A HAMMERSTEIN PREDISTORTION LINEARIZATION DESIGN BASED ON THE INDIRECT LEARNING ARCHITECTURE

Lei Ding *Raviv Raich* *G. Tong Zhou*

School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332-0250, USA

ABSTRACT

Power amplifiers (PAs) are inherently nonlinear devices and are used in virtually all communications systems. Digital baseband predistortion is a highly cost effective way to linearize the PAs, but most existing architectures assume that the PA has a memoryless nonlinearity. For wider bandwidth applications such as WCDMA, PA memory effects can no longer be ignored, and memoryless predistortion has limited effectiveness. In this paper, we model the PA as a Wiener system and construct a Hammerstein predistorter, obtained using an indirect learning architecture. Linearization performance is demonstrated on a 3-carrier UMTS signal.

1. INTRODUCTION

Power amplifiers (PAs) are indispensable components in a communication system and are inherently nonlinear. It is well known that there is an approximate inverse relationship between the PA efficiency and its linearity. Hence, nonlinear PAs are desirable from an efficiency point of view. The price paid for higher efficiency is that nonlinearity causes spectral regrowth (broadening) which leads to adjacent channel interference. It also causes in-band distortion which degrades the bit error rate (BER) performance. Newer transmission formats such as CDMA and OFDM are especially vulnerable to PA nonlinearities, due to their high peak to average power ratio; i.e. large fluctuations in their signal envelopes. In order to comply with spectral masks imposed by regulatory bodies and to reduce BER, PA linearization is necessary.

Of all linearization techniques, digital baseband predistortion is among the most cost effective. A predistorter is a functional block that precedes the PA. It generally creates an expending nonlinearity since the PA has a compressing characteristic. Ideally, we would like the PA output to be a scalar multiple of the input to the predistorter-PA chain. For a memoryless PA, (i.e.; the current output depends only on the current input), memoryless predistortion is sufficient. There has been intensive research on memoryless predistortion during the past decade [3].

For wider bandwidth applications such as WCDMA, PA memory effects can no longer be ignored. Moreover, higher power amplifiers such as those used in wireless basestations exhibit memory effects. The cause of memory effects can be electrical or electro-thermal as suggested in [7]. Memoryless predistortion for a PA with memory often results in poor linearization performance. Although Volterra series is a general nonlinear model with memory, its predistortion is complex and its real-time implementation difficult. In [2], Clark *et.al.* used a Wiener model; i.e., a linear time-invariant (LTI) system followed by a memoryless nonlinearity, to capture the nonlinear memory effects in the PA associated with wideband signals. In this paper, we also adopt the Wiener PA model, which has the advantage that its predistortion can be easily carried out. A Hammerstein system is a memoryless nonlinearity followed by a LTI system, and can therefore linearize a Wiener PA.

In the current literature, predistorters with memory mainly fall into the data predistorter category [5, 6], in the sense that predistortion is applied before the pulse shaping filter. The main drawback of data predistortion is its dependence on the signal constellation and the pulse shaping filter. Both Volterra model based [5] and Hammerstein model based [6] data predistorters have been proposed. In [5], Volterra data predistorter is constructed using the indirect learning architecture. In [6], the Hammerstein data predistorter is obtained using a stochastic gradient method.

As opposed to data predistortion, we shall pursue signal predistortion in this paper; i.e., predistortion occurs after the pulse shaping filter. To construct a Hammerstein predistorter, one approach is to first identify the Wiener PA and then find the Hammerstein predistorter as its inverse. Since Wiener system identification is generally more difficult to carry out than Hammerstein system identification, we pursue an alternative approach which generates the Hammerstein predistorter without first identifying the Wiener PA. Unlike [6], our Hammerstein predistorter will be constructed using an indirect learning architecture similar to the one used in [5]. In this setup, finding the predistorter is essentially equivalent to identifying a Hammerstein system. the PA can be modeled as a Wiener system,

2. INDIRECT LEARNING ARCHITECTURE

Fig. 1 shows the indirect learning structure that is used for Hammerstein predistorter identification. The PA has a Wiener structure (LTI followed by memoryless nonlinearity). The feedback path labeled "Predistorter Training" (block A) has a Hammerstein structure if we view $y(n)/K$ as its input and $\hat{z}(n)$ as its output. The actual predistorter is an exact copy of the feedback path (copy of A); it has $x(n)$ as its input and $z(n)$ as its output. Ideally, we would like $y(n) = Kx(n)$, which renders $z(n) = \hat{z}(n)$ and the er-

0-7803-7402-9/02/$17.00 ©2002 IEEE

Authorized licensed use limited to: James Proctor. Downloaded on October 29,2024 at 19:40:24 UTC from IEEE Xplore. Restrictions apply.

Exhibit 1027
Nokia of America Corporation, et al. v. Smart RF Inc.

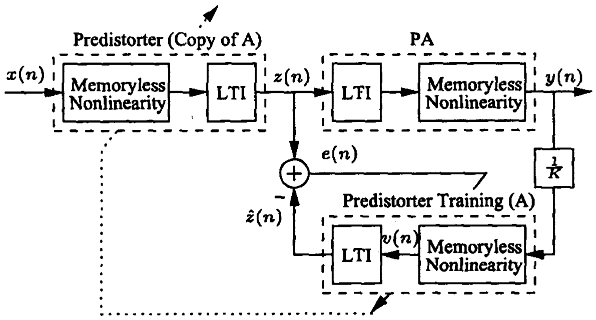

Figure 1. The indirect learning architecture for the Hammerstein predistorter.

ror term $e(n) = 0$. Given $y(n)$ and $z(n)$, our task is to find the parameters of block A, which yields the predistorter. The algorithm converges when the error energy $||e(n)||^2$ is minimized.

Here we consider that the PA characteristics do not change rapidly with time – changes in PA characteristics are often due to temperature drift, aging etc which have long time constants. After gathering a block of $y(n)$ and $z(n)$ data samples, the training branch (block A) can process the data off-line, which lowers the requirement of the processing power of the predistortion system. Once the predistorter identification algorithm has converged, the new set of parameters are plugged into the high speed predistorter, which can be readily implemented by Application-Specific Integrated Circuits (ASIC) or Field Programmable Gate Arrays (FPGA). When the predistorter coefficients have been found and it is believed that the PA characteristics are hardly changing, the setup in Fig. 1 can be run in open loop; i.e., we temporarily shutdown the training branch, until changes in PA characteristics require a new predistorter.

3. IDENTIFICATION OF THE HAMMERSTEIN PREDISTORTER

The predistorter training branch can be described by:

$$v(n) = \sum_{k=0}^{(K-1)/2} c_{2k+1}\, y(n)|y(n)|^{2k}, \qquad (1)$$

$$z(n) = \sum_{p=1}^{P} a_p z(n-p) + \sum_{q=0}^{Q} b_q v(n), \qquad (2)$$

which implies that for the predistorter, we model the memoryless nonlinearity as an odd-order polynomial and the LTI system as a general pole/zero system. Combining the two equations above, we obtain

$$z(n) = \sum_{p=1}^{P} a_p z(n-p) + \sum_{q=0}^{Q} b_q \left(\sum_{k=0}^{(K-1)/2} c_{2k+1} y(n-q)|y(n-q)|^{2k} \right). \qquad (3)$$

Given $y(n)$ and $z(n)$, our objective is to estimate the a_p, b_q and c_{2k+1} coefficients. Parameter estimation of this model is a classical Hammerstein system identification problem. If no additional assumptions are made on the system's input signal $y(n)$, iterative Newton and Narendra-Gallman algorithms are the two most popular iterative estimation methods [4]. The two algorithms exhibit similar performance as shown in [4]. The main drawback of these algorithms is that they are sensitive to the initial guesses and may converge to a local minimum. A recent method proposed by Bai [1] uses an optimal two stage identification algorithm, which can lead to a global optimum. The model structure introduced in [1] is a Hammerstein system followed by a memoryless nonlinearity. However, we can easily modify the results of [1] to suit our model. Note that for a given set of $\{y(n), z(n)\}$ values, the b_q's and the c_{2k+1}'s are not unique (i.e.; multiplying b_q with a constant and dividing c_{2k+1} by the same constant yields the same model). To avoid this problem, we assume that $\sum_{q=0}^{Q} |b_q|^2 = 1$ and the real part of b_0 is positive as suggested in [1].

Next, we will review the Narendra-Gallman (NG) and the optimal two stage identification (LS/SVD) algorithms.

3.1. Narendra-Gallman algorithm

The NG algorithm starts with initial guesses for the a_p and b_q coefficients, denoted by $a_p^{(0)}$ and $b_q^{(0)}$, respectively. At the ith iteration eq. (3) can be rewritten as

$$z(n) - \sum_{p=1}^{P} a_p^{(i)} z(n-p) = \sum_{k=0}^{(K-1)/2} c_{2k+1} u_{2k+1}(n) \qquad (4)$$

$$u_{2k+1}(n) = \sum_{q=0}^{Q} b_q^{(i)} y(n-q)|y(n-q)|^{2k}.$$

At this stage out objective is to solve for c_{2k+1}. Using matrix notations we can reformulate eq. (4) as

$$\mathbf{z}_0 - \mathbf{Z}\mathbf{a}^{(i)} = \mathbf{U}\mathbf{c}, \qquad (5)$$

where $\mathbf{Z} = [\mathbf{z}_1, \ldots, \mathbf{z}_P]$, $\mathbf{z}_l = [\mathbf{0}_l^T, z(1), \ldots, z(N-l)]^T$, where $\mathbf{0}_l$ is a $1 \times l$ all zero vector, $\mathbf{a}^{(i)} = [a_1^{(i)}, \cdots, a_P^{(i)}]^T$, $\mathbf{U} = [\mathbf{u}_1, \cdots, \mathbf{u}_K]$, $\mathbf{u}_{2k+1} = [u_{2k+1}(1), \cdots, u_{2k+1}(N)]^T$, and $\mathbf{c} = [c_1, \cdots, c_K]^T$. The least-squares solution for eq. (5) is

$$\hat{\mathbf{c}}^{(i+1)} = (\mathbf{U}^H\mathbf{U})^{-1}\mathbf{U}^H \left(\mathbf{z}_0 - \mathbf{Z}\mathbf{a}^{(i)}\right), \qquad (6)$$

where ^{H} denotes Hermitian transpose. In the second step, based on the $c_{2k+1}^{(i+1)}$'s obtained, we rewrite eq. (3) as,

$$\mathbf{z}_0 = \mathbf{Z}\mathbf{a} + \mathbf{V}\mathbf{b} = [\mathbf{Z}\ \mathbf{V}] \begin{bmatrix} \mathbf{a} \\ \mathbf{b} \end{bmatrix}, \qquad (7)$$

where $\mathbf{V} = [\mathbf{v}_1, \cdots, \mathbf{v}_Q]$, $\mathbf{v}_l = [\mathbf{0}_l^T, v(1), \cdots, v(N-l)]^T$, $\mathbf{b} = [b_0, \cdots, b_Q]^T$, and $v(n)$ is given in eq. (1). The least-squares solution for eq. (7) is,

$$\begin{bmatrix} \hat{\mathbf{a}}^{(i+1)} \\ \hat{\mathbf{b}}^{(i+1)} \end{bmatrix} = \left([\mathbf{Z}\ \mathbf{V}]^H [\mathbf{Z}\ \mathbf{V}]\right)^{-1} [\mathbf{Z}\ \mathbf{V}]^H \mathbf{z}_0, \qquad (8)$$

With the new $\hat{\mathbf{a}}^{(i+1)}$ and $\hat{\mathbf{b}}^{(i+1)}$ estimates, we can go back to the first step and continue until the algorithm converges.

Authorized licensed use limited to: James Proctor. Downloaded on October 29,2024 at 19:40:24 UTC from IEEE Xplore. Restrictions apply.

Exhibit 1027
Nokia of America Corporation, et al. v. Smart RF Inc.

3.2. Optimal two stage identification algorithm

Since the difficulty in estimating the b_q's and c_{2k+1}'s is that they appear together as the coefficient on the r.h.s. of eq. (3), if we define

$$d_{q,2k+1} = b_q c_{2k+1}, \tag{9}$$

we can first estimate $d_{q,2k+1}$ using least-squares and then find b_q and c_{2k+1} from $d_{q,2k+1}$. Substituting eq. (9) into eq. (3), we obtain

$$z(n) = \sum_{p=1}^{P} a_p z(n-p) + \sum_{q=0}^{Q} \sum_{k=0}^{(K-1)/2} d_{q,2k+1}\, g_{q,2k+1}(n), \tag{10}$$

where $g_{q,2k+1}(n) = y(n-q)|y(n-q)|^{2k}$. Rewriting in a matrix form, we obtain

$$\mathbf{z}_0 = \mathbf{Za} + \mathbf{Gd} = [\mathbf{Z}\ \mathbf{G}] \begin{bmatrix} \mathbf{a} \\ \mathbf{d} \end{bmatrix}, \tag{11}$$

where $\mathbf{G} = [\mathbf{g}_{01}, \cdots, \mathbf{g}_{0K}, \cdots, \mathbf{g}_{Q1}, \cdots, \mathbf{g}_{QK}]$, $\mathbf{g}_{q,2k+1} = [g_{q,2k+1}(1), \cdots, g_{q,2k+1}(N)]^T$, and $\mathbf{d} = [d_{01}, \cdots, d_{0K}, \cdots, d_{Q1}, \cdots, d_{QK}]^T$. The least-squares solution for eq. (11) is

$$\begin{bmatrix} \hat{\mathbf{a}} \\ \hat{\mathbf{d}} \end{bmatrix} = \left([\mathbf{Z}\ \mathbf{G}]^H [\mathbf{Z}\ \mathbf{G}]\right)^{-1} [\mathbf{Z}\ \mathbf{G}]^H \mathbf{z}_0, \tag{12}$$

Eq. (9) can be alternatively expressed as

$$\mathbf{D} = \begin{bmatrix} d_{01} & d_{03} & \cdots & d_{0K} \\ d_{11} & d_{13} & \cdots & d_{1K} \\ \vdots & \vdots & & \vdots \\ d_{Q1} & d_{Q3} & \cdots & d_{QK} \end{bmatrix} = \mathbf{bc}^{\mathbf{T}}, \tag{13}$$

where $\mathbf{b} = [b_0, \ldots, b_Q]^T$, $\mathbf{c} = [c_1, \ldots, c_K]^T$. Since the matrix $\mathbf{D}$ has rank one, a natural way to estimate $\hat{\mathbf{b}}$ and $\hat{\mathbf{c}}$ from $\hat{\mathbf{D}}$ is to perform a singular value decomposition (SVD) on $\hat{\mathbf{D}}$ and then find the eigenvectors corresponding to the largest singular value. Let the SVD of $\hat{\mathbf{D}}$ be given by,

$$\hat{\mathbf{D}} = \sum_{i=1}^{min\{(Q+1),(K+1)/2\}} \sigma_i \mu_i \nu_i^H, \tag{14}$$

where μ_i's and ν_i's are $Q+1$ and $(K+1)/2$ dimensional orthonormal vectors, respectively. Then $\hat{\mathbf{b}}$ and $\hat{\mathbf{c}}$ can be estimated as

$$\hat{\mathbf{b}} = s_\mu \mu_1, \quad \hat{\mathbf{c}} = s_\mu \sigma_1 \nu_1^*, \tag{15}$$

where * denotes conjugate and s_μ is the first non-zero element of μ_1. These estimates can be shown to be the closest $\hat{\mathbf{b}}$ and $\hat{\mathbf{c}}$ to $\hat{\mathbf{D}}$ in the least-squares sense [1].

In summary, the NG algorithm is a simple and robust algorithm. Although it may have convergence problems, it can perform well in many cases as will be shown in the next section. The LS/SVD algorithm avoids the potential local minimum problem of the NG algorithm. However, using SVD to find the b_q's and c_{2k+1}'s may not result in the best b_q's and c_{2k+1}'s that minimize the squared error criterion. Our examples in the next section will show that both work well for identifying the Hammerstein predistorter although one may outperform the other in a particular scenario.

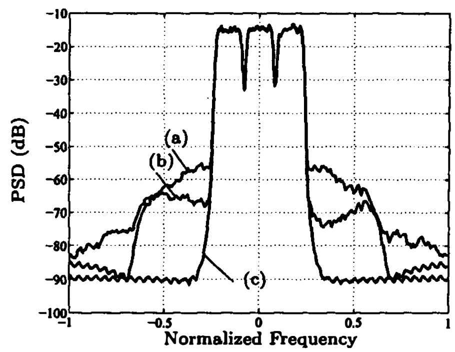

Figure 2. Comparison of the PSDs. (a) Output without predistortion; (b) Output with memoryless predistortion; (c) Output with Hammerstein predistortion, NG and LS/SVD algorithms (similar performance).

4. SIMULATIONS

In this section, we illustrate through computer simulations the performance of the Hammerstein predistorter identified using the indirect learning architecture. In the first example, the LTI portion of the Wiener PA model has a pole/zero form, whose system function is given by

$$H(z) = \frac{1 + 0.3z^{-2}}{1 - 0.2z^{-1}}. \tag{16}$$

For the memoryless nonlinear portion of the Wiener PA model, we use a 5th order nonlinearity with coefficients,

$$\begin{aligned} c_1 &= 14.9740 + 0.0519j, \quad c_3 = -23.0954 + 4.9680j, \\ c_5 &= 21.3936 + 0.4305j, \end{aligned} \tag{17}$$

which were extracted from an actual Class AB PA.

The baseband input signal is a 3-carrier Universal Mobile Telecommunications System (UMTS) signal. Hammerstein predistorter identification is carried out based on 8000 data samples. The predistorter parameters usually converge after a few iterations. Next, we compare the spectra of the input and output signals to asses the effectiveness of the predistorter in reducing spectral regrowth. In this example, we assume that the LTI portion of the Hammerstein predistorter is a pole/zero system with two poles and one zero (correct model orders for the inverse of the $H(z)$ of eq. (16)). In addition, we make the assumption that the nonlinearity of the predistorter is 5th order.

Performance of predistorter identified with the LS/SVD and NG algorithms is demonstrated in Fig. 2. Both algorithms fully suppress the spectral regrowth exhibited by the PA output when no predistortion is applied. In contrast, we observe in Fig. 2 that 5th order memoryless predistortion does not fully suppress the spectral regrowth.

In the second example, the LTI portion of the Wiener PA is $H(z) = 1+0.3z^{-2}$ (FIR), and the LTI portion of the Hammerstein predistorter is assumed to be FIR as well. Our objective here is to see whether the algorithm can correctly

Authorized licensed use limited to: James Proctor. Downloaded on October 29,2024 at 19:40:24 UTC from IEEE Xplore. Restrictions apply.

Exhibit 1027
Nokia of America Corporation, et al. v. Smart RF Inc.

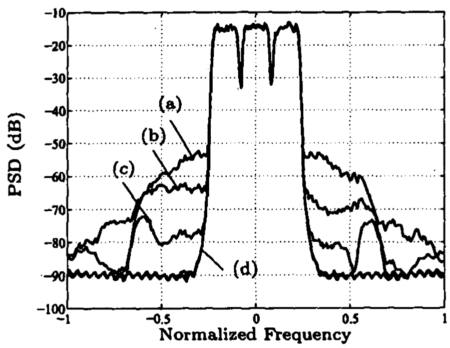

Figure 3. Comparison of the PSDs. (a) Output without predistortion; (b) Output with memoryless predistortion; (c) Output with Hammerstein predistortion (NG); (d) Output with Hammerstein predistortion (LS/SVD).

identify an FIR filter that approximates the inverse of the FIR system in the PA. We assume that the FIR system in the predistorter has 15 taps. The results are shown in Fig. 3. The two algorithms exhibit different behaviors in this time: the NG algorithm performs worse than the LS/SVD algorithm. When examining the concatenated response of the two LTI blocks (one from the Wiener PA and the other from the Hammerstein predistorter), we observe that the predistorter's LTI system identified by the NG algorithm can only compensate for the PA's LTI system within the signal bandwidth. However, the LS/SVD algorithm is able to find a good FIR system for the predistorter, both within and outside of the signal bandwidth.

In the third example, we perturbed the Wiener PA model coefficients so it is a full Volterra model (not Wiener any more). Our objective is to see whether the Hammerstein predistorter has any robustness. The result is shown in Fig. 4. We still observe significant reduction of spectral regrowth with the Hammerstein predistorter.

In all cases, memoryless predistortion is not very effective in suppressing spectral regrowth, which underscores the notion that PA memory effects must be taken into account when designing the predistorter.

5. CONCLUSIONS

We employed the indirect learning structure to identify the Hammerstein predistorter for a PA modeled by a Wiener model. We compared the performance of two Hammerstein system identification algorithms; i.e., the NG and LS/SVD algorithms, in this context. For a Wiener model with a simple pole/zero LTI structure, both algorithms show similar performance. However, when the LTI portion of the Wiener PA as well as that of the Hammerstein predistorter are FIR, the LS/SVD algorithm outperforms the NG algorithm. Simulation results illustrate the effectiveness of the proposed predistorter design.

Acknowledgment: This work was supported in part by National Science Foundation grant MIP 9703312 and by the State of Georgia's Yamacraw Initiative. The authors would also like to thank Dr. Zhengxiang Ma and Dr. Dennis R. Morgan of Bell Laboratories, Lucent Technologies for inspiring this work.

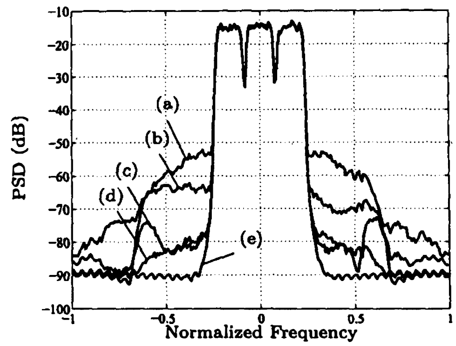

Figure 4. Comparison of the PSDs. (a) Output without predistortion; (b) Output with memoryless predistortion; (c) Output with Hammerstein predistortion (NG); (d) Output with Hammerstein predistortion (LS/SVD); (e) Input signal.

REFERENCES

[1] E. W. Bai, "An optimal two stage identification algorithm for Hammerstein-Wiener nonlinear systems," *Proc. American Contr. Conf.*, pp. 2756-2760, Philadephia, Pennsylvania, Jun. 1998.

[2] C. J. Clark, G. Chrisikos, M. S. Muha, A. A. Moulthrop and C. P. Silva, "Time- domain envelope measurement technique with application to wideband power amplifier modeling," *IEEE Trans. Microwave Theory Tech.*, vol. 46, no. 12, pp. 2531- 2540, Dec. 1998.

[3] S. C. Cripps, *RF Power Amplifiers for Wireless Communications*, Artech House, Norwood, MA, 1999.

[4] E. Eskinat, S. H. Johnson, and W. L. Luyben, "Use of Hammerstein models in Identification of nonlinear systems," *AIChE J.*, vol. 37, no. 2, pp. 255-267, Feb. 1991.

[5] C. Eun and E. J. Powers, "A predistorter design for a memory-less nonlinearity proceded by a dynamic linear system," *Proc. GLOBECOM*, pp. 152-156, Singapore, Nov. 1995.

[6] H. W. Kang, Y. S. Cho, and D. H. Youn, "On compensating nonlinear distortions of an OFDM system using an efficient adaptive predistorter," *IEEE Trans. Commun.*, vol. 47, no. 4, Apr. 1999.

[7] J. H. K. Vuolevi, T. Rahkonen, and J. P. A. Manninen, "Measurement technique for characterizing memory effects in RF power amplifiers," *IEEE Trans. Microwave Theory Tech.*, vol. 49, no. 8, pp. 1383-1388, Aug. 2001.

Authorized licensed use limited to: James Proctor. Downloaded on October 29,2024 at 19:40:24 UTC from IEEE Xplore. Restrictions apply.

Exhibit 1027
Nokia of America Corporation, et al. v. Smart RF Inc.