

# A Block-based LMS using the Walsh Transform for Digital Predistortion of Power Amplifiers

Maxandre Fellmann, François Rivet, Nathalie Deltimple, Eric Kerhervé, Hervé Lapuyade, Yann Deval  
Univ. Bordeaux, CNRS, Bordeaux INP, IMS, UMR 5218, F-33400, Talence, France

**Abstract**—A novel non-linear adaptive filter for the linearization of Radio Frequency (RF) Power Amplifiers (PAs) is presented. In this study, we aim at reducing the Digital Predistortion (DPD) complexity and enhancing its convergence speed for reduced computation time. The Walsh Transform is used as a computational basis for evaluating a predistorter (PD) model. The mathematical properties of the Walsh theory are exploited to adapt a memory polynomial (MP) in the sequency domain. A block-based Walsh LMS is introduced to seek the optimal PD coefficients. Simulations and results of linearization of class-AB PAs are exhibited. The comparison with conventional DPD algorithms shows that the proposed method converges 10 times faster with a reduction of 12% of the complexity for similar accuracy. Finally, a complete DPD architecture based on the Walsh Transform is proposed.

**Index Terms**—Digital Predistortion, Walsh Transform, Non-linear Adaptive Filtering, Power Amplifier (PA).

## NOTATION AND DEFINITION

Scalars are in italics and vectors or matrices are in bold:

$$\mathbf{x} = \begin{pmatrix} x[1] \\ x[2] \\ \vdots \\ x[N] \end{pmatrix}$$

For complex data, the transpose  $(\cdot)^T$  must be replaced by the Hermitian (complex conjugate) transpose  $(\cdot)^H$ .

Uppercase variables correspond to the Fast Walsh Transform (FWT) of lowercase variables:

$$\mathbf{X} = W\{\mathbf{x}\}$$

The logical convolution, also known as dyadic convolution is defined by the symbol  $\otimes$ .

## I. INTRODUCTION

**D**IGITAL communications techniques (OFDM, carrier aggregation) increase considerably the spectral efficiency of communications to meet their exponential growth. However, these techniques generate signals with a high Peak-to-Average-Power Ratio (PAPR). They are very sensitive to the non-linearities and memory effects introduced by the Power Amplifier (PA) [1], [2]. These effects distort the amplitude and phase of the useful signal, leading to asymmetric spectral regrowth in adjacent communication channels [3]–[7] and degradation of the error vector magnitude (EVM) [8]. Linearization techniques minimize distortions and allow the

PA to operate near their saturation point which improves the linearity versus power added efficiency (PAE) trade-off [9], [10]. DPD techniques have been investigated in the past decades as powerful linearization solutions. These methods invert amplitude, phase, and memory effects non-linearities with a predistorter (PD). Depending on the position of the PD, the DPD can operate in baseband (BB) as shown in Fig. 1, intermediate frequencies (IF), or radio frequencies (RF) [11].

The most adopted method is the BB pre-distortion which operates at the lowest operating frequency compared to IF and RF and can be easily implemented on a DSP or an FPGA. In the literature, the derived functions of the Volterra series [12], [13] are conventionally used to model the dynamic behavior of the PA as well as its inverse characteristic.

These include memory polynomial (MP) [14], [15], generalized memory polynomial (GMP) [16], Hammerstein and Wiener models [17] and neural networks [18]. However, the number of required coefficients becomes larger as the complexity of the models increases to reach mandatory EVM and adjacent channel power ratio (ACPR). This results in a time-consuming process to compute the DPD. Researches have focused on improving PD mathematical models [19]–[21] in an effort to reduce the number of terms to save time without losing accuracy. Another way to reduce the processing time is to lower the complexity and enhance the convergence speed of Least Mean Square algorithms (LMS) [15], [22] or Recursive Least Square algorithms (RLS) [23], [24], also known as adaptive filters, which compute the PD terms. While these methods provide a scalar approach to data processing, other comparable algorithms employ a vector approach [25] to enhance their performance, at the expense of hardware resources [26]. To address this concern, researchers have proposed the use of transforms, such as the Fourier Transform [25], [27].

In this paper, we use a block LMS associated with the Walsh Transform [28] to compute the coefficients of linear and non-linear adaptive filters with a 10-time faster convergence and 12% reduction of complexity compared to conventional scalar LMS approaches. Also, similar linearization performances to these methods are obtained with an MP PD computed with the proposed method. One potential application of the proposed method is the implementation of DPD in energy-efficient devices like smartphones. The method enables the rapid development of a straightforward PD MP model, which may not necessarily deliver the best DPD performances. Nevertheless, this model is effective in meeting spectral pollution regulations while keeping costs low, as it does not require an excessive

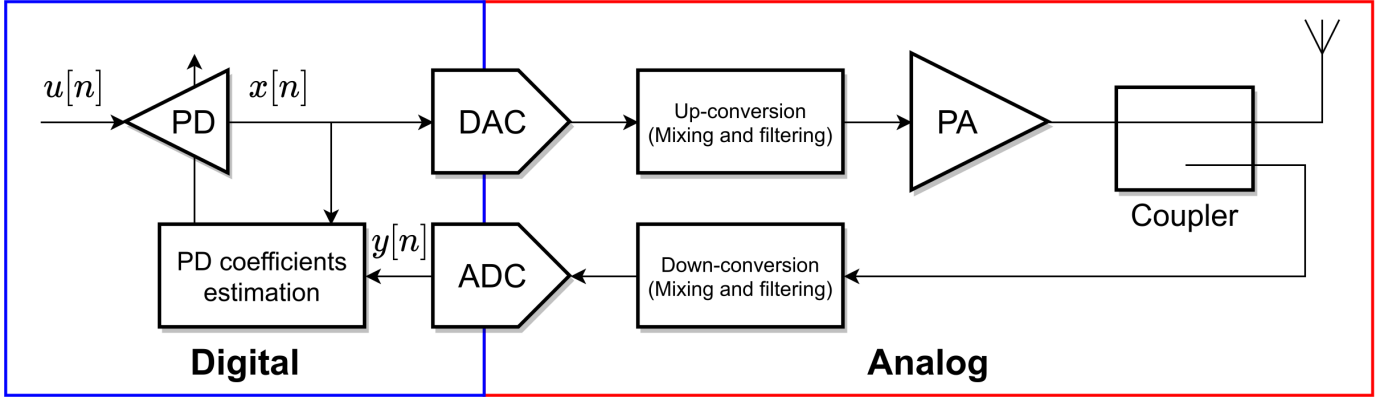


Fig. 1. Typical baseband DPD architecture.

amount of energy for computation.

The paper is organized as follows: Section II introduces a block-based LMS using the Walsh Transform. Section III presents simulations and performance results of the proposed adaptive filter. In Section IV, experimental results are presented as a proof of concept (PoC). Section V presents a complete DPD architecture based on the Walsh Transform. Finally, conclusions and future work are stated in Section VI.

## II. THE WALSH-BASED BLOCK LMS

### A. The Walsh Transform

The Walsh Transform and the Fourier Transform, among other transformations, are a possible base of the adaptive filtering process. The Walsh functions  $\mathbf{W}$  are an ordered set of rectangular waveforms and only take the values  $+1$  and  $-1$ . Figure 2 displays the first 8 Walsh sequences. The binary nature of the sequences makes them very suitable for digital implementation. Therefore, the discrete Walsh Transform (DWT) described by Eq. 1 can be computed much faster than the discrete Fourier Transform (DFT).

$$X[i] = \sum_{n=1}^N x[n]W[i, n] \quad (1)$$

The Walsh Transform decomposes the time-discrete signal into a sequency-discrete signal. The set of Walsh coefficients

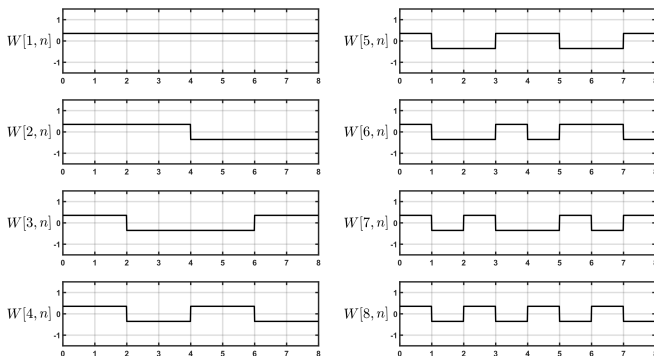


Fig. 2. The first 8 Walsh sequences.

$X[i]$  represents the sequency spectrum [29] of  $x[n]$  in the same sense that a set of Fourier coefficients represents a frequency spectrum. The sequency domain convolution, also known as dyadic convolution [29], is described in Eq. 2.1:

$$\mathbf{Z} = \mathbf{X} \circledast \mathbf{Y} \iff Z[i] = \frac{1}{N} \sum_{j=0}^{N-1} X[j \oplus i]Y[j] \quad (2.1)$$

With  $j \oplus i$  the dyadic sum of integers  $j$  and  $i$ :

$$j \oplus i = \sum_{b=0}^{\text{number of bits}} |j_b - i_b|2^b \quad (2.2)$$

The Walsh Transform also has analog properties to the Fourier Transform. A dyadic convolution in the Walsh domain is equivalent to a point-wise multiplication in the time domain.

$$\mathbf{W}\{\mathbf{x}\mathbf{y}\} = \mathbf{X} \circledast \mathbf{Y} \quad (3.1)$$

Reciprocally:

$$\mathbf{W}\{\mathbf{x} \circledast \mathbf{y}\} = \mathbf{X}\mathbf{Y} \quad (3.2)$$

An  $N$ -point DWT can also be performed with matrix multiplication as an  $N$ -point Fast Walsh Transform (FWT). As an example for  $N = 4$ :

$$\mathbf{X} = \mathbf{x} \times \mathbf{W}_4 \quad (4.1)$$

With:

$$\mathbf{W}_4 = \frac{1}{\sqrt{4}} \begin{pmatrix} + & + & + & + \\ + & + & - & - \\ + & - & - & + \\ + & - & + & - \end{pmatrix} \quad (4.2)$$

The Walsh matrix is symmetric, hence  $\mathbf{W} = \mathbf{W}^T$ . Moreover, as the Walsh sequences constitute an orthonormal basis we have:

$$\mathbf{W}\mathbf{W}^T = \mathbf{I} \iff \mathbf{W}\mathbf{W} = \mathbf{I} \quad (5)$$

Therefore  $\mathbf{W} = \mathbf{W}^T = \mathbf{W}^{-1}$ . A Walsh power spectrum can also be computed. The Walsh power spectrum  $\mathbf{P}$  of a signal

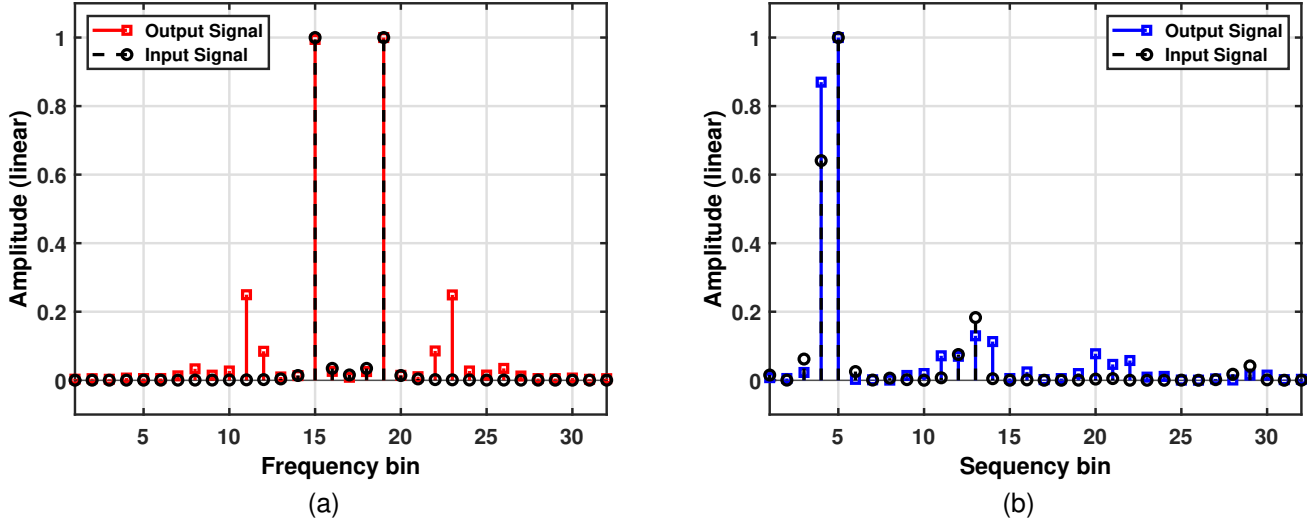


Fig. 3. (a) Normalized Fourier power spectra and (b) normalized Walsh power spectra of a two-tone input (black) and output (red/blue) of a non-linear system.

$x$  can be obtained after averaging  $A$  energy spectra windows  $S_a[i] = |X[i]|^2$ :

$$P[i] = \frac{1}{A} \sum_{a=1}^A S_a[i] \quad (6)$$

This theorem is known as the sequency Wiener-Khinchine theorem [29]. Comparisons of Fourier power spectra and Walsh power spectra are illustrated in the following example.

The input signal  $x$  is composed of two tones at frequency bins  $f_b[1] = 15$  and  $f_b[2] = 19$ . A non-linear function (representing a PA) is applied to the signal. The Fourier power spectra of  $x$  and  $y$  are shown in Fig. 3(a). The non-linearities of the PA create intermodulation products (IM) in  $y$  at frequency bins  $2f_b[1] - f_b[2] = 11$  and  $2f_b[2] - f_b[1] = 23$ .

Similarly, as illustrated in Fig. 3(b), the Walsh power spectra also display intermodulation products at several sequency bins. From this result, one can extract the non-linearities of a system to feed a non-linear adaptive filtering method based on the Walsh Transform as described in the following sub-section.

### B. The Walsh block LMS (WLMS)

WLMS algorithm combines the convergence properties of an overlap-save block-based LMS algorithm with the Walsh Transform to reduce its computational complexity. The block LMS algorithm is fully described in Appendix A-D, it consists of the following set of equations:

$$\mathbf{y}_j = \boldsymbol{\chi}_j \times \mathbf{c}_j \quad (7.1)$$

$$\mathbf{e}_j = \mathbf{f}(\mathbf{d}_j - \mathbf{y}_j) \quad (7.2)$$

$$\mathbf{c}_{j+1} = \mathbf{c}_j + 2\mu\boldsymbol{\chi}_j^T \mathbf{e}_j \quad (7.3)$$

With  $\boldsymbol{\chi}_j$  the circulant matrix of the input data set  $\mathbf{x}_j$ ,  $\mathbf{d}_j$  the desired output of the model,  $\mathbf{e}_j$  the error vector,  $\mathbf{c}_j$  the coefficients of the model and  $\mathbf{f}$  a diagonal window matrix.

The following base change formula is applied to pass  $\boldsymbol{\chi}_j$  into the Walsh domain :

$$\mathbf{X}_j = \mathbf{W}_{2N} \boldsymbol{\chi}_j \mathbf{W}_{2N}^{-1} \quad (8)$$

Where  $\mathbf{W}_{2N}$  is the symmetric matrix that applies a  $2N$ -DWT. Since  $\boldsymbol{\chi}_j$  is circulant, it has been shown in [30] that  $\mathbf{X}_j$  is diagonal by part and antisymmetric. The structure of  $\mathbf{X}_j$  is shown in Fig. 4(a) for  $2N = 8$ . Depending on the application, the Walsh sequences forming  $\mathbf{W}_{2N}$  can be arranged in an order called the Hadamard order [31]. Figure 4(d) illustrates the Hadamard matrix structure  $\mathbf{H}_{2N}$ . Using the Hadamard matrix in Eq. 8,  $\mathbf{X}_j$  becomes block-diagonal and antisymmetric [30].

Using this diagonal structure, the number of non-zero coefficients ( $N_Z$ ) of  $\mathbf{X}_j$  can be deduced. It is defined in Eq. 9.

$$N_Z = 2 + \sum_{n=1}^{\log_2(2N/4)} 2^{2n} \quad (9)$$

$\mathbf{D}_j$ ,  $\mathbf{Y}_j$ ,  $\mathbf{E}_j$ ,  $\mathbf{C}_j$  correspond to the  $2N$ -DWT of  $\mathbf{d}_j$ ,  $\mathbf{y}_j$ ,  $\mathbf{e}_j$ ,  $\mathbf{c}_j$  respectively. The index  $2N$  in  $\mathbf{W}_{2N}$  no longer appears in the following equations for simplicity. The output of the filter in the WLMS is:

$$\mathbf{Y}_j = \mathbf{X}_j \mathbf{C}_j \quad (10)$$

Since  $\mathbf{X}_j$  is block-diagonal and antisymmetric, the number of multiplications  $N_M$  to process Eq. 10 is equal to  $N_Z$  and the number of additions  $N_A$  is:

$$N_A = 2 + \sum_{n=1}^{\log_2(2N/4)} 2^n(2^n - 1) \quad (11)$$

The error vector in the Walsh domain is:

$$\begin{aligned} \mathbf{E}_j &= \mathbf{Wf}(\mathbf{d}_j - \mathbf{y}_j) \\ \mathbf{E}_j &= \mathbf{Wf}(\mathbf{W}^{-1}\mathbf{Wd}_j - \mathbf{W}^{-1}\mathbf{WY}_j) \end{aligned} \quad (12.1)$$

By factoring by  $\mathbf{W}^{-1}$ :

$$\mathbf{E}_j = \mathbf{F}(\mathbf{D}_j - \mathbf{y}_j) \quad (12.2)$$

with:

$$\mathbf{F} = \mathbf{W} \mathbf{f} \mathbf{W}^{-1} \quad (12.3)$$

According to the gradient method, the updating equation of the filter coefficients is:

$$\begin{aligned} \mathbf{C}_{j+1} &= \mathbf{C}_j - \mu \hat{\nabla} \mathbf{C}_j \\ &= \mathbf{C}_j - \mu \frac{\partial |E_j|^2}{\partial \mathbf{C}_j} \end{aligned} \quad (13.1)$$

with  $\hat{\nabla} \mathbf{C}_j$  the estimated gradient of the squared error in the Walsh domain. The squared error is defined as:

$$|E_j|^2 = \mathbf{E}_j^T \mathbf{E}_j = (\mathbf{D}_j^T - \mathbf{Y}_j^T) \mathbf{F}^T \mathbf{F} (\mathbf{D}_j - \mathbf{Y}_j) \quad (13.2)$$

$f$  is diagonal and  $\mathbf{W} = \mathbf{W}^{-1} = \mathbf{W}^T$  (Eq. 5), therefore:

$$\mathbf{F}^T = (\mathbf{W}^{-1})^T \mathbf{f}^T \mathbf{W}^T = \mathbf{W} \mathbf{f} \mathbf{W}^{-1} = \mathbf{F} \quad (13.3)$$

Moreover, since  $f$  has its right  $N \times N$  bottom corner equal to  $\mathbf{I}$ , it is known that  $\mathbf{f} \mathbf{f} = \mathbf{f}$ . Therefore, it can be stated:

$$\begin{aligned} \mathbf{F}^T \mathbf{F} &= \mathbf{F} \mathbf{F} \\ &= \mathbf{W} \mathbf{f} \mathbf{W}^{-1} \mathbf{W} \mathbf{f} \mathbf{W}^{-1} = \mathbf{W} \mathbf{f} \mathbf{W}^{-1} = \mathbf{F} \end{aligned} \quad (13.4)$$

Equation 13.2 can be restated as:

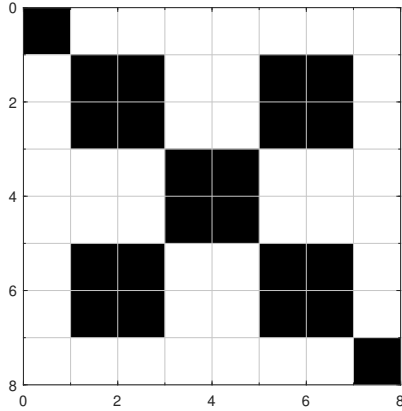
$$\begin{aligned} |E_j|^2 &= (\mathbf{D}_j^T - \mathbf{C}_j^T \mathbf{X}_j^T) \mathbf{F} (\mathbf{D}_j - \mathbf{X}_j \mathbf{C}_j) \\ &= \mathbf{D}_j^T \mathbf{F} \mathbf{D}_j - 2 \mathbf{D}_j^T \mathbf{F} \mathbf{X}_j \mathbf{C}_j + \mathbf{C}_j^T \mathbf{X}_j^T \mathbf{F} \mathbf{X}_j \mathbf{C}_j \end{aligned} \quad (13.5)$$

We assume that  $\mathbf{X}_j$  and  $\mathbf{D}_j$  are independent of  $\mathbf{C}_j$  and that the system is time-invariant. Taking the partial derivative of the squared error in regard to  $\mathbf{C}_j$ :

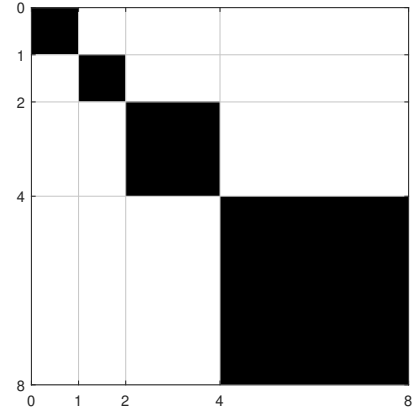
$$\begin{aligned} \frac{\partial |E_j|^2}{\partial \mathbf{C}_j} &= 0 - 2 \mathbf{D}_j^T \mathbf{F} \mathbf{X}_j + 2 \mathbf{C}_j^T \mathbf{X}_j^T \mathbf{F} \mathbf{X}_j \\ &= -2 \mathbf{F} \mathbf{X}_j (\mathbf{D}_j^T - \mathbf{C}_j^T \mathbf{X}_j^T) \\ &= -2 \mathbf{X}_j^T \mathbf{F} (\mathbf{D}_j - \mathbf{X}_j \mathbf{C}_j) = -2 \mathbf{X}_j^T \mathbf{E}_j \end{aligned} \quad (13.6)$$

Therefore the updating equation of the filter coefficients in the WLMS is:

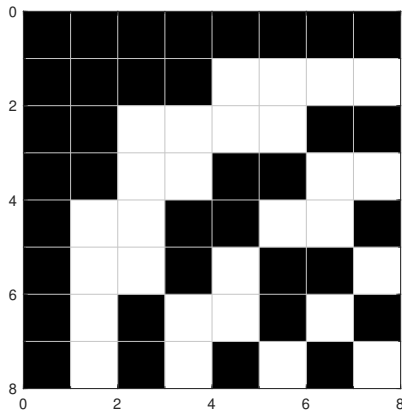
$$\mathbf{C}_{j+1} = \mathbf{C}_j + 2\mu_W \mathbf{X}_j^T \mathbf{E}_j \quad (13.7)$$



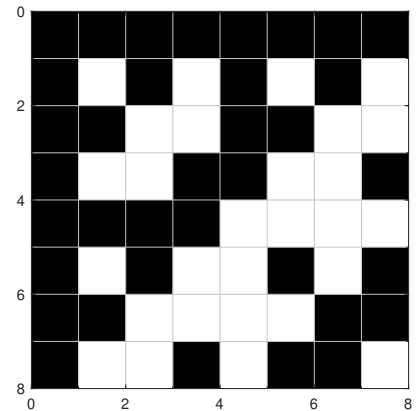
(a)



(b)



(c)



(d)

Fig. 4. (a)  $\mathbf{X}_j$  with Walsh ordering (black  $\neq 0$ , white = 0). (b)  $\mathbf{X}_j$  with Hadamard ordering (black  $\neq 0$ , white = 0). (c) Walsh matrix  $\mathbf{W}_8$  (black = 1, white = -1) (d) Hadamard matrix  $\mathbf{H}_8$  (black = 1, white = -1).

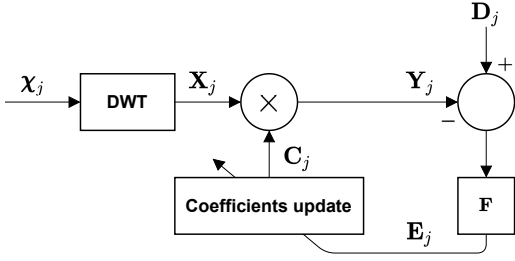


Fig. 5. WLMS block diagram.

$\mu_W$  has the same purpose as  $\mu$  in the conventional LMS algorithm. Figure 5 shows the WLMS block diagram.

In the WLMS, the estimated gradient of the squared error is computed as an average of the data instead of its instantaneous value as in the scalar time domain approach. Therefore, it is a more accurate representation of the true gradient and leads to faster convergence of the algorithm [25].

Equation 12.2 and 13.7 perform a linear system identification. However, in the case of DPD, the adaptive filter is non-linear. As explained in the introduction, the derived functions of the Volterra series are conventionally used to model a non-linear system. Therefore, for a scalar LMS approach, given by Eq. 26.1 to Eq. 26.3 in Appendix A-A can be restated as:

$$y_k = \sum_{q=1}^Q \sum_{m=0}^{M-1} c_{qm} x_{k-m} |x_{k-m}|^{q-1} \quad (14.1)$$

$$y_k = \mathbf{x}_k \mathbf{c}_k^T$$

where  $Q$  is the non-linearity order of the system,  $M$  the memory depth and  $\mathbf{x}_k$  the state vector of MP input samples defined as:

$$\mathbf{x}_k^T = (x_k, \dots, x_k |x_k|^{Q-1}, \dots, x_{k-M+1} |x_{k-M+1}|^{Q-1}) \quad (14.2)$$

To present the form of the block LMS needed in the WLMS, the input data set is segmented into  $Q$  vectors (one for each  $q$  order) of length  $2N$  points vectors that overlap on the last  $N$  points every  $j$  iteration. Also,  $M$  is assumed to be less than or equal to  $N$ . If  $M$  is not a power of two,  $N$  is the nearest higher power of 2 to  $M$ .

$$\mathbf{x}_{j,q}[n] = x[(j-1)N+n] |x[(j-1)N+n]|^{q-1} \quad (15)$$

Therefore, the adaptive filter is  $\mathbf{c}_j$  and is also segmented into  $Q$  sub-adaptive filters of length  $2N$  with their last  $2N - M$  values set to 0.

$$\mathbf{c}_{j,q}^T = (c_{j,q0}, c_{j,q1}, \dots, c_{j,qM-1}, 0, \dots, 0) \quad (16)$$

The final output vector is the sum of each circular convolution between  $\mathbf{x}_{j,q}$  and  $\mathbf{c}_{j,q}$ :

$$\mathbf{y}_j = \sum_{q=1}^Q \mathbf{x}_{j,q} \times \mathbf{c}_{j,q} \quad (17)$$

With its last  $N$  points being the result of the linear convolution. The error vector is the same as in Eq. 33 and the update of the coefficients is defined by:

$$\mathbf{c}_{j+1,q} = \mathbf{c}_{j,q} + 2\mu \mathbf{x}_{j,q}^T \mathbf{e}_j \quad (18)$$

The WLMS equations for the identification of an MP model are derived by running Eq. 16 to Eq. 18 through the same process as described above.

The base change:

$$\mathbf{X}_{j,q} = \mathbf{W} \mathbf{x}_{j,q} \mathbf{W}^{-1} \quad (19)$$

The model adaptation:

$$\mathbf{Y}_j = \sum_{q=1}^Q \mathbf{X}_{j,q} \mathbf{C}_{j,q} \quad (20)$$

The error equation is the same in Eq. 12.2. The update of each sub-filter's coefficients is Eq. 21.

$$\mathbf{C}_{j+1,q} = \mathbf{C}_{j,q} + 2\mu_W \mathbf{X}_{j,q}^T \mathbf{E}_j \quad (21)$$

### III. SIMULATION RESULTS AND COMPARISON WITH OTHER TECHNIQUES

This section presents a comparison between the WLMS algorithm and conventional algorithms: LMS, Normalized LMS (NLMS), and Recursive Least Square (RLS). For each algorithm, the study assesses various factors such as convergence speed, accuracy, sensitivity to noise, and complexity. Additional details on the conventional methods can be found in Appendix A.

#### A. Convergence speed, accuracy, and noise simulations

First, the accuracy and convergence speed are evaluated throughout the Normalized Mean Square Error (NMSE) given by Eq. 22.

$$\text{NMSE} = \frac{\sum_{j=1}^J |d_j - y_j|^2}{\sum_{j=1}^J |d_j|^2} \quad (22)$$

The system to be identified is a FIR filter of length  $N = 32$ . The input is a complex randomized data set drawn from the standard normal distribution and quantified on 10 bits. For each algorithm, the convergence constant was chosen to achieve the fastest convergence rate without causing the algorithm to diverge:  $\mu = 0.1$  (LMS),  $\mu_n = 1$   $\alpha = 3$  (NLMS),  $\delta = 1$   $\lambda = 0.97$  (RLS),  $\mu_W = 0.01$  (WLMS).

The NMSE evolution of each algorithm was evaluated over  $J = 1000$  iterations. This experience was repeated 100 times to evaluate the average behavior of each algorithm. Figure 6 displays the simulation results.

One can notice from the results that the RLS and WLMS algorithms demonstrate a much faster convergence rate compared to the LMS and NLMS algorithms. To achieve an NMSE of -40dB, RLS and WLMS required only 150 and

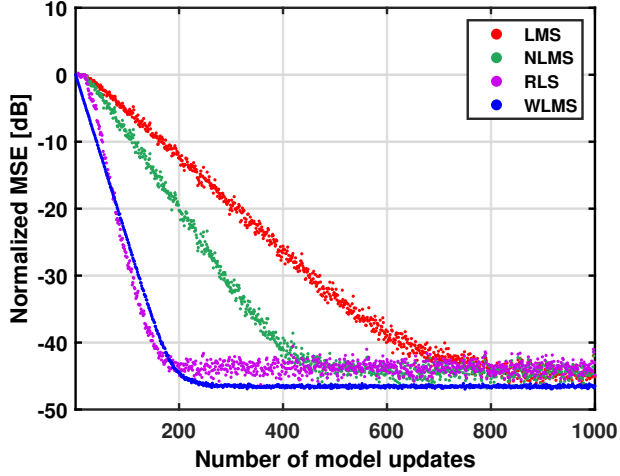


Fig. 6. NMSE of LMS, NLMS RLS and WLMS over 1000 model updates.

160 iterations, respectively, while LMS and NLMS took 400 and 600 iterations, respectively. Furthermore, the minimum NMSE threshold, which is noise and quantization dependent [32], [33] is comparatively lower in the WLMS than in LMS, NLMS, and RLS. The reason for this is that the estimation of the error gradient in WLMS is calculated as an average of the data rather than the instantaneous value, effectively minimizing the influence of noise. Consequently, fluctuations in the NMSE are significantly reduced in WLMS compared to the other methods.

Although this study initially focused on linear system estimation through an equation, a second study was conducted to benchmark the DPD algorithms for the estimation of MP models to linearize a PA. The simulation utilized an NXP Airfast LDMOS PA with an operating frequency of 3.6-3.8GHz, 29dB gain, and an output compression point at 47dBm with inherent non-linearities and memory effects due to its Doherty topology. The PA model was available in Mathworks [34]. The study used two different inputs. The first input was a 16-QAM Root Raised Cosinus filter (RRC) with 50MHz bandwidth, 5.6dBm input power, and 7.5dB PAPR. The second input was a 64-QAM OFDM, 50MHz bandwidth, 2.7dBm input power, and 10.7dB PAPR. For both inputs, the parameters of the PD MP model were  $Q = 5$  and  $M = 8$ . The results of the study are presented in Fig. 7(a) and Fig. 7(b), which illustrate the power spectra of the PA output and the evolution of the ACPR reduction for each algorithm over 20 iterations with the second input. For each algorithm, the convergence parameters were chosen to achieve the fastest convergence rate without causing the algorithm to diverge:  $\mu = 0.02$  (LMS),  $\mu_n = 0.5$   $\alpha = 3$  (NLMS),  $\delta = 1$   $\lambda = 1$  (RLS),  $\mu_W = 0.08$  (WLMS).

The parameters of the PD are equal in the four algorithms, therefore the PD coefficients will converge to the same values at some point, leading to the same ACPR reduction. However, as illustrated in Fig. 7(b), the WLMS has the second-fastest convergence speed and accuracy for a finite number of iterations with the chosen parameters. Also, a few non-symmetric spectral regrowths, resulting from the memory

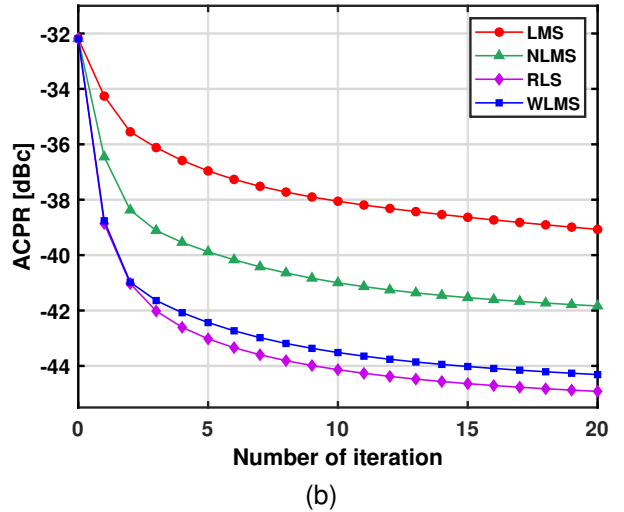
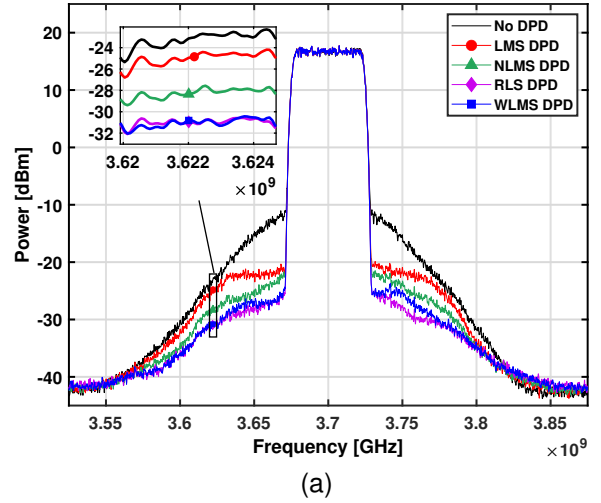


Fig. 7. (a) Power spectrum of PA output with and without DPD after 20 iterations of each algorithm. (b) ACPR evolution over 20 iterations of each algorithm.

effect of the PA [5]–[7], remain after the application of DPD. This phenomenon arises from the attempt to compensate for the memory effect by employing an inverse memory effect generated by the PD within the MP model. Due to the limited number of iterations in our study, the obtained model does not fully compensate for the memory effect. Table I presents a comparison of the performance of the MP DPD with each algorithm for both modulation schemes. The evaluated metrics are the ACPR over two different adjacent bandwidths (50MHz and 100MHz), NMSE and EVM.

From the Table I results, one can notice that RLS DPD generally provides the best performance in both modulation schemes, achieving the lowest ACPR, NMSE, and EVM values. However, it is important to note that the WLMS outperforms the LMS/NLMS methods in the given scenarios.

As a first additional study, the behavior of the WLMS DPD was examined in both modulation scenarios by adjusting the parameters  $Q$  and  $M$ . Detailed simulation results are provided

TABLE I  
ACPR, NMSE AND EVM COMPARISONS AFTER 20 ITERATIONS OF LMS, NLMS, RLS, AND WLMS DPD FOR 2 DIFFERENT INPUTS.

Waveform	16-QAM RRC				64-QAM OFDM			
	ACPR L/U (dBc)		NMSE (dB)	EVM (%)	ACPR L/U (dBc)		NMSE (dB)	EVM (%)
	50MHz	100MHz			50MHz	100MHz		
No DPD	-34.5 / -34.3	-41.0 / -41.5	/	6.37	-32.2 / -32.5	-41.5 / -42.4	/	7.21
LMS DPD	-40.3 / -39.3	-45.2 / -45.1	-32.2	4.62	-39.1 / -38.9	-44.5 / -45.1	-30.3	4.36
NLMS DPD	-42.0 / -40.8	-48.1 / -47.1	-33.3	4.47	-41.8 / -41.1	-48.3 / 48.0	-31.6	4.05
RLS DPD	-45.6 / -45.4	-51.3 / -51.0	-37.9	3.54	-45.0 / -44.8	-50.6 / -50.7	-35.3	0.69
WLMS DPD	-45.2 / -43.9	-50.4 / -49.9	-36.4	3.62	-44.3 / 43.3	-50.0 / -47.7	-34.51	0.72

in Appendix B-A. Notably, for a wide range of parameter values, the WLMS algorithm exhibited consistent and stable behavior in both modulation schemes.

A second additional study investigated the impact of noise and SNR on the WLMS algorithm. The simulation was performed by varying the SNR range from 25dB to 50dB and the number of quantization bits from 5 to 16. The results of the simulation are presented in Appendix B-B. The WLMS algorithm demonstrated a stable behavior and did not diverge in the presence of noise. It was able to effectively linearize the PA at an SNR of 35dB and with quantization of 7 bits of data.

### B. Complexity

This section provides a detailed analysis of the computational complexity of the algorithms. To simplify the complexity calculation, subtractions are considered additions.

To filter  $L$  points of data with a filter of size  $QM$ , the conventional LMS algorithm requires  $L$  iterations. In each iteration, the adaptation of the model and the update of the coefficients both need  $QM$  additions and  $QM$  multiplications. Therefore, the LMS requires  $2QM$  additions and multiplications in total.

To filter  $L$  points of data with a filter of size  $QM$ , the NLMS algorithm requires  $L$  iterations. In each iteration, the adaptation of the model needs  $QM$  additions and  $QM$  multiplications. The update of the coefficients needs  $2QM$  additions and  $2QM$  multiplications. Therefore the NLMS requires  $3QM$  additions and multiplications in total.

To filter  $L$  points of data with a filter of size  $QM$ , the RLS algorithm requires  $L$  iterations. In each iteration, the adaptation of the model needs  $QM$  additions and  $QM$  multiplications. The update of  $\hat{\mathbf{R}}^{-1}$  requires  $2(QM)^2$  additions and  $3(QM)^2$  multiplications. The update of the coefficients needs  $(QM)^2$  additions and  $QM + (QM)^2$  multiplications. Therefore, the RLS requires  $QM + 2(QM)^2$  additions and  $2QM + 4(QM)^2$  multiplications.

For the WLMS, since it has a block approach, to filter  $L$  points of data with a filter of size  $QM$  it needs  $L/N$  iterations (with  $N$  the nearest higher power of 2 to  $M$ ).  $2Q + 1$  DWT are processed, each composed of  $2N \log_2(2N)$  additions. The adaptation of the model requires  $Q(2N + N_A)$  additions and  $QN_M$  multiplications (cf. Eq. 9 and Eq. 11). The update of the coefficients requires  $QN_A$  additions and  $QN_M$  multiplications. Therefore, the WLMS requires  $(2Q + 1)(2N \log_2(2N)) + 2Q(N + N_A)$  additions and  $2QN_M$  multiplications.

Table II compares the computational resources for LMS, NLMS, RLS and WLMS for different filter sizes  $M$  over  $L = M$  data points.

The WLMS algorithm has significantly lower computational complexity in terms of both additions and multiplications compared to the RLS. For example, when  $Q = 7$  and  $M = 8$ , the WLMS algorithm only requires 2192 additions and 1204 multiplications, while RLS requires 75712 additions and 101696 multiplications. The NLMS algorithm is a direct competitor to the WLMS algorithm in terms of computational complexity. Both algorithms have a linear complexity in terms of the number of filter taps. The WLMS requires more additions than the NLMS since the Walsh Transform is only composed of additions and subtractions. However, the number of multiplications required has been reduced thanks to the block-diagonal structure of  $\mathbf{X}$  and becomes 12% lower than the number of multiplications in the NLMS when  $M$  is a power of two. Indeed, as the transformation can only be applied to vectors having a size equal to a power of 2, when  $M$  is not a power of 2, it is necessary to apply a zero padding to verify this requirement making the computation more expensive. This phenomenon is shown in Fig. 8(a) and Fig. 8(b) displays the evolution of the ratio of multiplications between WLMS and NLMS ( $Mu_{WLMS}/Mu_{NLMS}$ ) for the different parameters. This phenomenon is also found in the ratio of the number of additions of both algorithms.

TABLE II  
COMPARISON OF COMPUTATIONAL REQUIREMENTS FOR LMS, NLMS, RLS AND WLMS WITH DIFFERENT FILTER SIZES.

	LMS		NLMS		RLS		WLMS	
	Add.	Mul.	Add.	Mul.	Add.	Mul.	Add.	Mul.
Q = 3, M = 4	96	96	144	144	756	2448	312	132
Q = 3, M = 6	216	216	324	324	5940	8100	732	387
Q = 5, M = 6	360	360	540	540	16380	22140	1188	645
Q = 5, M = 8	640	640	960	960	38720	5160	1584	860
Q = 7, M = 8	896	896	1344	1344	75712	101696	2192	1204

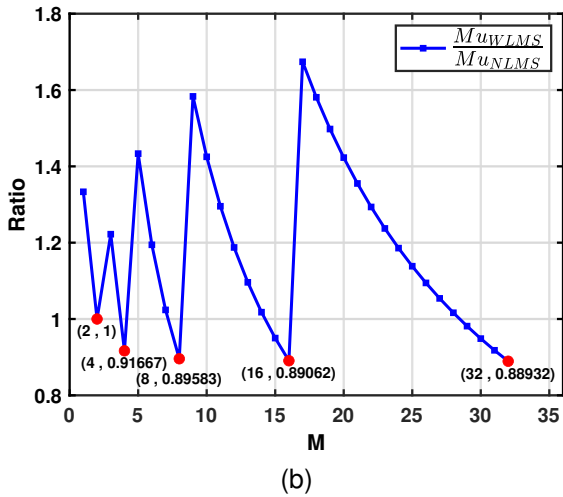
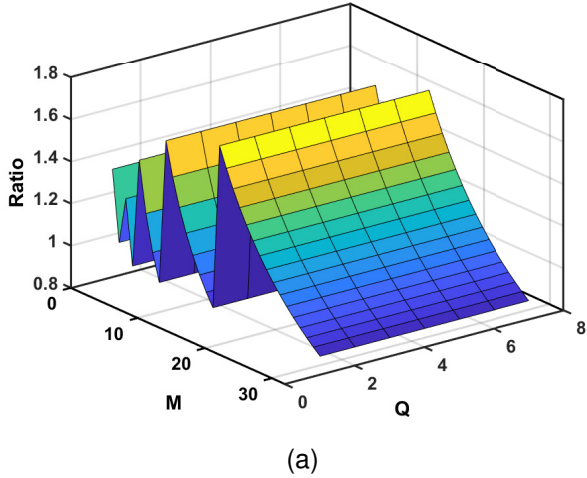


Fig. 8. (a) Ratio of multiplications between WLMS and NLMS for multiple  $Q$  and  $M$  orders. (b) Ratio of multiplications between WLMS and NLMS for multiple  $M$  orders with  $Q$  fixed.

### C. Comparative Performance

An overview of the comparative performances of the four algorithms is presented in Fig. 9. The RLS algorithm has better performances in terms of convergence speed and accuracy compared to other conventional algorithms, but it may suffer from noisy systems and also consumes more energy due to its high complexity of calculus [33]. Therefore, the RLS would be preferable in a DPD application where performance is a priority and noise/energy consumption is not a major issue. On the other hand, in terms of accuracy and convergence speed, the WLMS outperforms the LMS and NLMS algorithms while having a reduced computational complexity compared to the NLMS. Therefore, the WLMS is the preferred choice in a DPD application where convergence speed, accuracy, low-noise sensitivity, and energy consumption are important factors. For instance, in the case where energy consumption is a critical criterion for low-power 5G devices and meeting the minimum ACPR and EVM values is sufficient to comply with ETSI regulations [35], the WLMS algorithm may be a suitable option.

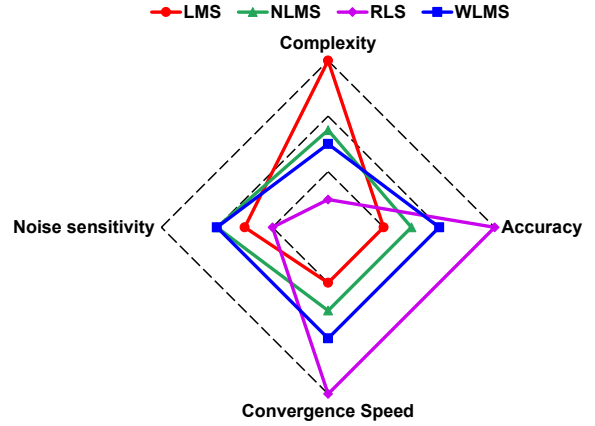


Fig. 9. Overview of the comparative performances of the LMS, NLMS, RLS and WLMS algorithms.

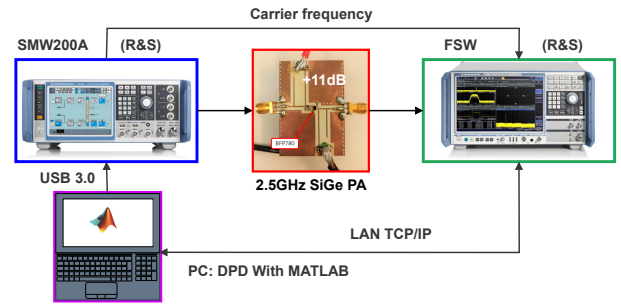


Fig. 10. Synoptic view of the DPD measurement bench.

## IV. EXPERIMENTS AND MEASUREMENT RESULTS

In order to validate the performances of the proposed algorithm, the WLMS DPD was applied on a 2.5GHz SiGe PA.

### A. Experimental Setup

Figure 10 provides detailed information about the measurement setup. The signal processing is performed in MATLAB. The input signal of the PA is generated by the *ROHDE & SCHWARZ (R&S) SMW200A* Vector Signal Generator [36], which is USB 3.0 (VISA) controlled. On the other hand, the *R&S FSW* Signal/Spectrum Analyzer [37] samples the output of the amplifier and is controlled with TCP/IP. The two devices are synchronized to minimize the frequency shift between their local oscillators. The PA that is being linearized is a deep AB-class PA. Its structure is a one-state common emitter with a BFP780 SiGe transistor from Infineon. It operates at 2.5GHz, has a linear gain of 11dB, and its output 1dB-compression point is equal to 14dBm. Figure 11 shows the experimental setup.

### B. Measurement results

The signal used is a 16-QAM OFDM signal with a bandwidth of 50MHz, an average power level of -3dBm, and a PAPR of 7dB. The evaluation of an MP PD with a  $Q = 4$  and  $M = 8$  was performed with LMS, NLMS, RLS and WLMS.



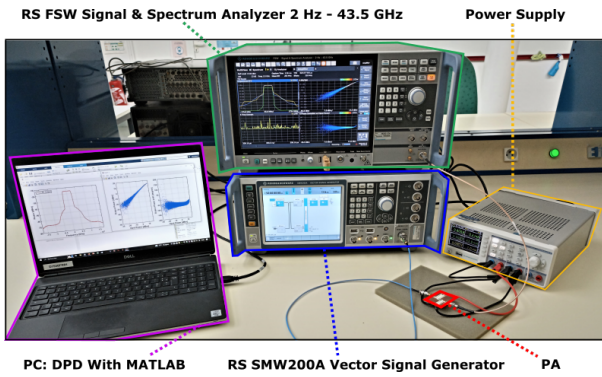


Fig. 11. DPD measurement bench.

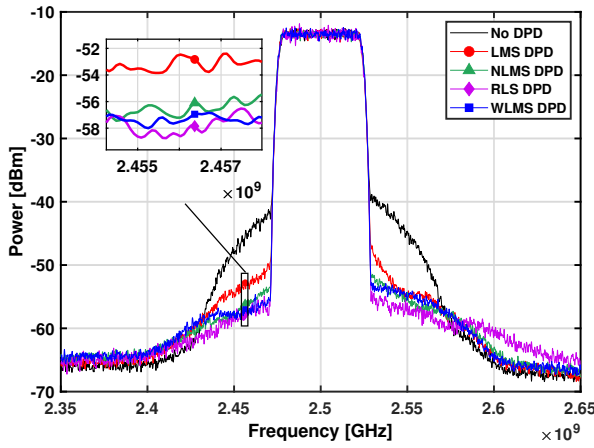


Fig. 12. Measured power spectrum of PA output with and without DPD after 20 iterations of each algorithm.

The evaluation was conducted on a dataset of 5000 samples and after 20 iterations. For each algorithm, the convergence parameters were chosen to achieve the fastest convergence rate without causing the algorithm to diverge:  $\mu = 0.01$  (LMS),  $\mu_n = 0.5$   $\alpha = 3$  (NLMS),  $\delta = 1$   $\lambda = 0.99$  (RLS),  $\mu_W = 0.05$  (WLMS). The results are presented in Fig. 12 which displays the measured PA output spectra with and without DPD using the four different algorithms. Table III compares the ACPR obtained with each algorithm for a 50MHz and 100MHz adjacent channel bandwidth. Figure 13(a) and Fig. 13(b) depict the linearized AM/AM and AM/PM characteristics, respectively, of the measured PA, using the WLMS.

Figure 12 shows that the WLMS can perform a strong ACPR reduction on AB-class PA with 50MHz signal band-

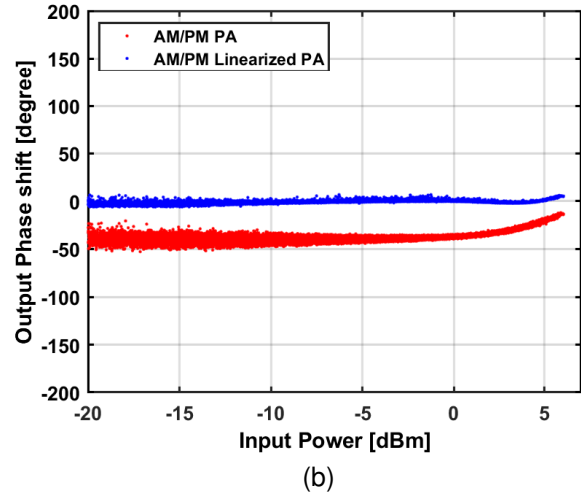
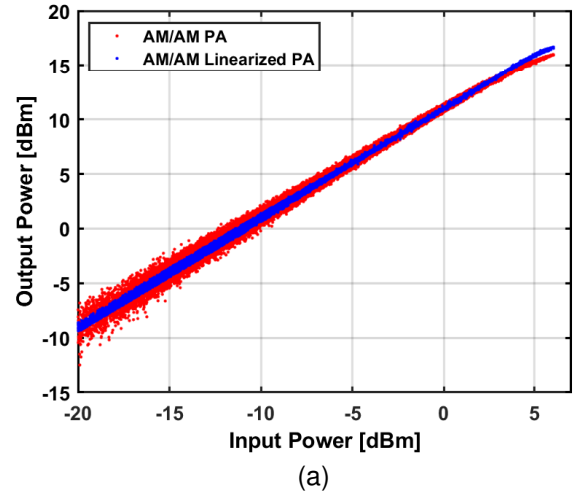


Fig. 13. (a) AM/AM characteristic without and with WLMS DPD. (b) AM/PM characteristic without and with WLMS DPD.

width. Furthermore, the AM/AM and AM/PM in Fig. 13(a) and Fig. 13(b) indicate that the memory effect is also significantly reduced but not completely eliminated, as indicated by the presence of the remaining asymmetric regrowths in the PA output spectrum. Table III indicates that WLMS DPD offers the second-best linearization performance, which is consistent with the simulation results. The only algorithm that outperforms WLMS is RLS DPD, but it comes at the cost of much higher computational complexity.

TABLE III  
ACPR, NMSE AFTER 20 ITERATIONS OF LMS, NLMS, RLS, AND WLMS DPD

Adjacent Bandwidth	16-QAM OFDM			
	ACPR L/U (dBc)		NMSE (dB)	EVM (%)
	50MHz	100MHz		
No DPD	-32.7 / -30.8	-46.8 / -44.2	/	6.99
LMS DPD	-40.5 / -39.5	-47.6 / -46.6	-33.1	5.09
NLMS DPD	-43.8 / -42.1	-49.2 / -48	-37.6	4.54
RLS DPD	-46.1 / -44.5	-53.5 / -48.8	-40.1	3.24
WLMS DPD	-44.3 / -42.2	-48.8 / -46.8	-37.8	4.36

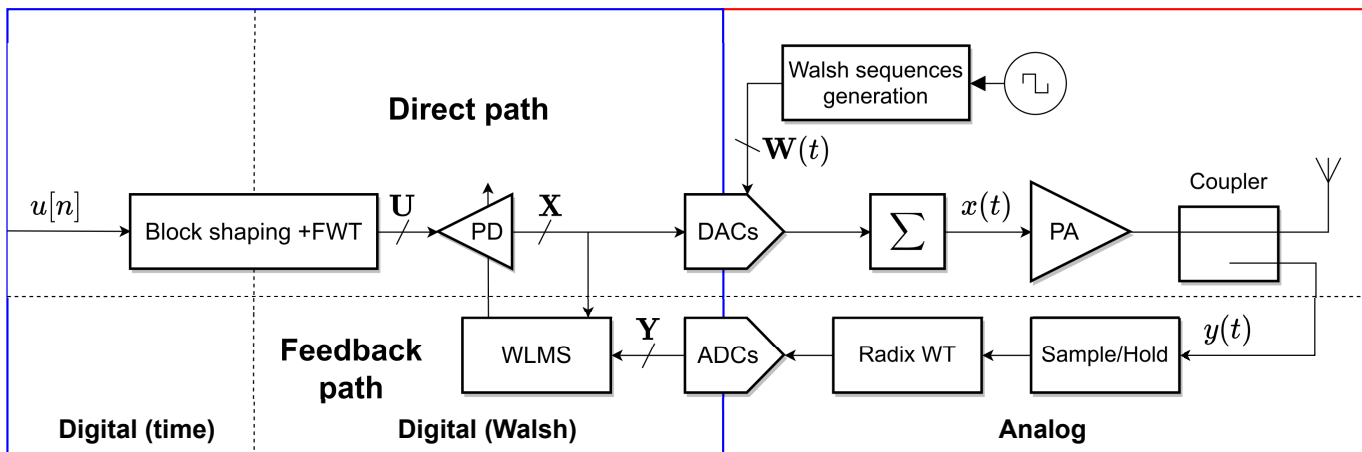


Fig. 14. DPD RF Transmitter based on the WLMS and the Walsh Transform.

### V. WALSH-BASED DPD ARCHITECTURE

Figure 14 illustrates a comprehensive DPD transmitter architecture that utilizes both the Walsh Transform and WLMS.

In the DPD transmitter architecture, the direct path comprises the upper part. To generate the Tx RF signal, the Inverse Walsh Transform (IWT) (also known as Walsh series decomposition) is carried out, as described in Eq. 23.

$$x(t) = \sum_{i=1}^N X_j(i, t) \times W(i, t) \quad (23)$$

Recent research indicates that using Walsh sequences for generating RF signals is a viable option [38], [39]. To produce these sequences, digital clocks generate the Walsh sequences  $W(i, t)$ , which are then weighted by the Walsh coefficients  $X_j$  using DACs. The resulting weighted sequences,  $X_j(i, t)W(i, t)$ , are combined to form the RF signal. The Walsh sequence set  $W(i, t)$  can be obtained from a single clock working at the system's highest frequency,  $f_W$ , using frequency dividers and XOR gates. This method is also wideband, as it generates signals ranging from the lowest frequency in the Walsh sequences ( $f_R = f_W/N$  with  $N$  being the number of sequences in the set) to  $f_W$ . Additionally, the Walsh coefficients change only every  $T_R = 1/f_R$ .

Moving to the feedback path, a coupler captures a copy of the PA's output signal, which is then subjected to analog Walsh Transform using a Radix architecture to compute its Walsh coefficients [40]. These coefficients are sampled and input into the Walsh DPD algorithm. By knowing the input/output Walsh coefficients of the PA, the WLMS computes the Walsh coefficients of the pre-distorted signal. The pre-distorted RF signal is generated by the direct path with the IWT.

The system has been simulated in MATLAB. The data is quantized over 8 bits. The signal used is a 5G NR-FR1 [41]. It is a 64-QAM OFDM with 100MHz bandwidth, a PAPR of 10.4dB, and an average power  $P_{in}$  of 2.6dBm. The PA to be linearized is the NXP Airfast LDMOS Doherty from the previous simulations. The PD model used has a  $Q = 5$  and  $M = 8$ . It was calibrated after 100 iterations of the WLMS ( $\mu_W = 0.08$ ), each using a set of 10000 samples. Figure 15(a)

displays the PA output spectra with and without DPD. Figure 15(b) shows the constellations of the input and output of the PA with and without DPD. Table IV presents a comparison of the ACPR (lower/upper) and EVM levels with and without WLMS DPD.

The proposed architecture achieves an 11.2dB improvement in ACPR and a 50% reduction in EVM for high PAPR large signal bandwidth. These results indicate that the architecture has the potential for direct RF signal generation with highly competitive linearization capabilities, made possible by the use of the Walsh Transform.

### VI. CONCLUSION

This work proposes a Walsh-based LMS algorithm for evaluating a PD model using the input/output Walsh coefficients of the system. Both simulation and measurement results are presented, showing that the novel algorithm can perform a 11dB ACPR reduction with up to 10 times the speed of conventional LMS algorithms at 12% lower computational complexity. As a result, the energy cost of the digital resources is reduced and the energy efficiency of the PA is increased as it operates in the non-linear region. A Walsh-based DPD system was also presented. The Walsh Transform offers a direct generation of the RF signal and a copy of the PA's output signal. Simulation results of the WLMS DPD on an RF PA are also shown, paving the way for a novel RF DPD system based on Walsh theory. Finally, future work will integrate the proposed architecture in 28nm FDSOI CMOS technology from STMicroelectronics.

### APPENDIX A LMS AND RLS ALGORITHMS

This section presents the different algorithms used to compare the proposed method, including LMS, NLMS, and RLS. The block LMS is also described, as the proposed method is based on this approach.

#### A. The LMS Algorithm

The LMS algorithm is a time-domain adaptive filter. Its goal is to find weights of the transversal filter  $c[n]$  to minimize the

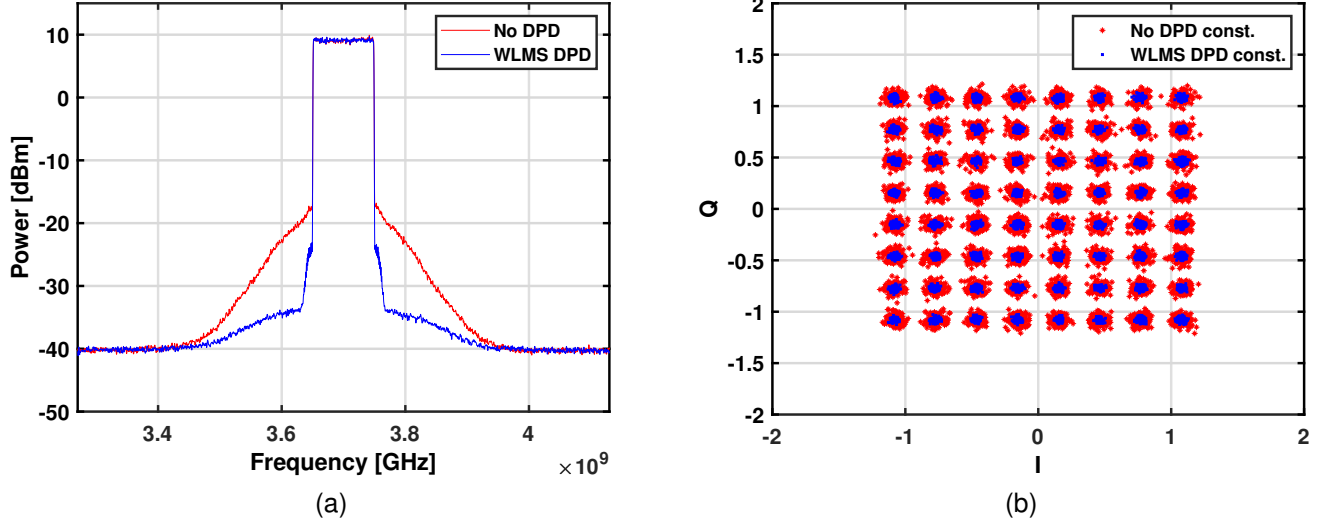


Fig. 15. (a) Power spectrum of PA output with and without DPD after 100 iterations of WLMS algorithm. (b) Output constellation with and without WLMS DPD.

TABLE IV  
SIMULATED ACPR AND EVM LEVELS OF PA OUTPUT SPECTRA.

	5G-NR-FR1 64-QAM				
	ACPR L/U (dBc)		EVM (%)		NMSE (dB)
	100MHz	200MHz	RMS	Peak	
No DPD	-32.1 / -32.3	-40.8 / -41.4	5.5	20.5	/
WLMS DPD	-43.3 / -43.2	-46.8 / -46.8	2.5	10.2	-37.4

error between the output of the filter  $y$  and a reference signal  $d$  [33]. It can be used to filter a noisy set of data or to find a model for a system or inverse model as for DPD. The optimal solution (Wiener [17]) of the filter/model is given by:

$$\mathbf{c}_{\text{opt}} = \mathbf{R}^{-1} \mathbf{p} \quad (24)$$

with  $\mathbf{R} = E[\mathbf{x}_k \mathbf{x}_k^T]$  and  $\mathbf{p} = E[d_k \mathbf{x}_k^T]$  assuming that  $x$  and  $d$  are jointly Wide Sense Stationary (WSS). Since the model has to be evaluated with a finite number of samples, an instantaneous estimation of matrix  $\mathbf{R}$  denoted  $\hat{\mathbf{R}}$  and vector  $\mathbf{p}$ , denoted by  $\hat{\mathbf{p}}$  are used in a steepest-descent algorithm to search for the Wiener solution as follows:

$$\begin{aligned} \mathbf{c}_{k+1} &= \mathbf{c}_k + \mu \hat{\mathbf{V}} \mathbf{c}_k \\ &= \mathbf{c}_k + 2\mu(\hat{\mathbf{p}}_k - \hat{\mathbf{R}}_k \mathbf{c}_k) \end{aligned} \quad (25.1)$$

with:

$$\begin{aligned} \hat{\mathbf{R}}_k &= \mathbf{x}_k \mathbf{x}_k^T \\ \hat{\mathbf{p}}_k &= d_k \mathbf{x}_k^T \end{aligned} \quad (25.2)$$

The result gradient estimate is given by:

$$\hat{\mathbf{V}} \mathbf{c}_k = -2e_k \mathbf{x}_k = \frac{\partial^2 e_k}{\partial \mathbf{c}} \quad (25.3)$$

The resulting gradient-based algorithm is known as the LMS algorithm using the equation updated as:

$$\begin{aligned} e_k &= d_k - y_k = y_k - \mathbf{x}_k^T \mathbf{c}_k \\ \mathbf{c}_{k+1} &= \mathbf{c}_k - 2\mu e_k \mathbf{x}_k \end{aligned} \quad (26.1)$$

where  $\mathbf{x}_k$  corresponds to the state vector of input samples stored in the adaptive filter:

$$\mathbf{x}_k^T = (x_k, x_{k-1}, \dots, x_{k-N+1}) \quad (26.2)$$

$\mathbf{c}_k$  the coefficients of the filter after the  $k$ th iteration:

$$\mathbf{c}_k^T = (c_{k,0}, c_{k,1}, \dots, c_{k,N-1}) \quad (26.3)$$

and  $\mu$  is the convergence factor that has to be chosen in an appropriate range to guarantee the convergence of the algorithm. As the value of  $\mu$  increases, the algorithm will converge more quickly, but this may also result in instability and divergence of the algorithm.

### B. The NLMS Algorithm

The NLMS algorithm [42] is similar to the LMS algorithm, with the only variance lying in the coefficient updating equation. In the NLMS algorithm,  $\mu$  is optimized to achieve faster convergence. It is normalized by  $\mathbf{x}_k^T \mathbf{x}_k$  to reduce the instantaneous error. Also, a factor  $\alpha$  is introduced to avoid large step size when  $\mathbf{x}_k^T \mathbf{x}_k$  becomes too small, leading to a divergence of the algorithm. The updating equation of the NLMS is given by:

$$\mathbf{c}_{k+1} = \mathbf{c}_k + \frac{\mu_n e_k \mathbf{x}_k}{\alpha + \mathbf{x}_k^T \mathbf{x}_k} \quad (27)$$

### C. The RLS Algorithm

The Wiener solution can be evaluated if  $\mathbf{R}^{-1}$  and  $\mathbf{p}$  are known. However, only an estimation of these parameters can

be computed ( $\hat{\mathbf{R}}_k$  and  $\hat{\mathbf{p}}_k$ ) [33]. The RLS uses a recursive estimation of  $\hat{\mathbf{R}}_k^{-1}$  to improve the convergence of the algorithm at the cost of an increase in computational complexity. The updating equation of the RLS is given by:

$$\mathbf{c}_{k+1} = \mathbf{c}_k + e_k \hat{\mathbf{R}}_{k+1}^{-1} \mathbf{x}_k \quad (28.1)$$

with:

$$\begin{aligned} \hat{\mathbf{R}}_1^{-1} &= \delta \mathbf{I} \\ \hat{\mathbf{R}}_{k+1}^{-1} &= \frac{1}{\lambda} \left( \hat{\mathbf{R}}_k^{-1} - \frac{\hat{\mathbf{R}}_k \mathbf{x}_k \mathbf{x}_k^T \hat{\mathbf{R}}_k^{-1}}{\lambda + \mathbf{x}_k^T \hat{\mathbf{R}}_k^{-1} \mathbf{x}_k} \right) \end{aligned} \quad (28.2)$$

The parameter  $\lambda$  is an exponential weighting factor that should be chosen in the range  $]0,1[$ . This parameter is also called the forgetting factor since the information from the distant past has an increasingly negligible effect on the coefficient updating.  $\delta$  can be chosen as the inverse input signal power estimate.

#### D. The Block LMS Algorithm

As described above, the conventional LMS, NLMS, and RLS filters use scalars ( $e_k$ ,  $y_k$ ,  $d_k$ ). However, in a block-based LMS, the output of the filter and the error are vectors. This algorithm uses the overlap-save method [43].

The order  $M$  of the adaptive filter is assumed to be less than or equal to  $N$ . In this method, the adaptive filter is of size  $2N$ . It has  $M$  coefficients followed by  $2N - M$  values set to 0:

$$\mathbf{c}_j^T = (c_{j,0}, c_{j,1}, \dots, c_{j,M-1}, 0, \dots, 0) \quad (29)$$

In a block LMS, the input data set  $x$  is segmented into  $2N$  point vectors that overlap on the last  $N$  points every  $j$  iteration:

$$x_j[n] = x[(j-1)N + n] \quad (30)$$

In a time overlap-save algorithm, the output vector  $\mathbf{y}_j$  is obtained by circular convolution of the input vector and the filter coefficients. It is defined in Eq. 31.

$$\begin{pmatrix} y_j[1] \\ y_j[2] \\ \vdots \\ y_j[N] \\ y_j[N+1] \\ \vdots \\ y_j[2N-1] \\ y_j[2N] \end{pmatrix} = \begin{pmatrix} x_j[1] & x_j[2N] & \dots & x_j[2] \\ x_j[2] & x_j[1] & \dots & x_j[3] \\ \vdots & \vdots & \ddots & \vdots \\ x_j[N] & x_j[N-1] & \dots & x_j[N+1] \\ x_j[N+1] & x_j[N] & \dots & x_j[N+2] \\ \vdots & \vdots & \ddots & \vdots \\ x_j[2N-1] & x_j[2N-2] & \dots & x_j[2N] \\ x_j[2N] & x_j[2N-1] & \dots & x_j[1] \end{pmatrix} \begin{pmatrix} c_{j,0} \\ c_{j,1} \\ \vdots \\ c_{j,M-1} \\ 0 \\ \vdots \\ 0 \\ 0 \end{pmatrix} \quad (31)$$

Which can be reduced as:

$$\mathbf{y}_j = \boldsymbol{\chi}_j \times \mathbf{c}_j \quad (32)$$

With  $\boldsymbol{\chi}_j$  the circulant matrix of the input data set  $\mathbf{x}_j$ .

The error vector  $\mathbf{e}_j$  is defined by:

$$\mathbf{e}_j = \mathbf{f}(\mathbf{d}_j - \mathbf{y}_j) \quad (33)$$

With  $\mathbf{f}$  a diagonal window matrix with its first  $N$  elements equal to 0 and its last  $N$  elements equal to 1. The update of the filter coefficients  $\mathbf{c}_j$  is given by:

$$\mathbf{c}_{j+1} = \mathbf{c}_j + 2\mu \boldsymbol{\chi}_j^T \mathbf{e}_j \quad (34)$$

Because the last  $N$  points of  $\mathbf{c}_j$  are equal to 0, the result of the linear convolution is the last  $N$  point of  $\mathbf{y}_j$  hence the use of the window matrix  $\mathbf{f}$ . This algorithm offers enhanced convergence speed compared to the conventional scalar LMS algorithm at the expense of hardware resources [26].

## APPENDIX B ADDITIONAL SIMULATIONS

This section presents additional simulations complementing the WLMS study presented in Section III. The simulations explore various models and consider different levels of SNR and quantization bits.

### A. ACPR simulations for different MP models

The ACPR evolution for different  $Q$  and  $M$  combinations of the predistorter during 20 iterations of the WLMS algorithm is shown in Fig. 16(a) and Fig. 16(b) for two inputs (50MHz 16-QAM RRC and 64-QAM OFDM).

The simulations demonstrate that the behavior of the algorithm remains stable in the given scenarios, regardless of the number of coefficients utilized for the PD.

### B. SNR and Quantization simulations

The ACPR evolution during 20 iterations of the WLMS algorithm with different bits of quantization and SNR (Gaussian noise) is shown in Fig. 17(a) and 17(b) for the second input (50MHz 64-QAM OFDM).

To achieve a reduction in ACPR, the simulations revealed that the output signal should be quantized to at least 6 bits, and the SNR must be 30dB. Also, the WLMS demonstrated stable performance during the linearization phase.

## ACKNOWLEDGMENT

The author would like to thank *ROHDE & SCHWARZ* for lending us a *R&S SMW200A* and a *R&S FSW* instruments used for the measurements. The project HERMES has received funding from the European Union's Horizon 2020 research and innovation program under grant agreement No 964246.

## REFERENCES

- [1] A. Younes, B. Boubakar, S. Bachir, and C. Duvanaud, "Interleaving technique implementation to reduce papr of ofdm signal in presence of nonlinear amplification with memory effects," *Journal of Telecommunications and Information Technology*, vol. 3, 09 2018.
- [2] M. B. Mabrouk, G. Ferré, E. Grivel, and N. Deltimple, "Interacting multiple model based detector to compensate power amplifier distortions in cognitive radio," *IEEE Transactions on Communications*, vol. 63, no. 5, pp. 1580–1593, 2015.
- [3] E. Ngoya and S. Mons, "Progress for behavioral challenges: A summary of time-domain behavioral modeling of rf and microwave subsystems," *IEEE Microwave Magazine*, vol. 15, no. 6, pp. 91–105, 2014.
- [4] J. Vuolevi, T. Rahkonen, and J. Manninen, "Measurement technique for characterizing memory effects in rf power amplifiers," in *RAWCON 2000. 2000 IEEE Radio and Wireless Conference (Cat. No.00EX404)*, 2000, pp. 195–198.

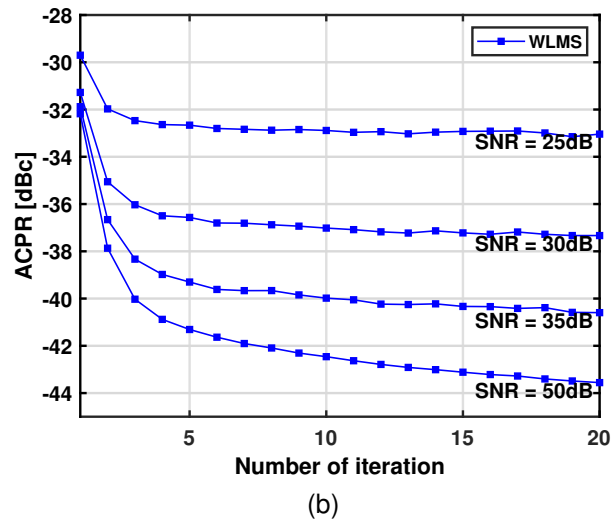
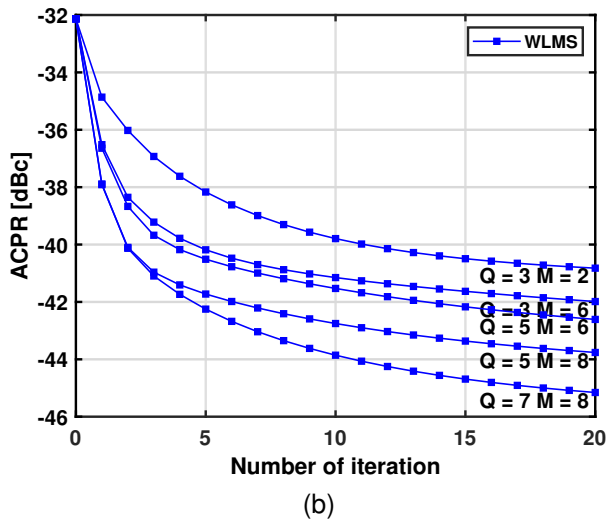
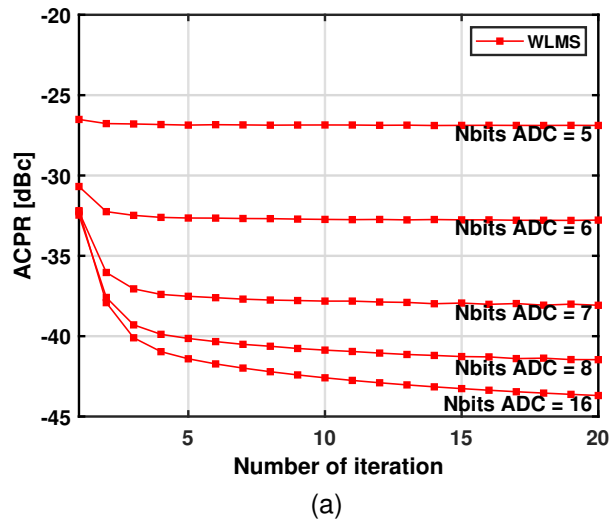
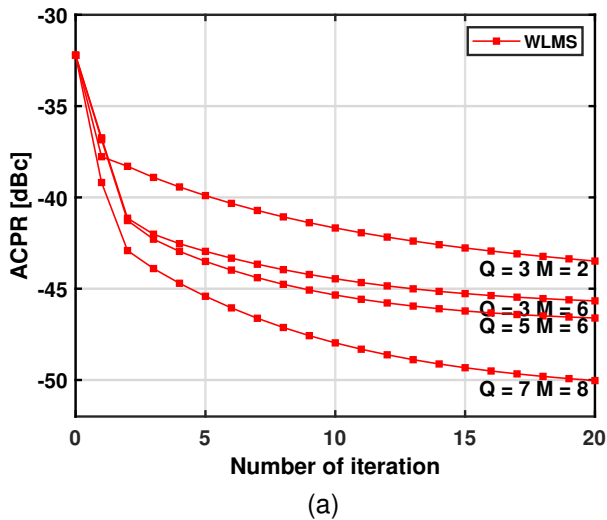


Fig. 16. (a) ACPR evolution over 20 iterations of WLMS for multiple models with a 16-QAM RRC input. (b) ACPR evolution over 20 iterations of WLMS for multiple models with a 64-QAM OFDM input.

Fig. 17. (a) ACPR evolution over 20 iterations of WLMS for multiple bits of quantization. (b) ACPR evolution over 20 iterations of WLMS for multiple SNR.

[5] I. Fijalkow, J.-L. Gautier, S. Traverso, M. Ariaudo, and A. Chaker, "Adaptive compensation of high frequency memory effects of a realistic amplifier in an ofdm system," in *2006 2nd International Conference on Information Communication Technologies*, vol. 2, 2006, pp. 2629–2633.

[6] H. Ku and J. Kenney, "Behavioral modeling of nonlinear rf power amplifiers considering memory effects," *IEEE Transactions on Microwave Theory and Techniques*, vol. 51, no. 12, pp. 2495–2504, 2003.

[7] J. Vuolevi, T. Rahkonen, and J. Manninen, "Measurement technique for characterizing memory effects in rf power amplifiers," *IEEE Transactions on Microwave Theory and Techniques*, vol. 49, no. 8, pp. 1383–1389, 2001.

[8] S. Cripps, *Advanced Techniques in RF Power Amplifier Design*, 2002, ch. 5, pp. 187–190.

[9] A. Katz, J. Wood, and D. Chokola, "The evolution of pa linearization: From classic feedforward and feedback through analog and digital predistortion," *IEEE Microwave Magazine*, vol. 17, no. 2, pp. 32–40, 2016.

[10] S. Cripps, *RF Power Amplifiers for Wireless Communications, Second Edition*, 2006, ch. 14, pp. 397–425.

[11] X. Feng, "Efficient baseband digital predistortion techniques for linearizing power amplifier by taking into account nonlinear memory effect," Theses, UNIVERSITE DE NANTES, Sep. 2015. [Online]. Available: <https://hal.archives-ouvertes.fr/tel-01206266>

[12] V. Volterra, *Theory of functionals and of integral and integro-differential*

*equations*. London: Blackie & Son, 1930.

[13] S. Narayanan, "Transistor distortion analysis using volterra series representation," *The Bell System Technical Journal*, vol. 46, no. 5, pp. 991–1024, 1967.

[14] J. Kim and K. Konstantinou, "Konstantinou, k.: Digital predistortion of wideband signals based on power amplifier model with memory. iee electronics letters 37, 1417-1418," *Electronics Letters*, vol. 37, pp. 1417 – 1418, 12 2001.

[15] L. Ding, G. Zhou, D. Morgan, Z. Ma, J. Kenney, J. Kim, and C. Giardina, "A robust digital baseband predistorter constructed using memory polynomials," *IEEE Transactions on Communications*, vol. 52, no. 1, pp. 159–165, 2004.

[16] D. Morgan, Z. Ma, J. Kim, M. Zierdt, and J. Pastalan, "A generalized memory polynomial model for digital predistortion of rf power amplifiers," *IEEE Transactions on Signal Processing*, vol. 54, no. 10, pp. 3852–3860, 2006.

[17] P. Gilabert, G. Montoro, and E. Albertí, "On the wiener and hammerstein models for power amplifier predistortion," vol. 2, 01 2006, p. 4 pp.

[18] R. Hongyo, Y. Egashira, T. M. Hone, and K. Yamaguchi, "Deep neural network-based digital predistorter for doherthy power amplifiers," *IEEE Microwave and Wireless Components Letters*, vol. 29, no. 2, pp. 146–148, 2019.

[19] O. Hammi, A. E. Abdelrahman, and A. Zerguine, "Multi-basis weighted memory polynomial for rf power amplifiers behavioral modeling," in

2016 *IEEE MTT-S International Wireless Symposium (IWS)*, 2016, pp. 1–4.

[20] R. N. Braithwaite, “Pruning strategies for a volterra series model used in digital predistortion (dpd) of rf power amplifiers,” in *2017 IEEE Topical Conference on RF/Microwave Power Amplifiers for Radio and Wireless Applications (PAWR)*, 2017, pp. 4–7.

[21] T. H. C. Bouazza, S. Bachir, and C. Duvanaud, “A new block-oriented model for reduced complexity digital predistortion in wideband applications,” in *2021 IEEE International Symposium on Circuits and Systems (ISCAS)*, 2021, pp. 1–5.

[22] Q. You, L. Gu, and J. Liu, “Improved variable step-size lms for digital predistortion in wideband power amplifiers,” in *2018 8th International Conference on Electronics Information and Emergency Communication (ICEIEC)*, 2018, pp. 116–119.

[23] S. Yao, H. Qian, K. Kang, and M. Shen, “A recursive least squares algorithm with reduced complexity for digital predistortion linearization,” in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2013, pp. 4736–4739.

[24] R. N. Braithwaite, “Fixed point considerations for digital predistortion of a rf power amplifier using recursive least square (rls) estimation,” in *2019 IEEE Topical Conference on RF/Microwave Power Amplifiers for Radio and Wireless Applications (PAWR)*, 2019, pp. 1–3.

[25] S. Im and E. Powers, “A block lms algorithm for third-order frequency-domain volterra filters,” *IEEE Signal Processing Letters*, vol. 4, no. 3, pp. 75–78, 1997.

[26] G. Clark, S. Parker, and S. Mitra, “A unified approach to time- and frequency-domain realization of fir adaptive digital filters,” *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 31, no. 5, pp. 1073–1083, 1983.

[27] D. Mansour and A. Gray, “Unconstrained frequency-domain adaptive filter,” *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 30, no. 5, pp. 726–734, 1982.

[28] J. L. Walsh, “A closed set of normal orthogonal functions,” *American Journal of Mathematics*, vol. 45, no. 1, pp. 5–24, 1923.

[29] G. Robinson, “Logical convolution and discrete walsh and fourier power spectra,” *IEEE Transactions on Audio and Electroacoustics*, vol. 20, no. 4, pp. 271–280, 1972.

[30] C. Zarowski and M. Yunik, “Spectral filtering using the fast walsh transform,” *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 33, no. 5, pp. 1246–1252, 1985.

[31] J. Johnson and M. Puschel, “In search of the optimal walsh-hadamard transform,” in *2000 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (Cat. No.00CH37100)*, vol. 6, 2000, pp. 3347–3350 vol.6.

[32] J. Bermudez and N. Bershad, “Nonlinear quantization effects on the lms algorithm-analytical models for the mse transient and convergence behavior,” in *Proceedings of 1994 28th Asilomar Conference on Signals, Systems and Computers*, vol. 1, 1994, pp. 627–631 vol.1.

[33] P. S. R. Diniz, in *Adaptive Filtering: Algorithms and Practical Implementation*, P. S. R. Diniz, Ed., vol. 4, no. 4. Springer, 2013, pp. 79–99, 209–230, 597–598.

[34] Mathworks, “Power Amplifier Characterization,” <https://fr.mathworks.com/help/comm/ug/power-amplifier-characterization.html>.

[35] *5G Radio Access Network*, 2020. [Online]. Available: [https://www.etsi.org/deliver/etsi\\_en/303600\\_303699/303645/01\\_01\\_01\\_60/en\\_303645v010101p.pdf](https://www.etsi.org/deliver/etsi_en/303600_303699/303645/01_01_01_60/en_303645v010101p.pdf)

[36] *SMW200A, ROHDE & SCHARWZ*. [Online]. Available: [https://www.rohde-schwarz.com/uk/products/test-and-measurement/vector-signal-generators/rs-smw200a-vector-signal-generator\\_63493-38656.html?change\\_c=true](https://www.rohde-schwarz.com/uk/products/test-and-measurement/vector-signal-generators/rs-smw200a-vector-signal-generator_63493-38656.html?change_c=true)

[37] *FSW, ROHDE & SCHARWZ*. [Online]. Available: [https://www.rohde-schwarz.com/uk/products/test-and-measurement/benchtop-analyzers/rs-fsw-signal-and-spectrum-analyzer\\_63493-11793.html](https://www.rohde-schwarz.com/uk/products/test-and-measurement/benchtop-analyzers/rs-fsw-signal-and-spectrum-analyzer_63493-11793.html)

[38] N. Bouassida, F. Rivet, Y. Deval, D. Duperray, and A. Cathelin, “A concurrent transmitter in cmos 28nm fdsoi technology based on walsh sequences generator,” in *2016 14th IEEE International New Circuits and Systems Conference (NEWCAS)*, 2016, pp. 1–4.

[39] R. Quéheille, F. Rivet, N. Deltimple, Y. Deval, and E. Kerhervé, “Demonstration of a walsh-based arbitrary waveform generator using components off-the-shelf,” in *2021 28th IEEE International Conference on Electronics, Circuits, and Systems (ICECS)*, 2021, pp. 1–4.

[40] O. U. Khan and D. D. Wentzloff, “1.2 gs/s hadamard transform front-end for compressive sensing in 65nm cmos,” in *2013 IEEE Radio and Wireless Symposium*, 2013, pp. 181–183.

[41] Mathworks, “5G NR-TM and FRC Waveform Generation,” <https://fr.mathworks.com/help/5g/ug/5g-nr-tm-waveform-generation.html>.

[42] D. Slock, “On the convergence behavior of the lms and the normalized lms algorithms,” *IEEE Transactions on Signal Processing*, vol. 41, no. 9, pp. 2811–2825, 1993.

[43] J. Shynk, “Frequency-domain and multirate adaptive filtering,” *IEEE Signal Processing Magazine*, vol. 9, no. 1, pp. 14–37, 1992.



**Maxandre Fellmann** Maxandre Fellmann received a Dipl.-Ing degree in Electronics from ENSEIRB-MATMECA electronics graduate School, Bordeaux, France, and a Master degree from the University of Bordeaux, France in 2021. He is currently pursuing a Ph.D. degree in electrical engineering at the IMS Laboratory, the microelectronics laboratory of the University of Bordeaux. His research interests include digital predistortion, signal processing, and circuit and system-level design of wireless systems. He is in charge of the SSC chapter for the IEEE Bordeaux student branch as the chapter’s chair.



**François Rivet** Dr. Francois Rivet received the Master and the PhD degree respectively in 2005 and 2009 from the University of Bordeaux. Since June 2010, he is tenured as Associate Professor at Bordeaux Institute of Technology (Bordeaux INP). His research is focused on the design of RFICs in the IMS Laboratory, the microelectronics laboratory of the University of Bordeaux. In 2014, he founded the “Circuits and Systems” research team. Dr. Rivet has publications in top-ranked journals, international conferences, national conferences and holds 17 patents. He received the Best Paper Award at Software Defined Radio Forum in 2008 at Washington DC, USA, Journées Nationales Micro-ondes in 2015 and 2017, LASCAS in 2021 at Arequipa, Peru and AVIC in 2022 at Hiroshima, Japan. He is a member of several Technical Program Committees (RFIC, ESSCIRC, ...) and steering committees (RFIC, ICECS, LASCAS).



**Nathalie Deltimple** Prof. Nathalie Deltimple received the Ph.D. degree in electrical engineering from the University of Bordeaux, Talence, France, in 2005. In 2006, she joined Bordeaux INP/ENSEIRB-MATMECA and the IMS Laboratory, Talence, France, where she is currently Professor at Bordeaux INP/ENSEIRB-MATMECA at the Electronics Department. Her main areas of research are the design of RF power amplifiers in CMOS and BiCMOS technologies for TX and wireless communications, the development of integrated efficiency enhancement, and linearization techniques. Since several years, Pr. Deltimple is involved in the Technical Program Committees or steering committee of IEEE international conferences (CAS-S NEWCAS and ICECS, MTT-S IMS, IWS, RWS) and EuMA EuMW conference. She was the General Chair of ICECS2018, is the co-General Chair of ISICAS2022 and was the TPC chair of IWS in 2016 and EuMW in 2019. She serves as a reviewer for various conferences and journals (IEEE JSSC, TCAS-II, MTT, MWCL, and AICSP). She is a member of the IEEE CAS Society, IEEE MTT Society, IEEE JSSC Society, IEEE Woman In Engineering and she is vice-chair of IEEE France section CAS chapter.



**Eric Kerhervé** Prof. Eric Kerhervé received the Ph.D. degree in EE from Bordeaux University, France (1994). He is Professor in Microelectronics and Microwave circuits in Polytechnic Institute of Bordeaux. He is director of the STMicroelectronics/IMS joint Lab (2015-...) and deputy director of IMS (2016-...). His main research activities focus on the design of RF and mmW circuits in Silicon and GaN technologies. He was involved in 7 European projects and 6 French projects to develop silicon RF/mmW power amplifiers. He has authored or co-

authored more than 200 technical papers in this field, and was awarded 24 patents. He has organized 8 RFIC/MTT workshops on advanced silicon technologies for radiofrequency and millimeter-wave applications. He was General Chair of IEEE ICECS'2006, IEEE NEWCAS'2011, EuMIC'2015 and JNM'2015. He is member of the Executive Committee of SiRF and member of the MTT-7 committee. He was 2-years associate editor of IEEE Transactions on Circuits and Systems II.



**Hervé Lapuyade** Dr. Hervé LAPUYADE (M'02) received the M.S. degree in electronics from ENSEIRB, the school of Electrical Engineers of Bordeaux, France, in 1991, and the Ph.D. degree from the University of Bordeaux, France, in 1996. He joined the University of Bordeaux in 1996 as an Assistant Professor, working in the domains of LASER testing and radiation hardened electronics at IXL, the Microelectronics laboratory of the University of Bordeaux. In 2000 he joined the IC Design Team of IXL, modifying his research interests toward the

domains of RFIC design and Design for Test (DfT). In 2007, IXL merged with two other research laboratories in the EE field to create IMS, the laboratory of Integration, from Materials to Systems of the University of Bordeaux. In this new research laboratory, Dr. Lapuyade is currently an Associate Professor in charge of high reliability electronics and Design for Availability (DfA) activities in the IC Design Team, IC Design Group.



**Yann Deval** Prof. Yann DEVAL (M'96, SM'07) joined the University of Bordeaux in the Southwest of France in 1993, as an Assistant Professor, focusing on the design of analog ICs, RFICs, high-speed mixed-signal ICs, and high-reliability electronics. He pursues his researches within the IMS, the laboratory of Integration, from Material to System. In 1999 Dr. Deval became an Associate Professor and, in 2004, a Full Professor. From 2006 to 2010 he was the head of the IC design Group, and from 2011 to 2016 the head of the Devices, Circuits and Systems (DCS)

Department. Since 2016, Dr. Deval is the director of IMS. This public research laboratory is composed of 150 Ph.D. students and 150 faculties, with the support of roughly 100 technicians, engineers and admin assistants.