Analysing digital predistortion technique for computation-efficient power amplifier linearisation in the presence of measurement noise

Girish Chandra Tripathi | Meenakshi Rawat

SDR Lab, Department of Electronics and Communication Engineering, Indian Institute of Technology Roorkee, India

Correspondence
Girish Chandra Tripathi, Meenakshi Rawat, SDR Lab, Department of Electronics and Communication Engineering, Indian Institute of Technology Roorkee, India.
Email: gtripathi@ec.iitr.ac.in; meenakshi.rawat@ece.iitr.ac.in

Abstract
The noise is an inherent part of a transceiver system, which becomes more severe on low-cost systems. The power amplifier (PA) further enhances this noise and the signal to be propagated to the receiver. The conventional approach of digital predistortion (DPD) assumes an ideal transceiver system while extracting data for the predistortion function generation. As a result, performance limitation arises due to residual signal–noise interaction. This study presents the accuracy and implementation issues of DPD on a low-cost transceiver having lower bit resolution in the presence of transceiver noise. Different model architectures, as well as processing algorithms, are compared in terms of numerical stability of the solution (condition number of observation matrix), efficient field-programmable gate array (FPGA) implementation (dispersion of coefficients, in-band model performance (normalised mean square error), and out-of-band model performance (adjacent channel power ratio). The simulation results are tested on FPGA and direct conversion transceiver-based platform using PA. A long-term evolution signal with 64 quadrature amplitude modulation is used for performance evaluation. The suitability of the various polynomial models for fixed-point implementation and the required memory size for implementing the DPD model is further established.

1 INTRODUCTION

The higher data rate communication is achieved using complex envelope techniques like long-term evolution (LTE) and LTE-advanced by incorporating higher quadrature amplitude modulation schemes [1]. However, such a signal imposes a strict linearity requirement on the radio-frequency power amplifier (PA) due to the high peak-to-average power. Therefore, it requires operating the PA in the saturation region for the highest efficiency, but it produces distortions; hence, it is essential to compensate for such distortions by the well-established technique of digital predistortion (DPD) [2–5].

The linearisation using DPD can be achieved using direct or indirect learning methodology. Figure 1 shows the popular implementation of DPD using indirect learning architecture (ILA) [6]. The possibility of identifying a post-distorter, which is equivalent to the predistorter, is mathematically established in [7] for Volterra-based models. This equivalence allows finding the PA inverse in an ‘indirect’ way using linear regression instead of cumbersome iterative or non-linear optimisation techniques utilised in direct learning architecture. In Figure 1, \( x(n) \) is the input signal to the predistorter, \( g(n) \) is the predistorter output and input to PA, and \( z(n) \) is the output of PA after the application of the predistorted signal \( g(n) \). Predistorter (or DPD model) can be any non-linear model that maps normalised baseband output to baseband input. The \( x(n) \) is predistorted by the DPD block, and this predistorted signal \( g(n) \) is transmitted through the PA. The output of PA, \( z(n) \), normalised by the small-signal (linear) gain \( G \) is provided in the DPD (coefficient estimator) block to generate \( g(n) \). The process of identifying the ‘inverse model’ is called the learning/training process. The cost function is a mean square error, minimised using linear parameter estimation methods such as least squares (LSs), least mean squares, or gradient descent methods. Once the parameters are identified, the learning/training phase is considered to be completed, and the predistorter model and coefficients are copied to the predistorter. Every data from the incoming signal is processed through this ‘predistorter.’ Such digitally predistorted data...
eventually cancels out the distortion of the PA. The $z(n)$ is non-linear with respect to the $g(n)$; however, $z(n)$ is linear with respect to the $x(n)$. Therefore, the predistortion process is also called the linearisation process [7].

As can be perceived from Figure 1, the ILA scheme has two parallel processes. Process 1 is ‘predistorter training or learning’ where the inverse model is identified based on available measured input and output of the transmitter system. The identification process for the inverse model is shown in Figure 2 [8]. The first step is to acquire the baseband complex input and the PA’s output signal using the appropriate measurement setup; afterwards, the data is de-embedded to the PA reference plane by normalising the PA output with small-signal gain. Because of the signal propagated from baseband to PA, including the transmitter and receiver system, a propagation delay occurs, introducing a mismatch between the input and output data samples. Hence, the immediate step is to estimate this delay and compensate for it. Once the delay is compensated, a suitable inverse model (behavioural model/digital predistorter) is identified. This model’s accuracy is validated using established norms such as normalised mean square error and adjacent error power ratio. As can be observed from Figure 1, that channel is not part of the problem statement, and the characteristics of the system do not change very rapidly. The ‘training process’ is generally carried out in digital signal processor (DSP) in offline mode at an intermittent interval.

The second process is the ‘application of the inverse model’. Once the model is identified, each input signal data is applied to the ‘inverse model.’ This ‘predistorted data’ is then processed via the transmitter. Due to the expansion characteristics of the inverse model, the predistorted data is divided by a constant value, which ensures that the maximum power of input signal $x(n)$ and the predistorted signal $g(n)$ are the same. It ensures that the maximum value of the signal processed through the Digital-to-analog converter (DAC) remains constant and DAC of the transmitter is not overdriven. Due to the need for continuous data processing at a high rate, the model is generally implemented in field-programmable gate array (FPGA) in the fixed-point calculation environment. The model’s coefficients are updated intermittently to account for slow changes in the transmitter/PA characteristics due to thermal effects.

The accurate inverse model for DPD depends on the transmitter and feedback receiver performance and accuracy; hence, it should be free from transceiver noise, $I/Q$ imbalance, and DC offset. The impairments such as $I/Q$ imbalance and DC offset [9–11] are well explored in the linearisation field, while the impact of noise is not much explored. The transceiver noise reduces the overall system signal-to-noise ratio (SNR), limiting the modelling performance of DPD and, ultimately, the linearisation performance. This source of noise in the transceiver may be quantisation noise of data converters, inherent noise of PA, and low-noise amplifiers. This impact may not quantify on a very high-end transceiver, but it is more prominent on low-cost systems. It should be noted that the DPD techniques are proposed to reduce the distortion introduced by the transmitter path only; therefore, noise or any property of the channel is not impacted by any of the DPD techniques. The aim is to provide a DPD technique with better numerical stability and lower complexity for low-cost transceivers utilising lower-bit DAC/Analog-to-digital converter (ADCs) and other measurement noises.

Low-cost transceivers (transmitter and feedback receiver) using lower precision/lower sampling rate also require high numerical stability and lower implementation complexity. Volterra series is the most comprehensive model to capture the non-linearity and memory effects of PA [12]. However, the Volterra series’ number of parameters increases very rapidly with the increase of non-linearity order and memory depth. The popular modelling method, which is derived from the Volterra series, is memory polynomial (MP) [2, 13], dynamic deviation reduction model (DDR), and generalised MP (GMP) [14]. DDR and GMP models containing advanced and delay memory terms are shown to have incrementally better performance over MPM with a higher number of coefficients [14]. The DDR, GMP, and MP models are based on the Vandermonde matrix (increasing polynomials order for each column). The Vandermonde matrices are known to be ill-conditioned with an increase in the order of the polynomial. As the MP model is a subset of GMP and DDR models, it has also been established that such models often have ill-conditioned observation data matrix as compared to MPM [13–15].
During the ‘DPD application’ process, in fixed-point arithmetic implementation on lower bits FPGA, the DPD performance reduces sharply compared to floating-point implementation. Polynomial models such as orthogonal MP (OMP) models are proposed to provide a solution for numerical stability problems [15]. Besides, techniques such as principal component analysis (PCA) [6, 16] independent component analysis (ICA) [17] are investigated as solutions for improving numerical stability. The lower-order polynomial models, such as b-spline and C-spline models, provide a better basis function for effective implementation on the FPGA [18, 19]. Although the numerical stability of these models is better than the Volterra-based models, the performance is still expected to deteriorate in the presence of transceiver noise.

The multiscale PCA has been proposed in the literature related to signal and image processing for the denoising process [20–22]. The multiscale PCA technique is proposed for the MP model in [23] for combating the noise in the system. The multiscale PCA technique provides encouraging results, specifically for out-of-band performance, where the PA output is more impacted by system noise. However, in the presence of noise, the issue of model’s numerical stability becomes significant. This work establishes that a noisy feedback path for DPD impacts the signal’s characteristics for which the DPD was trained. A numerically unstable model compromises the generalisation capability and may lead to limited linearisation performance. Moreover, it is also desirable for any low-cost setup to have a lower digital computational burden supported by limited DSP resources.

Therefore, this work investigates prominent polynomial-based model topologies, algorithms for enhancing the numerical stability of the LS algorithm while targeting the problem of the system noise, and low-cost fixed-point implementation for a practical, low-cost transceiver setup. With comparative analysis, it is reported for the first time that denoising along with numerically stable basis function models improves the performance on a lower resolution system leading to lower memory consumption for DPD application. The study is structured as follows: Section 2 describes the prominent behavioural modelling topologies within the polynomial framework for PA non-linearity. Section 3 describes the algorithms to enhance numerical stability. The measurement setup for proof-of-concept is described in Section 4. Section 5 presents a comparative analysis in terms of (a) in-band DPD model performance, (b) out-of-band DPD model performance, (c) robustness of DPD models for different measurement noise level, (d) robustness of DPD models in low-precision implementation environment, and (e) memory requirement for application of DPD. Section 6 showcases the experimental comparison results. The conclusion is provided in Section 7.

2 | BEHAVIOURAL MODELS

As discussed in the previous section, the ILA scheme requires ‘inverse modelling’ for DPD identification. The first requirement of the DPD application is the selection of an appropriate model topology. A polynomial-based model with time-delay terms lends itself easily for FPGA implementation. Therefore, the popular models having polynomial basis are considered as follows:

2.1 | MP model

The MP model is the most popular DPD model, which is a simplified form of the Volterra. The output signal from the MP model can be represented as [2]

\[ z(n) = \sum_{i=0}^{M} \sum_{q=0}^{N-1} c_{iq} x(n-i) |x(n-i)|^q \]  

where \( M \) is memory, and \( N \) is the non-linearity order of the PA, respectively. The \( c_{iq} \) is the model coefficient, and \( x(n) \) is the input signal.

2.2 | GMP model

The GMP model includes additional delay terms to improve the performance of the MP model. The output signal from the GMP model can be represented as [14]

\[ z(n) = \sum_{i=0}^{N_0-1} \sum_{l=0}^{N_b-1} a_{iol} x(n-i) |x(n-i)|^q + \sum_{i=0}^{N_c-1} \sum_{l=0}^{M_b-1} b_{iol} x(n-i) |x(n-i-l)|^q + \sum_{i=0}^{N_c-1} \sum_{l=0}^{M_c-1} c_{iol} x(n-i) |x(n-i-l+i)|^q \]  

where \( N_b, N_c, L_b, L_c, M_b, M_c \) are the numbers of coefficients for aligned signal and magnitude (memory polynomial), the number of coefficients for signal and delaying envelope and the number of coefficients for signal and leading envelope, respectively. \( N_b \) and \( L_b \) are the non-linearity order and delay taps for the aligned signal. Section 4.2 describes the time alignment of the input and output signals. \( N_b \) and \( N_c \) represent the non-linearity order of lagging and leading signals, \( L_b \) and \( L_c \) are the number of the delay taps for aligned signal, and \( M_b \) and \( M_c \) are the number of the delay taps for lagging signal and leading signals.

2.3 | OMP model

When the PA is highly non-linear and exhibits strong memory effects, the Volterra and its derivative polynomial models require higher non-linearity order and memory depth to represent the PA behavioural model. Such modelling leads to an ill-conditioned data matrix, limiting the generalisation capability of the solution and the effectiveness of the DPD solution. Also, the ill-conditioned data matrix leads to a high dispersion of the coefficients, which further increases the implementation complexity on FPGA in a fixed-point environment [24, 25].
An MP model of limited order is already established in the literature to represent the gains of a PA for the entire input envelope range as given by Equation (1) [18, 26]. A higher-order polynomial is required to mimic the high non-linear and saturated PA. The extraction of such models can lead to a numerically ill-conditioned regression matrix, and in the case of sparse data, a highly oscillatory solution is obtained [27].

Hence, the coefficients become extremely sensitive for this data [18], and frequent update in the system is required even if there is a small error in interpretation at input data. OMP was further proposed to enforce the orthogonality between the columns of the observation matrix of the MP model. The output of the OMP model is given as [15]

\[
\tilde{z}(n) = \sum_{q=1}^{N} \sum_{l=1}^{q} c_q (-1)^{q+l} \frac{(q + l)!}{(l - 1)! (l + 1)! (q - l)!} |x|^{q-l-1} x
\]

where \(N\) and \(c_q\) are non-linearity order and coefficient of the model.

### 2.4 | Cubic-spline (CS) model

These issues of a high-order polynomial can also be mitigated using a CS instead of MP and other regular polynomials. A piece-wise structure is more useful in comparison to high-order polynomials. The lower order polynomials do not have oscillations like the Runge phenomenon [28]. The matrix obtained using piece-wise polynomials has a smaller condition number, making the system additionally stable to errors. The CS coefficients have a local scope, that is, when an estimated value is used for a specific knot, the error introduced will only create local distortion in the shape of the spline, which does not spread to the entire input range [29, 19]. This results in a more robust extraction and is used for successive linearising data after the training is performed. Moreover, the replacement of high-order polynomial with the lower order splines allows the effective FPGA implementation. The spline is built on lower-order piece-wise polynomials, which provide interpolation between a set of points known as control points. This approach allows using lower non-linearity order and memory depth. The memory model representing the spline-based PA model can be given as [18]

\[
z(n) = \sum_{m=1}^{M} x(n-m) H(n-m)
\]

where

\[
H(n-m) = \sum_{k=1}^{K} c_{mk} \phi_{nk}(n-m)
\]

\[
\phi_{nk} = \sum_{j=0}^{3} \left[ |x(n-m)| \right. - \left. |x_k(n-m)| \right]^{j}
\]

where \(x_k(n)\) is the value of \(x(n)\) at the knot \(k\), \(K\) is the number of knots in spline, \(c_{mk}\) is the complex coefficients. Equation (6) represents the CS, which is a piece-wise polynomial of order three. The CS in Equation (6) is continuous at the knots where the splines connect. This continuity is also maintained at knots for their first and second derivatives [18]. The observation matrix contains monomials given by Equation (6).

### 3 | ALGORITHMS TO ENHANCE NUMERICAL STABILITY OF THE DPD MODELS

Equations (1) to (4) can be represented in matrix form as

\[
\tilde{z} = \tilde{W} \tilde{B}
\]

where \(\tilde{B}\) is a vector containing the model-coefficients, \(\tilde{W}\) is the observation data matrix. The LS method is used to compute \(\tilde{B}\) as

\[
\tilde{B} = \left( \tilde{W}^{H} \tilde{W} \right)^{-1} \tilde{W}^{H} \tilde{z}
\]

In this work, we have used Moore–Penrose pseudoinverse for the implementation of least squares for all the DPD models, which is based on the singular vector decomposition method.

The implementation of the above-mentioned polynomial-based models such as GMP, MP, OMP, and CS models depends on the observation matrix’s size and numerical properties. There are several popular dimension reduction techniques, such as ICA [30], t-distributed stochastic neighbour embedding [31], PCA, and so forth. This work investigates the impact of the following two established techniques on the DPD models’ performance presented in Section 2.

### 3.1 | PCA

The PCA technique has been utilised extensively for DPD application as one of the dimensionality reduction methods by using the principal component as a new set of variables [6].

The extracted principal components are orthogonal. The correlation matrix using \(\tilde{W}\) can be written as \(R = \tilde{W}^{H} \tilde{W}\). Further, Eigenvalue decomposition is applied as \(R = O \Lambda O^{H}\), where \(O = [a_1, a_2, a_3, ... a_{(M+1)(\sqrt{N}+1)}]\) denotes a matrix of eigenvectors and \(\Lambda = \text{diag}(\lambda_1, \lambda_2, \lambda_3, ... \lambda_{(M+1)(\sqrt{N}+1)})\) denotes the diagonal matrix of eigenvalues. The new length of the data matrix is selected using the weight of eigenvalues of data variance. Since the \(K\) eigenvectors have all significant variations of data, the residual data is discarded. The new principal components are created as

\[
P = [a_1, a_2, a_3, ... a_K]
\]
The new data matrix is further denoted as
\[ \vec{V} = \vec{WP} \quad (10) \]
Hence,
\[ \vec{z} = \vec{V} \vec{C} \quad (11) \]
where \( \vec{C} = [c_1, c_2, c_3, \ldots, c_L]^T \) is a coefficient vector of the PCA-MP model. Similarly, the PCA can be implemented with GMP and OMP to form GMP-PCA and OMP-PCA, respectively.

### 3.2 Multiscale PCA (MSPCA)

Wavelet-based MSPCA has been proposed in [23] to further allow denoising of the observation data matrix.

The wavelets are very useful for signal analysis and noise removal in several areas due to their capability to divide a given function into different scale components. The wavelet transform (WT) concentrates signal features in a limited large-magnitude wavelet coefficient [32]. The small value wavelet coefficients are typically noise, which can be removed without affecting the signal quality using a proper threshold level. Figure 3(a) shows the steps involved in the denoising of a signal. WT decomposition is performed by choosing a mother wavelet and the desired level for decomposition. The thresholding is performed on each level, depending on the hard or soft decision. In an inverse WT (IWT), wavelet reconstruction is performed and filtered to get the denoised signal. The MSPCA approach is implemented in the following steps [33, 34]:

1. Acquire the observation data matrix using CS as in Equation (4) and normalise each column of the observation matrix.
2. Computation of wavelet decomposition of every column in observation matrices, \( G_m W \) (\( m = 1, \ldots, K \)) along with a scaling function coefficient matrix \( H_k W \), where \( K \) is the number of scales, \( G \) and \( H \) represents filter and \( W \) is the observation data matrix as shown in Figure 3(b).
3. Apply PCA to scaling function coefficient matrices and wavelet to choose relevant principal components to determine control limits of \( T^2 \) and \( Q \) at each scale. \( T^2 \) statistics encapsulate the PCA model variation, and \( Q \) statistics encapsulate residual space generated by the PCA model.
4. Reconstruct the coefficient vector and approximated data matrix from the selected scales’ scores using the IWT.
5. Apply PCA to the reconstructed observation data matrix and retain suitable principal components.

These steps are summarised in Figure 3(b). In this case, we have used three levels, and Daubechies least-asymmetric wavelet and the number of retained principal components are selected using Kaiser rule [35], where it keeps two principal components, both for the PCA approximations and the final PCA, but one principal component is kept for details at each level. Therefore, the results can be improved by suppressing noise because the details at initial levels are composed essentially of noise with small contributions from the signal.

The selection of a mother wavelet function is a crucial step in WT, capable of denoising, component separation, coefficient reconstruction, and feature extraction from the signal in time and frequency domains depending on requirements. The common standard families of WT basis functions are Haar, Coiflets, Daubechies, and Symlets. The Daubechies wavelet is a good choice for denoising the signals [35, 36]. The Harr wavelet, which is a symmetric special case of Daubechies wavelet, is useful for edge detection. Hence, we have selected the Daubechies wavelet for our work. The decomposition is performed by successive low-pass filters (LPFs) and high-pass filters (HPFs) as shown in Figure 4(a) [20–22].

The signal sequence \( r[n] \) is passed through successive LPF and HPF. The HPF produces detailed coefficients \( k[l] \), and the LPF produces coarse approximation coefficients \( l[l] \) at the first level. Then, it is further passed through LPF and HPF, and this process lasts until the anticipated level of decomposition is achieved. The maximum number of levels depends on the length of the signal, and it can be written as
\[ P_{\text{max}} = \log_2 U \quad (12) \]
where \( U \) is the length of the signal and \( P_{\text{max}} \) is the maximum number of levels. To reconstruct the original signal, IWT is used as shown in Figure 4(b). After the necessary processing, the coefficients obtained in the decomposition are passed through
LPF and HPF for the reconstruction of the signal. In [37], the author presents a wavelet-based soft and hard threshold to denoise the data. The soft and hard threshold is given as

\[
T_h = \begin{cases} 
0 & |r| < \lambda \\
 r - \frac{\lambda}{|r|} & |r| \geq \lambda
\end{cases} 
\]

(13)

\[
T_s = \begin{cases} 
0 & |r| < \lambda \\
 r - \frac{\lambda}{|r|} & |r| \geq \lambda
\end{cases} 
\]

(14)

where 'r' is the wavelet coefficient, \( \lambda \) is the threshold level and is given as

\[
\lambda = \sigma \sqrt{2 \log C} 
\]

(15)

where \( \sigma = \frac{\text{median}(|d_j|)}{0.6745} \) and \( d_j \) is coefficients at level \( j \). \( C \) is the number of wavelet coefficients. These two types of wavelet threshold can be used on the basis of characteristics of noise. Here, the threshold is applied on \( d_j \) which corresponds to the noise component. By iteration experimentation, we found Daubechies least-asymmetric wavelet with four vanishing moments with third-level decomposition is suitable to mitigate the noise using the hard threshold.

4 | EXPERIMENTAL SETUP

For measurement and preliminary validation of the proposed CS-MSPCA, the experimental setup is shown in Figure 5. The crucial parts of the setup include: Xilinx Zynq-7000 SoC ZC706 evaluation kit, analogue device ADRV9371 radio card, ZX-V63+ PA form minicircuits, DC power supply for biasing of PA and a vector signal analyser to visualise the signal. This Software-defined radio (SDR) is a direct conversion architecture having 2 × 2 MIMO (multiple-input and multiple-output) capability. The working frequency range is 300–6000 MHz. The maximum transmitter sampling rate of the SDR is 307.2 MSPS, and the receiver sampling rate is up to 153.6 MSPS. It supports the transmitter synthesis and observation receiver bandwidth up to 250 MHz and 100 MHz bandwidth support at the receiver. It has a 12-bit auxiliary analogue-to-digital converter and 10-bit digital-to-analogue converter. The in-phase (I) and quadrature-phase (Q) signals are generated in Matlab. This signal, after generation in baseband, is uploaded to FPGA memories by SD card. The data from FPGA is sent to the digital-to-analogue converter at a sampling rate of 307.2 MSPS after it is up-converted at a frequency of 2000 MHz and sent to PA. The output of the PA is captured at a sampling frequency of 153.6 MSPS and down-converted digitally and stored in the SD card of the SDR.

4.1 | Device and signal under test

A PA ZX60-V63+ from Minicircuits with 50–6000 MHz frequency range is used having 17.8 dBm output power and 1 dB compression with a typical gain of 20.3 dB at 2500 MHz. The PA is driven by an LTE 10 MHz signal having a PAPR of 10.85 dB.

4.2 | Time-delay compensation

Before the alignment of the signal, the sampling rate of the transmitted and captured baseband signals should be matched. The transmitter baseband signal is down-sampled to 153.6 MSPS so that it can match the receiver sampling rate. The delay is compensated between baseband signals using a frequency-domain cross-correlation method [38]. The frequency-domain
cross-correlation of the input signal $x$ and output signal $\zeta$ at PA output is written as

$$R( x \ast \zeta) = R( x) \cdot R( \zeta)$$

where $R$ represents the Fourier transform. Equation (16) can be re-formulated as

$$R( x \ast \zeta) = S \cdot R( x) \cdot R( x) e^{i(2\pi f/\tau + \phi)}$$

where $S$ is the gain, $f$ is the sampling frequency, $\tau$ is the time delay, and $\phi$ is the phase difference of output signal and input signal. When $\arg(R( x \ast \zeta))$ is plotted versus frequency, $\tau$ is the linear curve slope, and $\phi$ is the intercept of the curve with the $y$-axis. The value of cross-correlation in Equation (17) is maximum when $(2\pi/\tau + \phi = 0)$; therefore, time adjusted signal is

$$z_{adj} = z e^{i(2\pi f/\tau+\phi)}$$

Figure 6(a) shows the $\arg(R( x \ast \zeta))$ versus frequency of the output signal. It can be perceived that there is a defined linear region with the constant slope in the frequency band where the signal is present as shown in Figure 6(b). This slope provides a group delay of the signal. The initial phase is given as a phase value at $f = 0$. There is no delay present in the phase (delay after compensation) as group delay, and the initial phase both are calculated accurately using a linear equation capable of providing interpolation.

## 5 PERFORMANCE COMPARISON

### 5.1 DPD model performance metrics

The impact of transceiver noise on modelling performance is studied by investigating two cases. In the first case, we utilise the measurement setup shown in Section 4 and signals/PA described in Section 4.1/4.2. The inverse characteristic of the PA is modelled using various models. The in-band performance is measured in terms of normalised mean square error (NMSE) given as

$$\text{NMSE(dB)} = 10\log_{10} \left( \frac{\sum_{n=1}^{N} |y_{\text{meas}}(n) - y_{\text{est}}(n)|^2}{\sum_{n=1}^{N} |y_{\text{meas}}(n)|^2} \right)$$

where $S$ is the sample size, $y_{\text{meas}}(n)$ is the measured output, and $y_{\text{est}}(n)$ is the estimated output. The out-of-band performance is measured in terms of adjacent channel error power ratio (ACEPR) given as

$$\text{ACEPR} = \frac{1}{2} \left( \frac{\int_{f-c}^{f+c} \left| E(f) \right|^2 df}{\int_{f-c}^{f+c} \left| Y_{\text{meas}}(f) \right|^2 df} + \frac{\int_{f-c}^{f+c} \left| E(f) \right|^2 df}{\int_{f-c}^{f+c} \left| Y_{\text{meas}}(f) \right|^2 df} \right)$$

where $f_c$ is the carrier frequency, $B$ is bandwidth, $Af$ is the frequency offset from the carrier frequency, $E[f]$ is the Discrete Fourier transform (DFT) of error $[e(n) = y_{\text{meas}}(n) - y_{\text{est}}(n)]$. $Y_{\text{meas}}(f)$ is the DFT of measured output $y_{\text{meas}}(n)$. Figure 7(a) shows that the NMSE curve of MP, MP-PCA, and MP-MSPCA models is almost the same at memory depth of $M = 3$ and non-linearity order of $N = 1$ to $N = 12$. It is to be noted that the out-of-band component of the signal is purely distortion of the signal and more close to the noise floor. Figure 7(b) shows that MP-MSPCA provides a significant improvement of up to 6 dB as compared to the MP model for out-of-band performance in terms of ACEPR. We introduced additive white Gaussian noise (AWGN) of SNR of 30 dB to the original signal for the second case. From Figure 7(a), deterioration in the model NMSE performance can be observed; however, AWGN-MP-MSPCA provides $-36$ dB NMSE as compared to the $-29$ dB NMSE for AWGN-MP and AWGN-MP-PCA models. A similar improvement trend can be observed in Figure 7(b), where AWGN-MP-MSPCA provides $-47$ dB ACEPR as compared to $-36$ dB ACEPR for MP and MP-PCA models. Although Figure 7 showcases the advantage of denoising, it is essential to test the algorithm on a range of practical SNR values to assess SNR’s influence. Therefore, we have tested it for a range of SNR values (0 to 40 dB) at fix non-linearity order of 4 and memory depth of 3. The value of

FIGURE 6  (a) Phase of $\arg(R( x \ast \zeta))$ with frequency, (b) spectrum of the signal
non-linearity is selected on the basis of Figure 7, which shows that the performance is optimum at a non-linearity order of 4.

Figure 8(a) shows the inverse modelling performance in terms of NMSE for MP, MP-MSPCA, OMP, OMP-MSPCA, and CS, CS-MSPCA. The MP, OMP, and CS inverse modelling performances decay as SNR is reduced while using the MSPCA; in all cases, the inverse modelling shows better performance (more than 5 dB reduction in NMSE) to other models, as it is able to denoise. Figure 8(b) shows the inverse modelling performance in terms of ACEPR for MP, MP-MSPCA, OMP, OMP-MSPCA and CS, CS- MSPCA, where using MSPCA, it shows approximately 20 dB at lower SNR values and more than 10 dB at higher SNR. In the case of the noisy signal, the application of the wavelet-based MSPCA method for the observation matrix of the MP model provides more than 5 dB improvement in in-band error and more than 11 dB improvement in out-of-band error. The issue of a numerically stable solution in the noisy measurement system is discussed further.

Figure 9 shows the in-band modelling performance of the different models in terms of NMSE with the change in non-linearity order. Figure 9 depicts the performance at memory

---

**Figure 7** (a) Normalised mean square error (NMSE) performance versus non-linearity order for inverse modelling at memory depth of $M = 3$, (b) adjacent channel error power ratio (ACEPR) performance versus non-linearity order for inverse modelling

**Figure 8** (a) NMSE performance versus signal-to-noise ratio (SNR) for inverse modelling (b) ACEPR performance versus SNR for inverse modelling at $N = 4$ and $M = 3$

**Figure 9** NMSE versus non-linearity order for inverse modelling
order four, and non-linearity varies from 1 to 12. The NMSE is approximately the same for MP, MP-PCA, OMP, and OMP-PCA models, while CS shows a slight improvement of approximately 0.25 dB. The coefficients are reduced further using CS-PCA. The application of the MSPCA technique provides better performance for all the models. CS-MSPCA offers the best performance. Similar trends for improvements can be observed from Figure 10, which depicts the out-of-band performance of the performance at memory order four (\( M = 4 \)), and non-linearity varies from 1 to 12. It is to be noted that although the CS-MSPCA model provides the best performance for both the in-band and out-of-band performance, this improvement is not significant. However, when this performance is combined with the model stability and lower dispersion of coefficients, it leads to a lower implementation cost.

### 5.2 Metrics for efficient DPD implementation

#### 5.2.1 Condition number

If the observation data matrix obtained in modelling is ill-conditioned (high condition number), it leads to numerical instability. Hence, it measures the worst-case loss of precision. Moreover, the b digits of precision is lost if the condition number of a matrix is 10^b during solving the system \([39]\). The higher condition problem makes the pseudoinverse calculation of the matrix very sensitive towards minor change using PCA; it is showing subspace of the PCA.

The condition number is defined as the ratio of the maximum eigenvalue (\( \lambda_{\text{max}} \)) and minimum eigenvalue (\( \lambda_{\text{min}} \)) of a matrix and can be written as

\[
\text{Cond.no.} = \frac{\lambda_{\text{max}}}{\lambda_{\text{min}}}
\]  

(21)

The condition number of the MP model is increasing with the increase in FPGA bit, and results may vary and increase in inaccuracy. OMP models are established in the literature for better conditioning of polynomial-based models \([15]\). The condition number of MP and OMP models is shown in Figure 11 for the non-linearity order of four and memory depth of four. Since the final model is achieved after pruning non-linearity order, the observation matrix of the MP model is ill-conditioned. The MP-PCA model has a better condition number than the MP model by retaining only significant principal components. The OMP model has better stability compared to MP, but the condition number increases at higher non-linearity order. MP-MSPCA can provide a much lower condition number. However, CS-MSPCA shows the least condition number because of the denoising and coefficient reduction as only the first significant eigenvalues are selected. Here, the OMP-PCA has a lesser condition number than OMP, and CS-PCA has the equivalent condition number to CS-MSPCA.

#### 5.2.2 Dispersion coefficient

Another performance metric is taken as a dispersion coefficient, which translates to the number of bits required for the fixed-point DSP implementation. It is defined as

\[
\text{Disp.Coeff.} = \frac{\max(\{e_i\})}{\min(\{e_i\})}
\]  

(22)

where \( e_i \) are the coefficients of the inverse model. The different models’ performance is shown in Figure 12 for different non-linearity orders/knots (control point for spline), with a memory depth of four chosen for different models. The dispersion coefficient of the MP-PCA and OMP models is much lower than the MP model due to the reduction in the number of coefficients. The CS model has even less dispersion coefficient, compared to the MP, MP-PCA, and OMP models. The CS-MSPCA provides
TRIPATHI AND RAWAT

5.3 Computation complexity

The computation complexity depends on the number of coefficients in the model. It increases with the increase in the non-linearity order and memory depth. Therefore, coefficient reduction techniques such as PCA are useful to reduce the size of the observation matrix. Table 1 compares the number of coefficients for different models where the non-linearity order of six and memory depth of three is used, while in the case of CS model, the non-linearity order of two and memory depth of four is used. The CS model allows the use of lower non-linearity order because of its piece-wise properties. Table 1 shows the CS-MSPCA requires the least number of coefficients in a noisy environment.

It is of interest that while inverse model extraction can be done intermittently for the DPD model, DPD application is a continuous process. Every transmitted data is processed continuously through DPD, therefore saving in computational cost enhances exponentially.

5.4 Robustness for fixed-point implementation

For the implementation of the DPD algorithm in FPGA’s fixed-point environment, bit resolution is a crucial aspect. It should be noted that even if the model extraction is carried out in the DSP utilising floating-point calculation, the extracted DPD model should be applied in FPGA for faster processing and uninterrupted transmission. Also, the application of the DPD algorithm requires a designated memory space in the FPGA. The memory size can be defined as [40]

\[
\text{Memory size} = \text{Bit resolution} \times \text{Matrix size}\] (23)

TABLE 1. Comparison of field-programmable gate array memory resources

| Model      | No. of coeff. | Bit resolution | Matrix size | Memory size (MB) |
|------------|---------------|----------------|-------------|------------------|
| MP         | 29            | 23             | 1,900,544   | 7.6              |
| MP-PCA     | 23            | 19             | 1,834,924   | 6.5              |
| MP-MSPCA   | 28            | 28             | 1,834,924   | 6.5              |
| OMP        | 37            | 37             | 1,834,924   | 6.5              |
| OMP-PCA    | 37            | 37             | 1,834,924   | 6.5              |
| OMP-MSPCA  | 37            | 37             | 1,834,924   | 6.5              |
| CS-MSPCA   | 37            | 37             | 1,834,924   | 6.5              |
| GMP        | 28            | 28             | 2,424,610   | 10.5             |
| GMP-PCA    | 28            | 28             | 2,424,610   | 10.5             |
| GMP-MSPCA  | 28            | 28             | 2,424,610   | 10.5             |

Note: MP is memory polynomial, PCA is principal component analysis, MSPCA is multiscale PCA, OMP is orthogonal MP, CS is cubic spline, GMP is generalised MP.
Figures 13(a) and (b) depict the modelling performance of MP, MP-PCA, MP-MSPCA, OMP, OMP-MSPCA, CS, and CS-MSPCA for 16, 32, 48, and 64-bit fixed-point word length. Model performance deteriorates for 16-bit fixed-point implementation. However, the CS-MSPCA model outperforms the other models significantly when the system has noise and lower memory space. Table 1 shows the corresponding memory size for the models under investigation. By processing signal at 16-bit resolution, the memory size requirement is halved as compared to the other models.

6 | COMPARISON OF LINEARISATION PERFORMANCES

The DPD for the proposed model is implemented on the testbed shown in Figure 5. An SNR of 30 dB is introduced in the captured output of the PA. The predistorted signal is generated in the Matlab, and the baseband signal is uploaded in the FPGA of the system. Figure 14 compares the power spectral density of linearised output for LTE 10 MHz signal while utilising different models for indirect learning DPD implementation. The ‘Hann’ window has been used for plotting the spectrum. The linearisation performance is investigated for in-band performance in terms of NMSE between transmitted and received data.

Out-of-band performance of the lineariser is measured in terms of an ACPR given as

\[
\text{ACPR} = \frac{1}{2} \left( \int_{-B/2}^{-B/2+\delta f} |Y(f)|^2 df + \int_{-B/2+\delta f}^{B/2} |Y(f)|^2 df \right)
\]

where \(f_c\) is the carrier frequency, \(B\) is bandwidth, \(\delta f\) is the frequency offset from the carrier frequency; here, \(\delta f = B\) for Table 2. \(Y(f)\) is the DFT of measured output \(y_{\text{meas}}(n)\). Table 2 summarises the in-band and out-of-band performances of the linearised signal for different models in the 16-bit fixed-point environment. The CS-MSPCA shows better performance as compared to the other given model. It is to be noted that when the MSPCA method is applied to the OMP model, it also provides a performance comparable to CS-MSPCA. However, Table 1 shows that the CS-MSPCA model requires a very less number of coefficients (eight coefficients) than the OMP model (22 coefficients), which eventually translates to the saving in computational cost.

7 | CONCLUSION

This study presents the performance analysis of different DPD models having polynomial-basis while keeping practical consideration in focus. The DPD methods were compared in terms of
in-band and out-of-band distortion mitigation capability, memory resource requirement, the capability to perform in a low-precision fixed-point calculation environment, and the robustness of models in the presence of measurement noise.

ACKNOWLEDGEMENTS
This work is an outcome of the project under the Visvesvaraya Ph.D. Scheme of Ministry of Electronics & Information Technology, Government of India, implemented by Digital Indian Corporation and IMPRINT-II, India under Grant IMP/2018/001569.

REFERENCES
1. Alexiou, A.: The road to 5G – visions and challenges. In: Alexiou, A. (ed.) 5G Wireless Technologies, pp. 1–15. The Institution of Engineering and Technology, Stevenage, UK (2017)
2. Ding, L., et al.: A robust digital baseband predistorter constructed using memory polynomials. IEEE Trans. Commun. 52(1), 159–165 (2004)
3. Zeleny, J., et al.: Receiver-aided predistortion of power amplifier nonlinearities in cellular networks. IET Sci. Meas. Technol. 6(3), 168–175 (2012)
4. Kwan, A., et al.: Look-up table-based digital predistorter implementation for field-programmable gate arrays using long-term evolution signals with 60 MHz bandwidth. IET Sci. Meas. Technol. 6(3), 181–188 (2012)
5. Beikmirza, M.R., Mohammadi, A., Mirzavand, R.: Power amplifier linearisation using digital predistortion and multi-port techniques. IET Sci. Meas. Technol. 10(5), 467–476 (2016)
6. Jaraut, P., Rawat, M.: Application of principal component analysis based effective digital predistorter technique for low-cost FPGA implementation. Int. J. RF Microwave Comput. Aided Eng. 27(6), e21095 (2017)
7. Theory of pth-order inverses of nonlinear systems. IEEE Trans. Circuits Syst. 23(5), 285–291 (1976)
8. Ghannouchi, F.M., Hammi O.: Behavioral modeling and predistortion. IEEE Microwave Mag. 10(7), 52–64 (2009)
9. Azza, M., Rawat, M., Ghannouchi, F.M.: Rational function based model for the joint mitigation of I/Q imbalance and PA nonlinearity. IEEE Microwave Wireless Compon. Lett. 23(4), 196–198 (2013)
10. Jaraut, P., Rawat, M., Ghannouchi, F.M.: Composite neural network digital predistortion model for joint mitigation of crosstalk, I/Q imbalance nonlinearity in MIMO transmitters. IEEE Trans. Microwave Theory Tech. 66(11), 5011–5020 (2018)
11. Rawat, M., et al.: Joint mitigation of nonlinearity and modulator imperfections in dual-band concurrent transmitter using neural networks. Electron. Lett. 49(4), 253–255 (2013)
12. Fan, C., Powers, E.J.: A new Volterra predistorter based on the indirect learning architecture. IEEE Trans. Signal Process. 45(1), 223–227 (1997)
13. Kim, J., Konstantinou, K.: Digital predistortion of wideband signals based on power amplifier model with memory. Electron. Lett. 37(23), 1417–1418 (2001)
14. Morgan, D.R., et al.: A generalized memory polynomial model for digital predistortion of RF power amplifiers. IEEE Trans. Signal Process. 54(10), 3852–3860 (2006)
15. Raich, R., Qian, H., Zhou, G.T.: Orthogonal polynomials for power amplifier modeling and predistorter design. IEEE Trans. Veh. Technol. 53(5), 1468–1479 (2004)
16. Gilbert, P.L., et al.: Order reduction of wideband digital predistorters using principal component analysis. In: IEEE MTT-S International Microwave Symposium Digest, Seattle, WA, pp. 1–4 (2013)
17. Jaraut, P., et al.: Independent component analysis for multi-carrier transmission for 4G/5G power amplifiers. In: 89th ARFTG Microwave Measurement Conference, Honolulu, HI, pp. 1–4 (2017)
18. Naraharisetti, N., et al.: Efficient least-squares 2-D cubic spline for concurrent dual-band systems. IEEE Trans. Microwave Theory Tech. 63(7), 2199–2210 (2015)
19. Wu, X., et al.: A spline-based Hammerstein predistorter for 3G power amplifiers with hard nonlinearities. In: IEEE International Conference on Future Computer and Communication, Wuhan, China, pp. V3-741–V3-745 (2010)
20. Allabakash, S., et al.: Wavelet transform-based methods for removal of ground clutter and denoising the radar wind profiler data. IET Signal Process. 9(5), 440–448 (2015)
21. Allabakash, S., et al.: Intermittent clutter suppression method based on adaptive harmonic wavelet transform for L-band radar wind profiler. IEEE Trans. Geosci. Remote Sens. 57(11), 8546–8556 (2019)
22. Shen, Y., et al.: Wavelet-based total variation and nonlocal similarity model for image denoising. IEEE Signal Process. Lett. 24(6), 877–881 (2017)
23. Tripathi, G.C., Rawat, M.: Robust linearization of power amplifier in the presence of transceiver noise. In: IEEE Asia-Pacific Microwave Conference, Singapore, pp. 1655–1657 (2019)
24. Wilkinson, J.H.: The Algebraic Eigenvalue Problem. Clarendon Press, Oxford, UK (1995)
25. Beckermann, B.: The condition number of real Vandermonde, Krylov and positive definite Hankel matrices. Numer. Math. 85(4), 553–577 (2000)
26. Younes, M., et al.: Two-dimensional piecewise behavioral model for highly nonlinear dual-band transmitters. IEEE Trans. Ind. Electron. 64(11), 8666–8675 (2017)
27. Safari, N., Holte, N., Roste, T.: Digital predistortion of power amplifiers based on spline approximations of the amplifier characteristics. IEEE Vehicular Technology Conference, Baltimore, MD, pp. 2075–2080 (2007)
28. De Boor, C.: A Practical Guide to Splines, 3rd ed. Springer-Verlag, Berlin, Germany (1978)
29. Barradas, F., et al.: Higher locality nonlinear basis functions of volterra series based models to improve extraction conditioning. In: IEEE MTT-S International Microwave Symposium, Tampa, FL, pp. I–4 (2014)
30. Hyvärinen, A., Karhunen, J., Oja, E.: Independent Component Analysis. John Wiley & Sons, Hoboken (2002)
31. Maaten, L.V., Hinton, G.: Visualizing data using t-SNE’. J. Mach. Learn. Res. 9, 2579–2605 (2008)
32. Chui, C.K.: An Introduction to Wavelets. Elsevier, Amsterdam (2016)
33. Bakshi, R.R.: Multiscale PCA with application to multivariate statistical process monitoring AICHE J. 44(7), 1596–1610 (1998)
34. Kano, M., et al.: Comparison of multivariate statistical process monitoring methods with applications to the Eastman challenge problem. Comput. Chem. Eng. 26(2), 161–174 (2002)
35. Daubechies, I.: Ten Lectures on Wavelets. Siam, Philadelphia (1992)
36. Vidakovic, B.: Statistical Modeling by Wavelets. John Wiley & Sons, New York (2009)
37. Donoho, D.L.: Denoising by soft thresholding. IEEE Trans. Inf. Theory 41(3), 613–627 (1995)
38. Tripathi, G.C., Rawat, M.: Delay compensation for 4G/5G transmitter system characterization. Microwave Opt. Technol. Lett. 59(8), 1887–1890 (2017)
39. Cheney, E.W., Kincaid, D.R.: Numerical Mathematics and Computing, 7th ed. Brooks/Cole, Pacific Grove, CA, USA (2008)

40. Quindroit, C., et al.: FPGA implementation of orthogonal 2D digital predistortion system for concurrent dual-band power amplifiers based on time-division multiplexing. IEEE Trans. Microwave Theory Tech. 61(12), 4591–4599 (2013)

How to cite this article: Tripathi GC, Rawat M. Analysing digital predistortion technique for computation-efficient power amplifier linearisation in the presence of measurement noise. IET Sci Meas Technol. 2021;15:398–410. https://doi.org/10.1049/smt2.12041