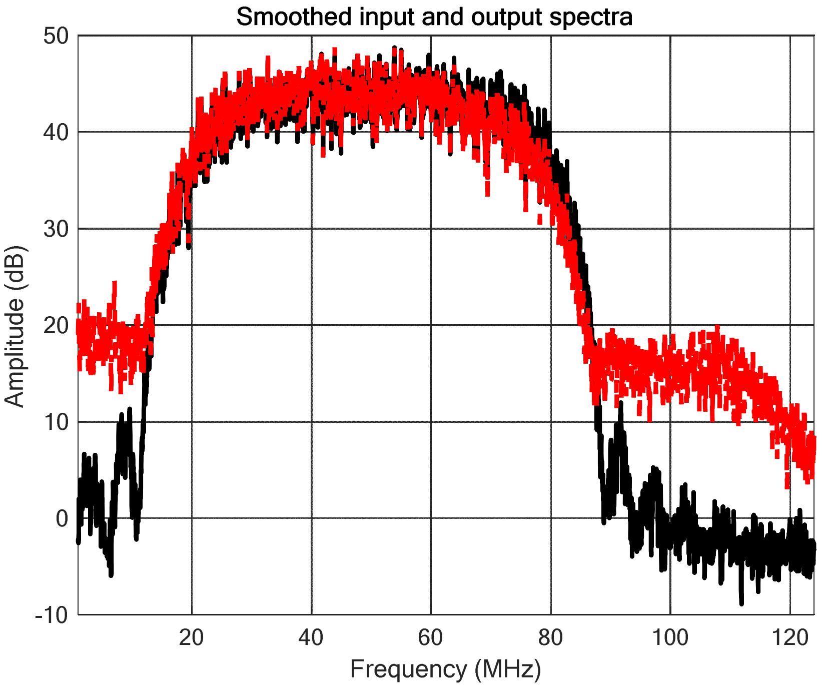

Figure 1. Input (black) and output (red) spectra. Spectral regrowth is clearly visible at the output.

Complex linear and nonlinear equalizers [30,31] were used to perform linear equalization with a FIR filter (a first-order Volterra kernel) and nonlinear calibration with Volterra kernels of higher order. The Frisch–Waugh–Lovell (FWL) decomposition [32] was used to separate the linear and nonlinear parts of the Volterra model, and allow pruning only on the nonlinear part.

Figure 2 shows the Symbol Error Rate (SER) and Error Vector Magnitude (EVM) after equalization with nine linear coefficients (plus offset correction). The SER is 0.4%, and the EVM is 5.4%. Figure 3 shows the received constellation. At the four diagonal corners of the constellation, heavy distortions are evident.

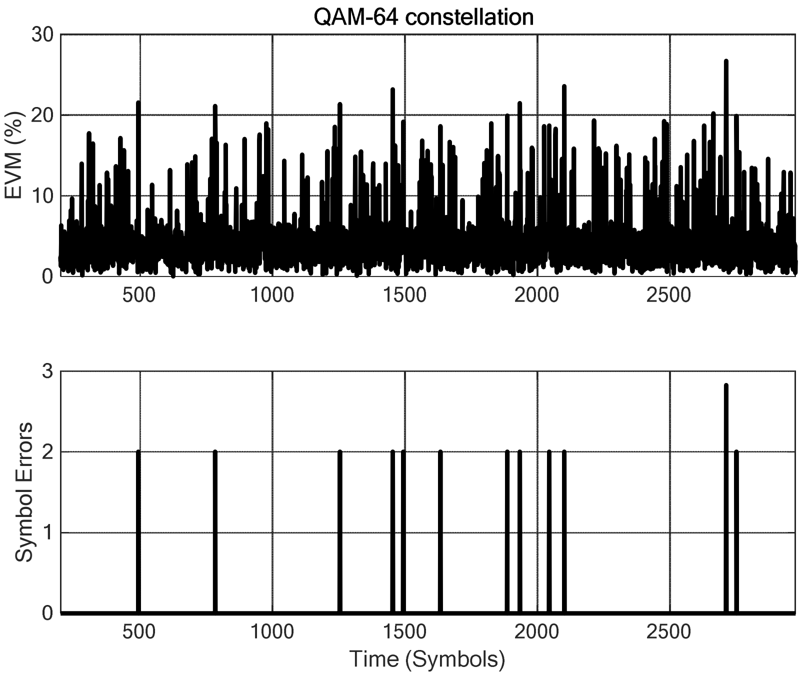

Figure 2. EVM (top) and SER (bottom) after equalization with 9 linear coefficients and offset correction. Transmission errors are evident, and the EVM is too large.

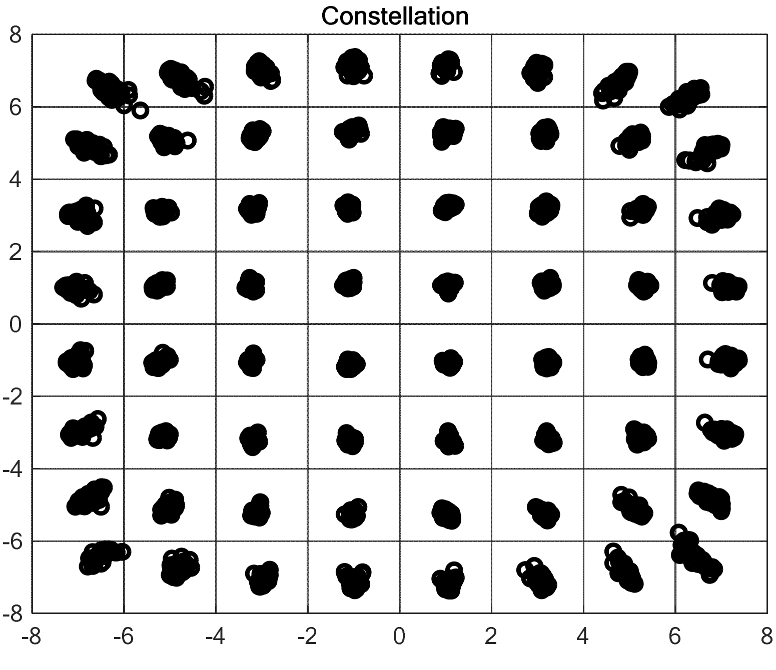

Figure 3. Received constellation after equalization. The QAM-64 constellation should form a square of 8 dots per dimension. Noise is limited (the central dots are small), but heavy distortion occurs at the diagonal corners, due to nonlinear effects. Such distortions produce transmission errors, as shown in Figure 2.

Longer filters do not improve the EVM and SER, because the EVM errors are due to distortions and noise and cannot be corrected via linear filtering. A real communication system must have a sufficiently long linear adaptive filter to take into account the entire impulse response of the channel. These measurements only take into account the linear frequency response of the amplifier, the pulse-shaping filter after the DAC and the anti-aliasing filter before the ADC (the attenuators being close to ideal in the band of interest). Hence, nine FIR coefficients are in general not sufficient for equalization, but are sufficient in this case because the frequency response is relatively smooth.

Figure 4 shows the SER and EVM after calibration. The linear kernel has a length of 8, the quadratic kernel length 2, the cubic kernel length 4, the quartic kernel length 0, and the quintic kernel length 2. Distortions were in fact dominated by odd-order terms, and shorter kernels had an insufficient impact on post-calibration EVM. SER is now zero, and the EVM has fallen to 2.9%, almost twice lower. Figure 5 shows the constellation, which is more similar to the ideal one, though some residual distortion is still evident.

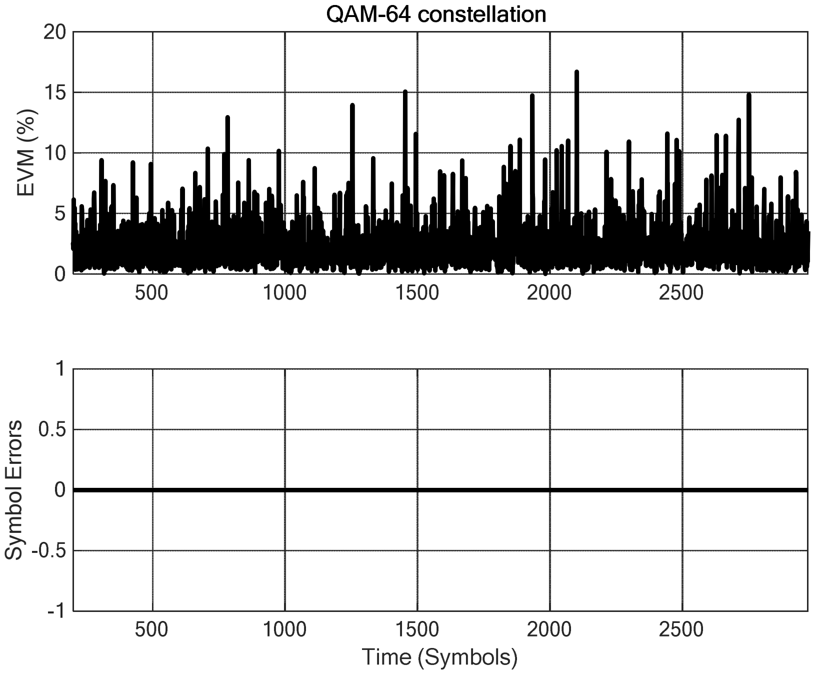

Figure 4. EVM (top) and SER (bottom) after calibration with 72 coefficients. Symbol errors are no longer present, but model complexity significantly increased.

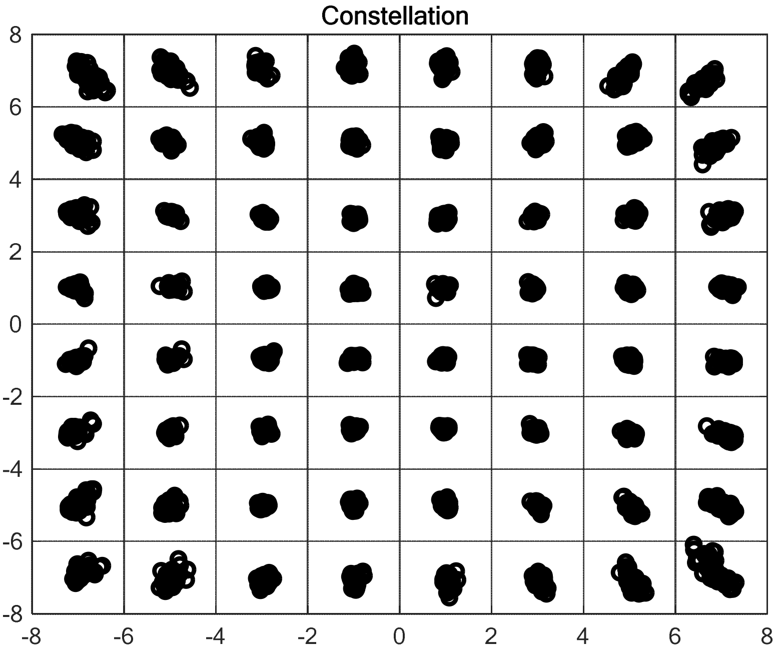

Figure 5. Received constellation after calibration. The shape of the constellation has significantly improved, though some nonlinear errors are still present at the four diagonal corners of the constellation, where amplitude is maximum.

Overall, complexity increases from 10 to 72 coefficients, hence pruning is required. The pruning algorithms described in Section 2 were exploited and compared, taking also into account their possible variants (OBD and WLASSO). Since DOMP is more efficient but analytically equivalent to OMP, the former was preferred. Iterative pruning [9,10] can be shown to be equivalent to OBS, but much less efficient computationally, and it has also been neglected. Model pruning was thus performed using the OBS, OBD, DOMP, LASSO and WLASSO techniques described previously.

Figure 6 shows the impact of the five pruning techniques on accuracy. The x-axis is the number of nonlinear coefficients (62, because 10 coefficients are in the linear part), and the y-axis is the RMS error between the desired and calibrated response, which is related to EVM.

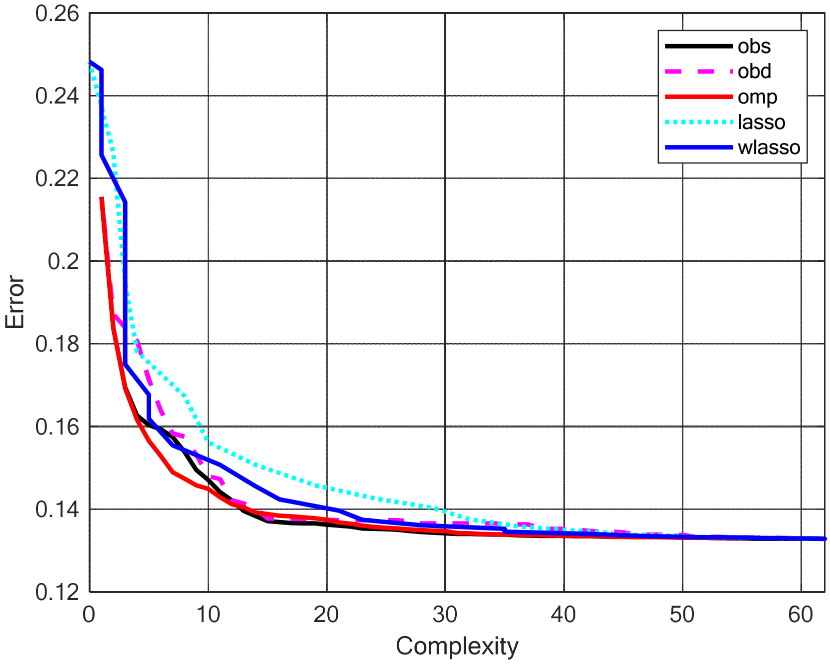

Figure 6. Accuracy vs. complexity for the five pruning techniques. OBS and DOMP outperform all the others. LASSO is the least efficient, whereas OBD is not bad for relatively large models. WLASSO is significantly better than LASSO, though still inefficient with respect to OBS and DOMP.

DOMP and OBS taken together outperform the other techniques, which have larger errors for the same complexity, or higher complexity for the same accuracy. Hence, using both DOMP and OBS, the "technological possibility frontier" of the optimal pruned models for each complexity level was obtained. Of course, none of these techniques was optimal, because the optimal would require trillions of regressions to test all the possible combinations of input variables. However, the combined use of DOMP and OBS allowed finding the best approximation of the ideal trade-off.

Figure 7 shows the OBS and DOMP techniques alone. The complete model of 62 coefficients can be pruned up to 12–15 nonlinear coefficients with limited loss in accuracy, from 0.133 to 0.14 of RMS error. OBS is more efficient in the center of the graph, whereas DOMP is more efficient for smaller but less accurate models. However, in the region where DOMP is more efficient, error increases rapidly, whereas the OBS-dominated region is flatter and closer in accuracy to the full model.

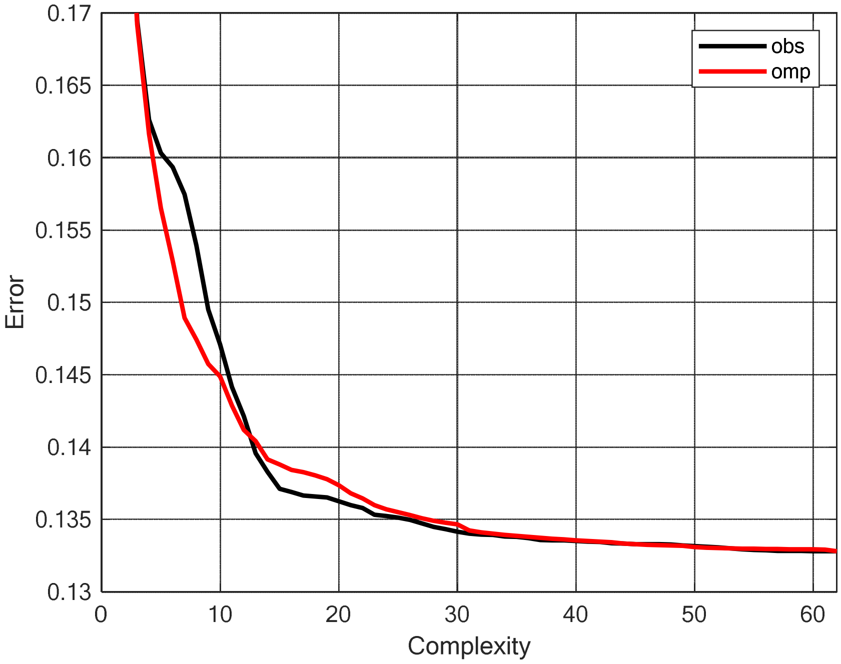

Figure 7. Zoom for OBS vs. DOMP, the two most promising pruning techniques. The two curves intersect each other, so that neither OBS nor DOMP are optimal by themselves. However, their combined use allows finding the best approximation of the optimal complexity-accuracy trade-off.

A very limited increase in the RMS error can be obtained reducing the number of nonlinear coefficients from 62 to less than 15, implying a pruning ratio better than 75%.

Figure 8 shows the EVM and SER of the pruned model. EVM is 3.1%, up from 2.9% before pruning, and SER remains zero. This result is obtained with 12 nonlinear coefficients, so that the complete model has 22 coefficients, as 9 parameters are for the linear section, and 1 for offset removal. A linear equalizer in a real scenario may have much more than 9 coefficients, possibly also 40 or more, so that the increased complexity in adding 12 nonlinear coefficients is more limited than it appears.

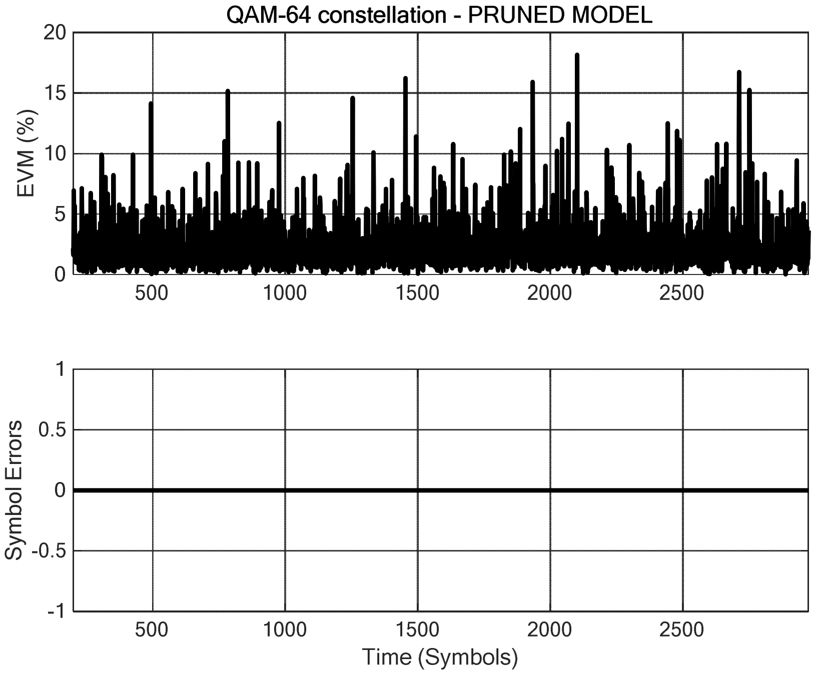

Figure 8. SER and EVM after pruning to 22 coefficients. The EVM increase is minimal, and SER is still zero, despite the fact that the nonlinear coefficients are decreased from 62 to just 12, close to an 80% reduction in complexity.

Figure 9 shows the constellation after pruning. It is remarkably similar to the constellation in Figure 5, before pruning. Though there is some residual nonlinearity, SER is zero, as shown in Figure 8.

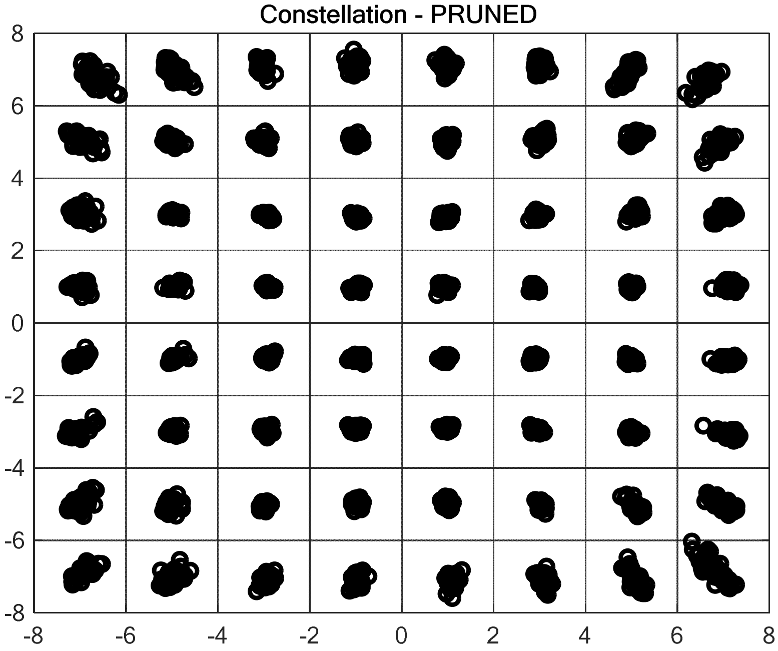

Figure 9. Constellation after pruning with 22 coefficients.

A total of 14 waveforms for each of the two input files were acquired. The 14 waveforms were used with and without averaging (with a process gain of 11.5 dB). No difference between the averaged and non-averaged data were observed with equalization, calibration and pruning. Hence, most of EVM is due to residual linear and nonlinear ISI, and not to noise, otherwise EVM would have fallen somewhat with averaging. If EVM were due to noise, the averaging of synchronous waveforms would have improved EVM by as much as 11.5 dB, but the limited impact of averaging proves that nonlinear effects dominate.

Estimation is performed over about 3174 symbols, but convergence is achieved after about 1400 symbols, as shown in Figure 10, where EVM computed over the entire (train + test) dataset on the *y*-axis is shown as a function of the number of training symbols on the *x*-axis. This is achieved with the pruned model of 12 nonlinear parameters. The train dataset includes the first symbols, and the test dataset includes the remaining symbols. The training time includes the estimation of the linear and nonlinear parts of the model, for a total of 22 parameters. In a real implementation, the linear section would be longer to allow for channel equalization, so that convergence time would be somewhat larger.

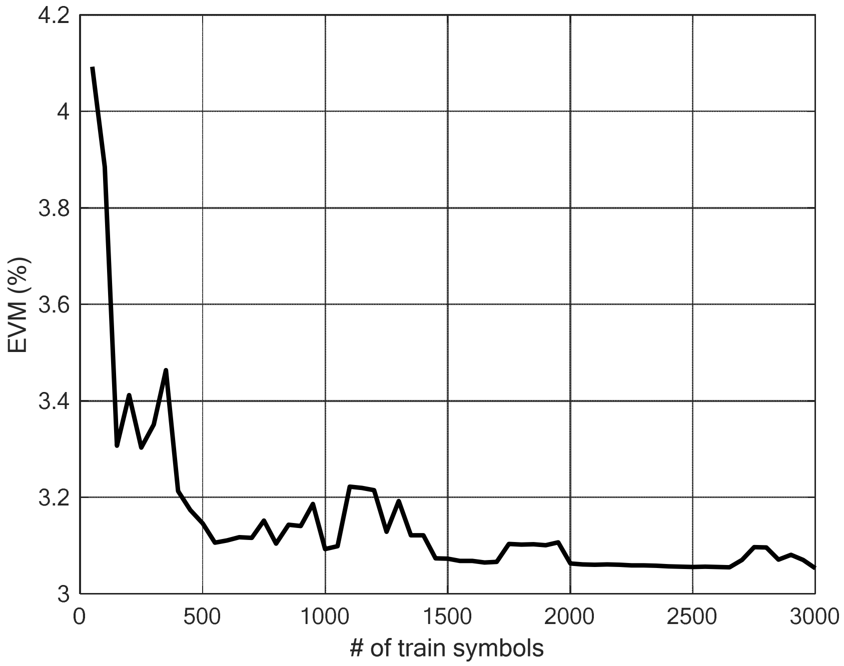

Figure 10. EVM vs. the number of symbols in the train dataset. Convergence is achieved after about 1500 samples. EVM is computed over both the train and test datasets, where the train dataset includes the first symbols, used for training, and the test dataset includes the remaining symbols, used to compute the EVM but not to estimate the model coefficients.

Finally, Figure 11 shows the EVM of the equalization (with 10 coefficients), calibration (with 72 coefficients) and pruned (with 22 coefficients) models as a function of the input power. Five acquisitions of the same waveforms were performed at the normalized power levels of 0 dB, −1 dB, −2 dB, −3 dB and −6 dB. Calibration allows halving the EVM with respect to mere equalization, and this effect is evident especially at larger input power levels. The pruned model is almost as accurate as the full model, though with 12 instead of 62 additional model coefficients to estimate.

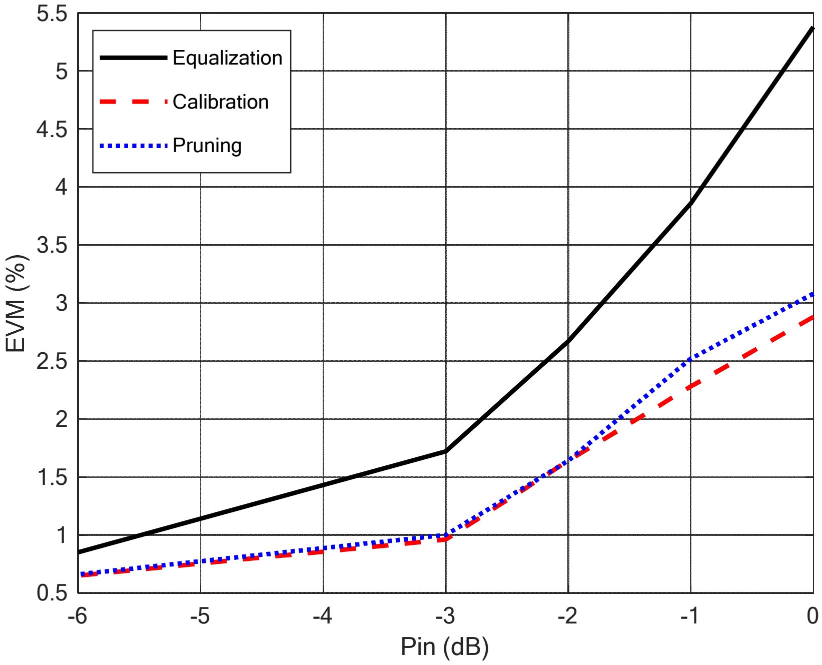

Figure 11. EVM of the equalization (black), calibration (red) and pruned (blue) models as a function of the input power level. Calibration allows reducing EVM by a factor 2, and pruning has limited impact on accuracy, though a very large one on model complexity (with a reduction from 62 to 12 nonlinear model coefficients).

5. Conclusions

This paper focused on pruning techniques for model complexity reduction in Volterra models, with the goal of comparing such techniques on the experimental data produced by an IF amplifier fed by a QAM-64 waveform.

Several pruning techniques were compared for the nonlinear calibration of a commercial IF amplifier, with the goal of finding sparse models with high accuracy. The results show the effectiveness of the OBS and DOMP techniques for model pruning, while showing the inefficiency of LASSO, WLASSO and OBD. More in detail, the combined use of OBS and DOMP was proved to be optimal, meaning that the best of the two provides the optimal trade-off between accuracy and complexity: OBS is better for more complex models, those closer in performance to the full model, whereas DOMP is better for smaller models, when the error is however larger. The other three techniques produce less sparse or less accurate models, and they are shown to be dominated by either the OBS or DOMP techniques. Hence, they do not need to be considered when performing pruning, as using both OBS and DOMP provides the best model for a given accuracy or complexity, so that the best choice between these two techniques is "optimal". More precisely, because all these algorithms are greedy, no optimal solution to the pruning problem can be found, but using OBS and DOMP allows finding the best solution among all the techniques that were compared. Hence, in addition to investigating the use of the OBS and OBD techniques for Volterra model pruning, we conclude that the combined used of the OBS and DOMP techniques is preferable in approximating the optimal complexity–accuracy trade-offs.

Calibration was performed directly on the complex BF components to allow minimizing linear and nonlinear ISI at the same time. Furthermore, the FWL theorem was used to separate the linear and nonlinear response of the Volterra kernels, and we performed pruning only on the nonlinear components whose response was relatively fixed. This allowed selecting the relevant Volterra terms for accurate but low-complexity nonlinear calibration without removing the linear terms, which were not pruned, to allow the equalization of generic channel frequency responses in real applications. The FWL theorem allowed concentrating the pruning process on the nonlinear part, and thus producing a simple nonlinear model without affecting the generalizability of the full model with respect to arbitrary linear channel responses.

QAM-64 waveforms were used to test the amplifier. Mere equalization was not sufficient to reduce SER to zero, because the amplifier was driven beyond its compression point and nonlinear distortions were significant. However, Volterra kernels were sufficient to reduce SER to zero, though with a large increase in computational complexity. Pruning techniques were used, and the number of nonlinear coefficients for proper calibration was reduced from 62 to 12. Hence, with 12 nonlinear coefficients, it is possible to decode the input QAM-64 waveforms with limited waveform error and no decision errors during decoding.

Significant reductions in complexity (by a factor 4–6 in terms of number of nonlinear model parameters) were achieved with a negligible impact on model accuracy, improving the Error Vector Magnitude (EVM) by about 6dB with models containing about 20 parameters (including those for linear equalization). The 64-QAM constellations were used, and estimation takes fewer than 16.000 samples (about 3.200 symbols) to converge using a batch least squares estimator.

The methodology described in this paper can be extended to any feedforward LIP model and is not limited to Volterra models. Hence, any model in this large class can be used to allow calibration in the complex domain, separation of linear and nonlinear submodels and the pruning of the nonlinear coefficients. For instance, Hammerstein and Functional-Link Artificial Neural Network (FLANN) models may be employed. The methodology can be used for any analog and RF component: RF amplifiers, such as power and low-noise amplifiers, active filters and mixers. The proposed analysis can be used whenever a nonlinear impairment can be modeled with a LIP feedforward model. Once the model is pruned, real-time VHDL implementations of the nonlinear calibration techniques can be developed and tested. Pruning reduces the computational complexity and improves numerical stability and convergence time. These possibilities will be investigated in future works.

Author Contributions: conceptualization, F.C., P.M., G.S., P.T. and A.T.; methodology, P.M.; software, P.M. and G.S.; validation, F.C., P.M., G.S., P.T. and A.T.; formal analysis, P.M.; investigation, F.C., P.M., G.S., P.T. and A.T.; resources, F.C. and A.T.; data curation, P.M.; writing—original draft preparation, P.M.; writing—review and editing, F.C., P.M., G.S., P.T. and A.T.; visualization, P.M. and F.C.; supervision, A.T.; funding acquisition, F.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research was founded by the University of Rome Sapienza.

Conflicts of Interest: The authors declare no conflicts of interest.

List of Acronyms

ADC	Analog-to-Digital Converter
BF	Baseband Frequency
DAC	Digital-to-Analog Converter
DOMP	Doubly Orthogonal Matching Pursuit
EVM	Error Vector Magnitude
FIR	Finite Impulse Response
FLANN	Functional-Link Artificial Neural Network
FPGA	Field-Programmable Gate Array
FWL	Frisch–Waugh–Lovell
IF	Intermediate Frequency
ISI	Inter-Symbol Interference
LASSO	Least Absolute Shrinkage and Selection Operator

LIP	Linear-In-the-Parameters
LMS	Least Mean Squares
OBD	Optimal Brain Damage
OBS	Optimal Brain Surgeon
OMP	Orthogonal Matching Pursuit
QAM	Quadrature Amplitude Modulation
RF	Radio Frequency
RLS	Recursive Least Squares
RMS	Root Mean Square
SER	Symbol Error Rate
WLASSO	Weighted Least Absolute Shrinkage and Selection Operator

References

1. Murmann, B. Digitally assisted analog circuits. *IEEE Micro* **2006**, *26*, 38–47.
2. Fayazi, M.; Colter, Z.; Afshari, E.; Dreslinski, R. Applications of artificial intelligence on the modeling and optimization for analog and mixed-signal circuits: A review. *IEEE Trans. Circuits Syst. I Regul. Pap.* **2021**, *68*, 2418–2431.
3. Le Duc, H.; Feuvrie, B.; Pastore, M.; Wang, Y. An adaptive cascaded ILA- and DLA-based digital predistorter for linearizing an RF power amplifier. *IEEE Trans. Circuits Syst. I Regul. Pap.* **2019**, *66*, 1031–1041.
4. Li, Z.; Niu, G.; Liang, Q.; Imura, K. Intermodulation linearity in high-k/metal gate 28 nm RF CMOS transistors. *Electronics* **2015**, *4*, 614–622.
5. Mirri, D.; Filicori, F.; Iuculano, G.; Pasini, G. A nonlinear dynamic model for performance analysis of large-signal amplifiers in communication systems. *IEEE Trans. Instrum. Meas.* **2004**, *53*, 341–350.
6. Younes, M.; Kwan, A.; Rawat, M.; Ghannouchi, F.M. Linearization of concurrent tri-band transmitters using 3-D phase-aligned pruned Volterra model. *IEEE Trans. Microw. Theory Techn.* **2013**, *61*, 4569–4578.
7. Anttila, L.; Valkama, M.; Renfors, M. Frequency-selective I/Q mismatch calibration of wideband direct-conversion transmitters. *IEEE Trans. Circuits Syst. II Express Briefs* **2008**, *55*, 359–363.
8. Huo, D.; Mao, L.; Wu, L.; Zhang, X. A linearity improvement front end with subharmonic current commutating passive mixer for 2.4 GHz direct conversion receiver in 0.13 μm CMOS Technology. *Electronics* **2020**, *9*, 1369.
9. Centurelli, F.; Monsurrò, P.; Rosato, F.; Ruscio, D.; Trifiletti, A. Calibrating sample and hold stages with pruned Volterra kernels. *Electron. Lett.* **2015**, *51*, 2094–2096.
10. Centurelli, F.; Monsurrò, P.; Rosato, F.; Ruscio, D.; Trifiletti, A. Calibration of pipeline ADC with pruned Volterra kernels. *Electron. Lett.* **2016**, *52*, 1370–1371.
11. Medawar, S.; Murmann, B.; Händel, P.; Björsell, N.; Jansson, M. Static integral nonlinearity modeling and calibration of measured and synthetic pipeline analog-to-digital converters. *IEEE Trans. Instrum. Meas.* **2014**, *63*, 502–511.
12. Nikaeen, P.; Murmann, B. Digital compensation of dynamic acquisition errors at the front-end of high-performance A/D converters. *IEEE J. Sel. Top. Signal Process.* **2009**, *3*, 499–508.
13. Björsell, N.; Suchanek, P.; Händel, P.; Rönnow, D. Measuring Volterra kernels of analog-to-digital converters using a stepped three-tone scan. *IEEE Trans. Instrum. Meas.* **2008**, *57*, 666-671.
14. Grimm, M.; Allén, M.; Marttila, J.; Valkama, M.; Thomä, R. Joint mitigation of nonlinear RF and baseband distortions in wideband direct-conversion receivers. *IEEE Trans. Microw. Theory Techn.* **2014**, *62*, 166–182.
15. Ge, L.; Zhang, W.; Liang, C.; He, Z. Threshold-based pruned retraining Volterra equalization for 100 Gbps/lane and 100-m optical interconnects based on VCSEL and MMF. *J. Lightwave Technol.* **2019**, *37*, 3222–3228.
16. Yu, Y.; Choi, M.R.; Bo, T.; He, Z.; Che, Y.; Kim, H. Low-complexity second-order Volterra equalizer for DML-based IM/DD transmission system. *J. Lightwave Technol.* **2020**, *38*, 1735–1746.
17. Yadav, G.; Chuang, C.-Y.; Feng, K.-M.; Yan, J.-H.; Chen, J.; Chen, Y.-K. Reducing computational complexity by using elastic net regularization based pruned Volterra equalization in a 80 Gbps PAM-4 signal for inter-data center interconnects. *Opt. Express* **2020**, *28*, 38539–38552.
18. Mathews, V.J.; Sicuranza, G.L. *Polynomial Signal Processing*; Wiley: Hoboken, NJ, USA, 2000.
19. Haykin, S. *Neural Networks and Learning Machines*, 3rd ed.; Pearson: Upper Saddle River, NJ, USA, 2009.
20. Hassibi, B.; Stork, D.G.; Wolff, G.J. Optimal brain surgeon and general network pruning. In Proceedings of the 1993 IEEE International Conference on Neural Networks, San Francisco, CA, USA, 28 March–1 April 1993; pp. 293–299.
21. Gorodkin, J.; Hansen, L.K.; Krogh, A.; Svarer, C.; Winther, O. A quantitative study of pruning by optimal brain damage. *Int. J. Neural Syst.* **1993**, *4*, 159–169.
22. Tian, X.; Becerra, V.; Bausch, N.; Vinod, G.; Santhosh, T.V. A method for measuring the robustness of diagnostic models for predicting the break size during LOCA. In Proceedings of the PHM 17 Annual Conference on Prognostics and Health Management Society, St. Petersburg, FL, USA, 2–5 October 2017; pp. 2–10.
23. Rubiolo, M.; Stegmayer, G.; Milone, D. Compressing arrays of classifiers using Volterra-neural network: Application to face recognition. *Neural Comput. Appl.* **2013**, *23*, 1687–1701.
24. Marmarelis, V.Z.; Zhao, X. Volterra models and three-layer perceptrons. *IEEE Trans. Neural Netw.* **1997**, *8*, 1421–1433.

25. Becerra, J.A.; Madero-Ayora, M.J.; Noguer, R.G.; Crespo-Cadenas, G. On the optimum number of coefficients of sparse digital predistorters: A Bayesian approach. *IEEE Microw. Wirel. Compon. Lett.* **2020**, *30*, 1117–1120.
26. Zhang, G.; Hong, X.; Fei, C.; Hong, X. Sparsity aware nonlinear equalization with greedy algorithms for LED-based bisible light communication systems. *J. Lightwave Technol.* **2019**, *37*, 5273–5281.
27. Becerra, J.A.; Madero-Ayora, M.J.; Crespo-Cadenas, G. Comparative analysis of greedy pursuits for the order reduction of wideband digital predistorters. *IEEE Trans. Microw. Theory Techn.* **2019**, *67*, 3575–3585.
28. Kekatos, V.; Giannakis, G.B. Sparse Volterra and polynomial regression models: Recoverability and estimation. *IEEE Trans. Signal Process.* **2011**, *59*, 5907–5920.
29. Proakis, J.G.; Manolakis, D.G. *Digital Signal Processing: Principles, Algorithms and Applications*; Prentice Hall: Upper Saddle River, NJ, USA, 1996.
30. Hanzo, L.; Webb, W.; Keller, T. *Single- and Multi-Carrier Quadrature Amplitude Modulation*; Wiley: Chichester, UK, 2000.
31. Proakis, J.G. *Digital Communications*, 3rd ed.; McGraw-Hill: New York, NY, USA, 1995.
32. Monsurrò, P.; Trifiletti, A. Faster, stabler and simpler—A recursive-least-squares algorithm exploiting the Frisch-Waugh-Lovell theorem. *IEEE Trans. Circuits Syst. II Express Briefs* **2017**, *64*, 344–348.
33. Patra, J.C.; Panda, G.; Baliarsingh, R. Artificial neural network-based nonlinearity estimation of pressure sensors *IEEE Trans. Instrum. Meas.* **1994**, *43*, 874–881.
34. Younes, M.; Ghannouchi, F.M. An accurate predistorter based on a feedforward Hammerstein structure. *IEEE Trans. Broadcasting* **2012**, *58*, 454–461.
35. Mouser Electronics, Inc. Mini-CIRCUITS, ZX60-100VH+ Coaxial Amplifier Datasheet. Available online: https://www.mouser.com/datasheet/2/1030/ZX60-100VH_2b-1701555.pdf (accessed on 18 September 2022).