Joint Polynomial and Look-Up-Table Predistortion Power Amplifier Linearization

Hsin-Hung Chen, Chih-Hung Lin, Po-Chiun Huang, and Jiunn-Tsair Chen

Abstract—Digital predistortion at baseband is an efficient and low-cost method for the linearization of a power amplifier (PA) in a wireless system employing a nonconstant-envelop modulation scheme, so as to reduce the adjacent channel interference. The polynomial and the look-up table (LUT) predistortion schemes are two commonly used approaches. However, in each of the two approaches, to reach a satisfactory adjacent channel power ratio (ACPR) in the PA output signal, people usually end up with a complex system having the involved algorithms converge rather slowly. In this brief, we propose a low-complexity joint-polynomial-and-LUT predistortion PA linearizer, where the two mutually dependent predistortion schemes can skillfully help each other. Simulation results show that the proposed joint linearizer can reduce the algorithm convergence time while achieving an excellent ACPR.

Index Terms—Linearization, look-up table (LUT), power amplifier (PA), polynomial.

I. INTRODUCTION

SCARCITY of radio resources motivates the current wireless communication systems to adopt (higher order) linear modulation schemes, such as $M$ phase shift key (PSK) or $M$ quadrature amplitude modulation (QAM), so as to achieve better power-spectral efficiency. However, the effect of the pulse-shaping filter in a linear modulator leads to appreciable amplitude fluctuation in the modulated signals, which may, in turn, cause the AM/AM nonlinear distortion and the AM/phase modulation (AM/PM) nonlinear distortion in the power amplifier (PA) of a transmitter. These nonlinear distortions inevitably result in adjacent channel interference (ACI). The nonlinearity of the PA also degrades the symbol error rate (SER) performance at the receiver. Therefore, the PA linearity plays an important role in the design of a wireless system.

The predistortion linearizer can be generally categorized into two groups: polynomial methods [2]–[4] and look-up table (LUT) methods [5]–[8]. For the polynomial methods, the inverse characteristic function of the PA is described by a polynomial. Authors of [3] introduce an adaptive polynomial predistortion technique by representing the involved signals in the polar coordinates, which can assure a fast rate of algorithm convergence. A neuro-fuzzy linearizer reported in [4], which can be seen as an accuracy-improved version of the polynomial linearizer, makes use of some techniques developed in the field of the neuro-network to optimize a continuous predistortion function. For the LUT methods, the inverse characteristic function of the PA is described by the memory contents of a LUT. In [5], Cavers introduces a “complex gain predistorter,” which employs a one-dimensional LUT with the amplitude of the input signal being the only addressing index of the LUT. However, with the LUT approach, the quantization effect is an inevitable problem. To mitigate this effect, some papers are devoted to nonuniform LUT linearizer [6]–[8], which adopts a nonuniformly indexed LUT to shift the linearization effort to those nonlinear regions of the PA characteristics.

The ability in nonlinear correction of the polynomial and that of the LUT predistorters are very different. The polynomial predistorter can easily trace the deep slopes of the inverse characteristic function of a PA, while the LUT predistorter enjoys the ability to trace the fast zigzag variations of the same function. However, the polynomial predistorter cannot handle the fast zigzag variations well due to its limit on the order of the polynomial, while the LUT predistorter cannot handle the deep slope well due to its limit on the size of the LUT. In other words, the polynomial predistorter suffers curve-fitting distortions and the LUT predistorter suffers quantization distortions, i.e., distortions at those parts of a PA output signal where its original input signal amplitudes are between the values specified by the LUT addressing index. In an attempt to prevent these distortions and thereby improve the accuracy of the linearizers, the polynomial and the LUT predistorter will each have to face different problems: For the polynomial predistorter, the digital signal processing (DSP) complexity of evaluating a polynomial increases dramatically if the predistorter is extended to use fifth- and seventh-order terms [5]. For the LUT predistorter, the adaptation and the initialization time of the linearization procedure becomes exceedingly long if we have to employ a large-size LUT [3].

Observing that the characteristics of the left-over nonlinear distortions after the application of the polynomial predistortion and after the application of the LUT predistortion are quite different, in this brief, we propose a joint polynomial and LUT predistortion linearizer, which is capable of: 1) further cancelling the nonlinear distortions remaining from either only the polynomial or only the LUT predistortion linearizer and 2) significantly reducing the required initialization and adaptation time of the linearization procedure to an acceptable range while maintaining a low DSP computational complexity. Hence, the proposed joint linearizer picks up the advantages of both the polynomial and the LUT linearizers, and it avoids some inevitable problems in using either only the polynomial linearizer or only...
the LUT linearizer. Note that [4], [6]–[8] try to push the performance limit of the conventional predistorters, in contrary to the proposed joint linearizer, which not only relaxes the performance demand on both the polynomial linearizer itself and the LUT linearizer itself, but also jointly coordinates both linearizers to compensate each other’s nonlinear distortions. Nevertheless, improvement of the conventional predistorters in the joint linearizer with techniques reported in [4], [6]–[8] would further improve the overall performance of the joint linearizer.

II. SYSTEM MODEL

The top circuit route of Fig. 1 shows the basic function blocks of a transmitter with a predistorted PA if we consider the shaded rectangle a black-boxed PA predistorter. The input to the predistorter is a modulated complex baseband signal $v_m$. Note that, in this brief, the complex arguments and functions are denoted by boldface letters. The baseband equivalent representation of the transmit waveform after the PA is defined as $v_d$.

The baseband predistortion block, the shaded rectangle in Fig. 1, is located before the D/A converter and controlled by a DSP block employing an adaptive PA linearization algorithm. The characteristic function of the predistorter is denoted as $F(\cdot)$ and that of the PA is denoted as $G(\cdot)$.

Therefore, the predistorter output can be written as

$$v_d = v_m \cdot F(r_m)$$ (1)

where $r_m = |v_m|$ is the amplitude of the input signal $v_m$. The output of the PA with the predistorted input can then be written as

$$v_a = v_d \cdot G(r_d)$$ (2)

where $r_d = |v_d|$ is the amplitude of the predistorted signal $v_d$.

From (1) and (2), we can rewrite $v_a$ as

$$v_a = v_m \cdot F(r_m) \cdot G(|v_m \cdot F(r_m)|).$$ (3)

In an ideal scenario, we have

$$v_a = \alpha \cdot v_m$$ (4)

where $\alpha$ is the overall complex signal gain.

III. PREDISTORTION LINEARIZERS

A. Polynomial Predistorter

The system block diagram of an Order-3 polynomial predistortion linearizer can be illustrated by Fig. 2(a). If we consider an Order-$(2l+1)$ polynomial linearizer, the polynomial-predistorted signal $v_d$ at time $k$ can be expressed as

$$v_d(k) = v_m(k) \cdot F_P(r_m(k))$$ (5)

where

$$F_P = \rho_1(k) + \rho_3(k) \cdot r_m^2(k) + \cdots + \rho_{2l+1}(k) \cdot r_m^{2l}(k)$$ (6)

denotes the predistortion polynomial with $\{\rho_{2l+1}(k)\}$, respectively, being the $(2l+1)$th-order polynomial coefficients at time $k$. Also note that, the polynomial predistortion block, $F_P(\cdot)$, is a function of only the input signal amplitude $r_m(k)$.

The least-mean-square (LMS) [9] algorithm is usually adopted for the polynomial predistorter to adjust the $i$th-order polynomial coefficients $\rho_i(k)$

$$\rho_i(k+1) = \rho_i(k) + \mu_{PC} \cdot v_m^i(k) \cdot r_m^{i-1}(k) \cdot (v_m(k) - v_a(k))$$ (7)

$$v_a(k) = v_d(k) \cdot G(r_d(k))$$ (8)

where $\mu_{PC}$ are the step-sizes used to control the tradeoff between the algorithm convergence speed and the residual error.

The main concept in the polynomial predistortion is curve-fitting. The conventional way is to measure the PA characteristics first, and predistort the PA in an analog way without adaptation. A more flexible way is to employ adaptive digital predistortion, where the PA characteristics need not to be known. Shown in Fig. 3 are several power spectrum density curves of a QPSK-modulated signal with a 2-dB output power backoff (OPBO) transmitted through a TWT PA with gain and phase nonlinearity [10]. Without linearization, the signal spectrum will spread into adjacent channels with a large amount of power. From Fig. 3,
at least an Order-5 polynomial predistorter is needed to obtain a near ideal power spectrum density. Therefore, tradeoff between the accuracy of nonlinear compensation and the DSP complexity becomes a serious drawback of the polynomial predistorter.

B. LUT Predistorter

Similarly, the system block diagram of the LUT predistorter can also be shown in Fig. 2(b) where the $Q$ block denotes a quantizer. The access of the LUT entries depends on the quantized value of $r_m(k)$. In other words, the predistortion complex gain, $F_L(Q(r_m(k)), k)$, provided by the LUT predistorter at time $k$, depends on the quantized amplitude $Q(r_m(k))$ of the input signal $v_m(k)$. Therefore, the predistorted signal after the LUT predistorter at time $k$ can be written as

$$v_d(k) = v_m(k) \cdot F_L(Q(r_m(k)), k)$$  \hspace{1cm} (9)

where $F_L(Q(r_m(k)), k)$ denotes the LUT entry indexed by $r_m(k)$ at time $k$. The LMS iterative update procedure of the complex gain, $F_L(Q(r_m(k)), k)$, provided by the LUT predistorter can be described as

$$F_L(Q(r_m(k + l_k)), k + l_k) = F_L(Q(r_m(k)), k) + \mu_L v_m^*(k)(v_m(k) - v_a(k))$$  \hspace{1cm} (10)

$$v_a(k) = v_d(k) \cdot G(r_a(k))$$  \hspace{1cm} (11)

where $\mu_L$ is the LMS step-size, and $l_k$ is the smallest integer that the amplitudes of $v_m(k)$ and $v_m(k + l_k)$ have the same quantized values, i.e., $Q(v_m(k))$ and $Q(v_m(k + l_k))$ refer to the same entry in the LUT. Since each iteration only updates one entry of the entire LUT, it may take a long time before we update the same entry of the LUT again. Therefore, we expect the LUT predistorter to convergence rather slowly when the size of the LUT is large.

Since: 1) each LUT entry specifies only one point-value of the inverse characteristic function of the PA and 2) the LUT entries are read out according to the quantized amplitude of the input signal, the process of addressing the LUT will inevitably introduce quantization distortion in the LUT predistorter. Shown in Fig. 4 is one example transfer function of a LUT predistorter. Only when the amplitude of the LUT input signal is exactly equal to its quantized value, the LUT can provide a correct value for nonlinear compensation without any quantization distortion. This quantization distortion will become the background noise and limit the performance of the LUT predistorter.

C. Proposed Predistorter

To mitigate the drawbacks emerged from the conventional predistortion linearizers, we next propose a novel joint predistortion linearizer, where both the polynomial and the LUT predistorters are employed for PA linearization. In other words, we intend to improve the PA linearity through baseband digital predistortion, but not suffering either high computation complexity introduced by the polynomial predistorter or the low algorithm convergence speed introduced by the LUT predistorter. The idea of employing both the polynomial and the LUT predistorters is that if these two predistorters can somehow help each other to further improve the PA linearity, we may not need to push predistorter’s limits by either increasing the polynomial’s order in the polynomial predistorter or enlarging the LUT size in the LUT predistorter, and thus avoid the respective problems in either predistorter. Fortunately, this is exactly what happened to the two predistorters since the polynomial predistorter can help reduce the slopes of the PA characteristic functions, which in turn helps the LUT predistorter to reduce the quantization distortion. The system block diagram of the proposed joint predistortion linearizer is shown in Fig. 1.

It should be noted that the joint predistortion linearizer works only when we first optimize the polynomial coefficients before we adaptively update the LUT entries, not the other way around. Therefore, the update procedure can be divided into two modes: P-mode and L-mode, respectively, for the optimization of the polynomial and the LUT predistorters. In the P-mode, as shown in Fig. 1, we first bypass the LUT predistorters. Thus, the update procedure of the polynomial coefficients is the same as that described before, (5)–(8). After the P-mode is over, that means the errors between $v_m(k)$ and $v_a(k)$ can not be further reduced. We assume the coefficients of the polynomial predistorter $\{p_{2l+1}(k)\}$ converge to $\{p_{2l+1}(N)\}$, after $N$ iterations in the P-mode. The number of the P-mode iterations, $N$, is decided on the fly when the difference between successive errors falls below a predetermined scenario-dependent threshold. Next, we
switch to the L-mode. The polynomial coefficients of polynomial predistorter do not need to be updated in the following L-mode. In the L-mode, as shown in Fig. 1, the converged polynomial predistorter is considered as part of the PA and the LUT predistorter is included in the system as the new adaptive predistorter. Therefore, the LUT update procedure in the L-mode is similar to (10)

\[
    F_L(Q(r_m(N + k + I_{N+k})), N + k + I_{N+k})
    = F_L(Q(r_m(N + k)), N + k) + \mu \nu_k(N + k)(v_m(N + k)
    - v_a(N + k))
\]

where \(I_{N+k}\) is the minimum integer that makes \(Q(r_m(N + k + I_{N+k})) = Q(r_m(N + k))\), \(\mu\) is the LMS stepsize, and all the LUT entries are initialized at one. However, the PA output in the L-mode becomes

\[
    v_a(N + k) = v_d(N + k) \cdot G(r_d(N + k))
    = v_d'(N + k) \cdot F_P(r_d'(N + k))
    \cdot G_P(|v_d'(N + k)F_P(r_d'(N + k))|)
    = v_d'(N + k) \cdot G_P(r_d'(N + k))
\]

where \(v_d'\) is the baseband signal between the LUT and the polynomial distorters, \(r_d' = |v_d'|\), and \(G_P(r_d'(N + k))\) can be considered as the characteristic function of a new nonlinear PA which includes the polynomial predistorter. In this way, with much smaller slopes in the characteristic function of \(G_P(\cdot)\), the polynomial predistorter helps effectively minimize the quantization distortion that the small-size LUT predistorter might suffer. Note that we assume a memoryless PA in this brief. The performance of the proposed approach may degrade when the PA has memory.

Reversing the optimization order between the P-mode and the L-mode, as done in [11], [12], may produce little improvement in the PA linearity. That is, by applying the small-size LUT predistorter first, it is impossible for the polynomial predistorter to help minimize the quantization distortion and it is also impossible for the LUT predistorter to help minimize the curve-fitting distortion. Simulation results in the Section IV will illustrate this phenomenon.

D. Complexity Comparison

In order to make a straightforward comparison of the complexity among the polynomial only, the LUT only and the proposed linearizers, we choose an Order-7 polynomial for the polynomial only linearizer, a 64-entry LUT for the LUT only linearizer, and an Order-3 polynomial plus a 16-entry LUT for the proposed joint linearizer, since the linearizers with such settings have similar IMD performance, as will be shown later in Section IV. In addition, the resolution of input data is assumed to be 10 bits. After carefully calculations, the hardware complexity of the proposed joint linearizer requires 5412 transistors, which is relative lower than 8540 transistors required for the polynomial-only linearizer and 7768 transistors required for the LUT-only linearizer, respectively, by 60% and 40%. Next, let us compare the computational complexity between the polynomial only linearizer and the proposed joint linearizer. Note that we ignore the computational complexity of the LUT only linearizer since the advantage of the proposed joint linearizer against the LUT only linearizer is shorter convergence time, not lighter computational loading. We assume the sampling rate of the transmit signal is \(f_s\). For example, like the popular indoor wireless LAN, a wideband communication system with bandwidth of 20 MHz and with an oversampling factor of 4, has its \(f_s\) equal to \(8 \times 10^6\), which is enormous. The computational complexity in terms of the number of multiplications per second can be evaluated as follows.

1) For an Order-7 polynomial, we need ten real multiplications per signal sample, or \(10 \cdot f_s\) real multiplications per second.

2) For the proposed joint linearizer, the computational complexity of the polynomial part is mainly restricted to the P-mode during the initialization period, since the computational complexity of the L-mode is negligible. Therefore, in the long run, the computational complexity of the proposed linearizer is virtually negligible as LUT-only linearizer. However, during the initialization period, \(4 \cdot f_s\) real multiplications per second are required.

IV. Simulation Results

In this section, we try an 8-tone test, which causes much more severe nonlinear distortion and IMD, so that we can better emulate practical modulated signals in a digital communication system [13]. We also choose a popular TWT PA model described by [10] for simulations. The AM/AM and the AM/PM properties as functions of the TWT PA model with input amplitude \(r\) can be expressed as

\[
    M(r) = \frac{2r}{1 + r^2},
\]

\[
    \Phi(r) = \frac{2r^2}{1 + r^2} \Phi_0, \quad \Phi_0 = \frac{\pi}{6}
\]

where \(M(r)\) and \(\Phi(r)\) are, respectively, the amplitude response and the phase response of the PA nonlinear characteristic function, \(G(\cdot)\). Note that the memory effects of the PA is ignored in our simulations.

A. Intermodulation Distortion

The simulation results of the IMD performance are shown in Fig. 5. In parts (a) and (b), the signal is transmitted without any predistorter and, respectively, with IBOs of 2 and 15 dB. With IBO=15 dB, the simulation results in part (b) indicate a reasonable IMD reduction, although power backoff is not an efficient way of PA linearization. In parts (c) and (d), the conventional predistorters, polynomial and LUT, are tested, respectively. We can see that the polynomial predistorter up to at least Order-7 can provide a satisfactory result but the complexity is a serious issue. Similarly, a LUT predistorter of up to at least 64 entries can also provide an acceptable outcome, but the complexity and long initialization time will limit its practical values. The proposed joint predistorter is shown in part (e), which uses an Order-3 polynomial predistorter first to trace the deep slope of the nonlinear characteristic function of the PA, and then uses a small size LUT to further compensate the residual error that
can not be compensated by the polynomial predistorter. It is observed that the performance of the joint predistorter can be as good as that of part (d). In other words, the IMD performance improves significantly if we introduce a small-size LUT linearization to cooperate with a low-order polynomial linearizer. On the other hand, higher order polynomial or large-size LUT linearizers will also help improve the IMD performance, but at a much higher cost. In part (f), the order of applying the polynomial and the LUT predistorters are exchanged: The small-size LUT is activated first and then the Order-3 polynomial predistorter follows. The performance is poor, as expected, because the large quantization distortions introduced by the small-size LUT predistorter are not easy for the polynomial predistorter to eliminate. Therefore, only with appropriate arrangement, the polynomial and the LUT predistorters can help each other to eliminate their individual drawbacks.

B. Adaptation Capability

The characteristics function of a PA drift because of all kinds of environment factors such as temperature, channel switching during operation, and aging of transistors. Therefore, the procedure of PA linearization needs to be adaptive. Illustrated in Fig. 6 is the adaptation learning curves of different pre-distortion linearizers. The polynomial predistorter can converges much faster than the LUT predistorter. The proposed joint predistorter employs an Order-3 polynomial and small-size LUT to achieve a compatible nonlinear compensation performance but enjoy a relatively faster convergence speed than the LUT-alone predistorters.

V. CONCLUSION

In this brief, we propose a joint polynomial and LUT predistorter for PA linearization. The performance of the conventional predistorter, polynomial and LUT, and the proposed joint predistorter are compared through computer simulations. From the simulation results, the proposed joint scheme provides a performance comparable to 1) that of a LUT predistorter with a large-size LUT and 2) that of a high-order polynomial predistorter. The proposed joint approach is shown to successfully mitigate the drawbacks seen in the conventional predistortion linearizers, high complexity in the polynomial predistorters and slow convergence speed in the LUT predistorters, and can be expected to become the choice for practical system implementation.

REFERENCES