Feedback Quantization in Crosscorrelation Predistorters

André B.J. Kokkeler

University of Twente

P.O. Box 217, 7500 AE Enschede, The Netherlands.

Phone: +31 53 4894291, Fax: +31 53 4894571,

kokkeler@cs.utwente.nl

Abstract—**Amplification of signals with fluctuating envelopes inevitably leads to distortion because of non-linear behavior of the Power Amplifier (PA). Digital Predistortion can counteract these non-linear effects. In this paper, the crosscorrelation predistorter is described and the effects of quantization in the feedback path are presented. One of the effects is that the quantization noise becomes correlated with the signal to be quantized, resulting in reduced performance of predistortion. A technique to reduce these effects is to inject a dither signal before quantization. Because of the quantization noise and dither signal, more data has to be used to obtain estimates of the PA behavior that are accurate enough for effective predistortion.**

Keywords— **Power Amplifiers, Digital Predistortion, Crosscorrelation, Dither, Analog-to-Digital Conversion, Memory Polynomial.**

I. INTRODUCTION

ONE of the techniques to linearize Power Amplifiers (PAs) is digital predistortion (see [1]). Predistortion followed by the PA (and its inherent distortion) should result in linear amplification. In digital predistortion, the digital baseband signal is predistorted before it is converted to the analog domain, frequency translated to RF and amplified (see figure 1).
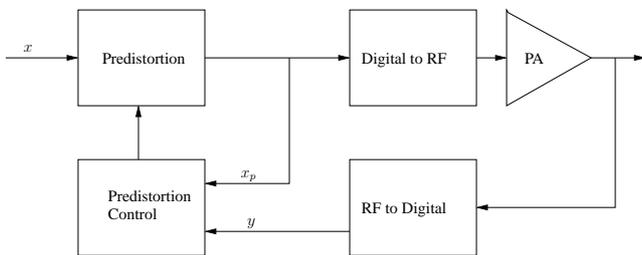

Fig. 1. Digital Predistortion

Because the input-output relation of the PA changes in time, a control mechanism constantly adapts the predistortion. A small fraction of the PA output is fed back and converted from RF to baseband, leading to the signal y. Together with the predistorted signal x_p, the input-output relation of the PA is determined in the Predistortion Control block. This relation is inverted and used for predistortion within the Predistortion block (see [2]). In our approach, the PA is modeled by an orthogonal memory polynomial (see [3]).

The resolution of the Analog to Digital Converter (ADC) in the feedback path is an important design issue. Recent communication standards make use of relatively large bandwidths. The combination of high resolution and high bandwidth leads to complex ADC designs, which results in high prices and high power consumption. Therefore, it is interesting to investigate digital predistortion using low resolution ADCs in the feedback path. In this document, first the crosscorrelation predistorter is introduced. Second, the performance of the crosscorrelation predistorter is presented using a high resolution ADC in the feedback path. Third, the results of simulations for different resolutions in the feedback path are presented. Fourth, dithering is introduced to increase the performance of the crosscorrelation predistorter in case of coarse quantization in the feedback path.

II. THE CROSSCORRELATION PREDISTORTER

A. General Scheme

The predistorted signal x_p and the PA output (converted to baseband) y are used to determine the non linear characteristics of the PA. The output is written as a function of the input: $y = f(x_p)$. The function f can be described by means of orthogonal polynomial basis functions $\psi_k(x)$ (see [3]):

$$y(t) = \sum_{k=1}^{K} \sum_{\tau=0}^{\tau_{max}-1} a_{k\tau}\psi_k(x_p(t-\tau)) \tag{1}$$

We will only give odd polynomials up to the fifth order:

$$\begin{aligned}
\psi_1(x) &= x \\
\psi_3(x) &= 15\mid x\mid^2 x - 20\mid x\mid x + 6x \\
\psi_5(x) &= 210\mid x\mid^4 x - 504\mid x\mid^3 + 420\mid x\mid^2 x \\
&\quad - 140\mid x\mid x + 15x
\end{aligned} \tag{2}$$

A general description of the crosscorrelation predistorter starts with the following definition:

$$\gamma_{k\tau}(t) = \psi_k(x_p(t-\tau)) \tag{3}$$

for $t \in \mathbb{Z}$. Via linear combination of these signals, an estimate of y can be constructed using the least squares criterion. As indicated in [4], the complexity of this algorithm is relatively high. To reduce complexity, the crosscorrelation predistorter was developed ([4], [5]). In the crosscorrelation predistorter, the signals $\gamma_{k\tau}$ and y are crosscorrelated with a reference signal. We chose to crosscorrelate with a single-bit representation x_Q of x_p. Single-bit quantization is defined as:

$$x_Q(x_p) = \text{sign}(\text{Re}(x_p)) + j\text{sign}(\text{Im}(x_p)) \tag{4}$$

where sign() is defined as:

$$\text{sign}(x) = \begin{cases} -1 & x < 0 \\ 1 & x \geq 0 \end{cases} \qquad (5)$$

$\text{Re}(x)$ indicates the real part of a complex value x and $\text{Im}(x)$ the imaginary part. Using a single-bit quantized value as one of the operands of a crosscorrelation, drastically reduces the complexity because the multiplications involved become very simple to implement. The number of points (or lags) of the crosscorrelation is even and equals N. Using vector notations, the single-bit quantized signal equals:

$$\mathbf{x_Q} = \left[x_Q(x_p(\tfrac{1}{2}N + 1)), ..., x_Q(x_p(T - \tfrac{1}{2}N + 1)) \right]^T \qquad (6)$$

T is the number of consecutive samples, available to the predistorter and is generally much larger than N. We define the matrix $\mathbf{\Gamma_{k\tau}}$ for $k = 1, ..., K$ and $\tau = 0, ..., \tau_{max} - 1$ as:

$$\mathbf{\Gamma_{k\tau}} = \Big[[\gamma_{k\tau}(1), ..., \gamma_{k\tau}(T - N + 1)]^T, ..., \\ [\gamma_{k\tau}(N), ..., \gamma_{k\tau}(T)]^T \Big]^T \qquad (7)$$

The matrix $\mathbf{Y}$ is defined similarly, using the signal y. The crosscorrelations are defined as:

$$\mathbf{r_{k\tau}} = \mathbf{\Gamma_{k\tau}}\mathbf{x_Q^*} \qquad (8)$$
$$\mathbf{r} = \mathbf{Y}\mathbf{x_Q^*} \qquad (9)$$

The crosscorrelation vectors $\mathbf{r_{k\tau}}$ and $\mathbf{r}$ are tapered with a Hanning taper:

$$h(j) = \frac{1}{2}(1 - \cos(2\pi(j - 1)/(N - 1))) \qquad (10)$$
$$\mathbf{H} = \text{diag}(h(1), ..., h(N)) \qquad (11)$$

where 'diag()' indicates a diagonal $N \times N$ matrix. A Discrete Fourier Transform is applied to the tapered crosscorrelation vectors:

$$\mathbf{f_{k\tau}} = \mathbf{W}\mathbf{H}\mathbf{r_{k\tau}} \qquad (12)$$
$$\mathbf{f} = \mathbf{W}\mathbf{H}\mathbf{r} \qquad (13)$$

where $\mathbf{W}$ equals the DFT kernel.

If the vectors (spectra) $\mathbf{f_{k\tau}}$ are concatenated into a matrix $\mathbf{F} = [\mathbf{f_{10}}, ..., \mathbf{f_{K0}}, ..., \mathbf{f_{1\tau_{max}}}, ..., \mathbf{f_{K\tau_{max}}}]$ and the vector $\mathbf{a}$ is defined as $\mathbf{a} = [a_{10}, ..., a_{K0}, ..., a_{1\tau_{max}}, ..., a_{K\tau_{max}}]$, the memory polynomial predistorter is described in the frequency domain as:

$$\mathbf{f} = \mathbf{F}\mathbf{a} \qquad (14)$$

This expression is rewritten into an equivalent expression:

$$\mathbf{g} = \mathbf{G}\mathbf{a} \qquad (15)$$

where

$$g_{k\tau}(n) = f_{k\tau}(n)/f(n) \qquad (16)$$
$$\mathbf{g_{k\tau}} = [g_{k\tau}(1), ..., g_{k\tau}(N)]^T \qquad (17)$$
$$\mathbf{G} = [\mathbf{g_{10}}, ..., \mathbf{g_{K0}}, ..., \mathbf{g_{1\tau_{max}}}, ..., \mathbf{g_{K\tau_{max}}}] \qquad (18)$$
$$\mathbf{g}(n) = 1, \quad n = 1, ..., N \qquad (19)$$

The least-squares solution $\hat{\mathbf{a}}$ equals:

$$\hat{\mathbf{a}} = (\mathbf{G^H}\mathbf{G})^{-1}\mathbf{G^H}\mathbf{g} \qquad (20)$$

where $^{\mathbf{H}}$ indicates the complex conjugate transpose.

The vector $\hat{\mathbf{a}}$ contains the coefficients of the memory polynomial that describes the behavior of the PA from input to output. Within the Predistortion block, the inverted relation is used to predistort the input signal x. This inversion procedure is described in [5].

III. PERFORMANCE OF THE CROSSCORRELATION PREDISTORTER

A simulator, based on the algorithm presented in the previous section, has been implemented. As a signal source we used two data generators: one for the real part of the signal and one for the imaginary part. A data generator is based on the signal source used in [6] and generates sine waves with different consecutive frequencies, equal amplitudes and randomly selected phase. The oversampling factor equals 10.

Our PA model is a Wiener-Hammerstein model, presented in [7]. This model consists of an IIR filter $H(z)$ followed by a memoryless polynomial which on its turn is followed by a second IIR filter $G(z)$. The filter specifications are:

$$H(z) = \frac{1 + 0.5z^{-2}}{1 - 0.2z^{-1}}, \quad G(z) = \frac{1 - 0.1z^{-2}}{1 - 0.4z^{-1}} \qquad (21)$$

The memoryless polynomial equals:

$$w(n) = \sum_{k=1}^{K} b_k \, v(n) \mid v(n) \mid^{k-1} \qquad (22)$$

where v indicates the output of filter H and w the input of filter G. The coefficients are: $b_1 = 1.0108 + 0.0858j$, $b_3 = 0.0879 - 0.1583j$, $b_5 = -1.0992 - 0.8891j$, $b_i = 0$ otherwise. In our simulations, we used blocks of 8K samples ($T = 8192$) and 64-point crosscorrelation functions ($N = 64$). For the predistorter, we used only odd polynomials up to the fifth order ($k = 1, 3, 5$) and the memory depth is 2 ($\tau = 0, 1$).

The performance of digital predistortion is characterized by the ACI level: the power in the channel adjacent to the primary channel. The bandwidth of the adjacent channel equals the bandwidth of the primary channel. The power in the adjacent channel is related to the power in the primary channel and for that reason measured in dB. In figure 2, the average ACI levels are presented. The average ACI levels with- and without predistortion are given as a function of the standard deviation of the predistorter input signal.

IV. EFFECTS OF QUANTIZATION

To analyze the effects of quantization, we extended the simulator with an ADC in the feedback path. The ADC consists of two quantizers, one for real part of the complex signal y and one for the imaginary part. For the quantizers, we use equal resolutions and a headroom factor equal to 4. By simulation, we determined the average ACI levels for different resolutions of the quantizers for a standard deviation at the input of 0.0875. The results are presented in figure 3.
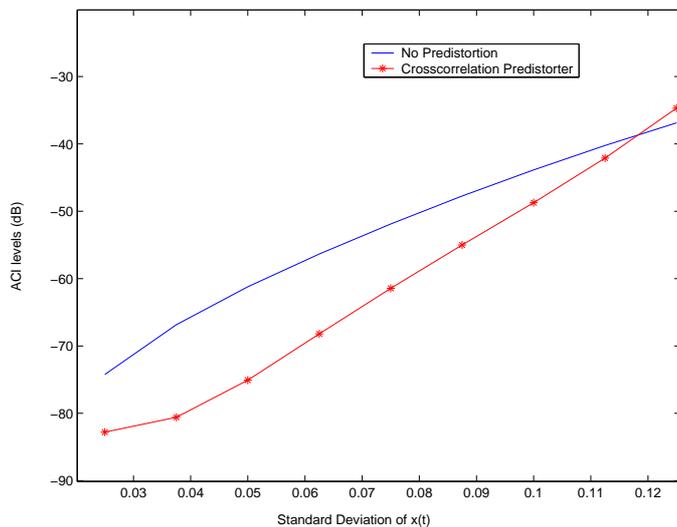
Fig. 2. Performance of the crosscorrelation predistorter

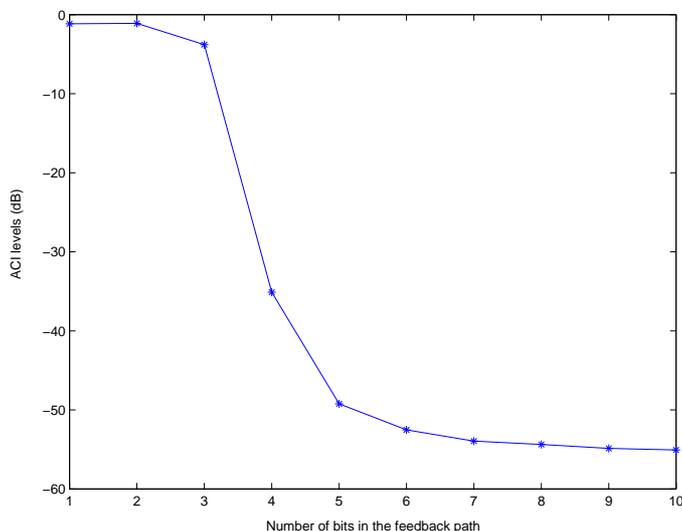Fig. 4. Performance of the predistortion schemes

Fig. 3. Effects of quantization

For resolutions higher than 4 bits, suppression of ACI levels is still achieved. For 4 bits and less, predistortion results in increased ACI levels. For these low resolutions, the quantization error is correlated with the original signal. The crosscorrelation spectrum $\mathbf{f}$ of y and x_Q becomes distorted, leading to errors in the estimates $\hat{\mathbf{a}}$ of the polynomial coefficients.

V. Effects of dithering

A well-known technique to reduce the correlation between the original signal before quantization and the quantization error, is dithering. In our simulations, we added independent uniformly distributed quantization noise with 1-lsb range to both the real and imaginary parts of the signal. When using a small number of bits in the feedback path, e.g. two bits, the quantization error together with the dither signal lead to low signal-to-noise ratios (SNRs) of the crosscorrelation spectrum $\mathbf{f}$. The SNR can be increased by using more data for estimating the polynomial coefficients, in other words, by increasing the Integration Length.
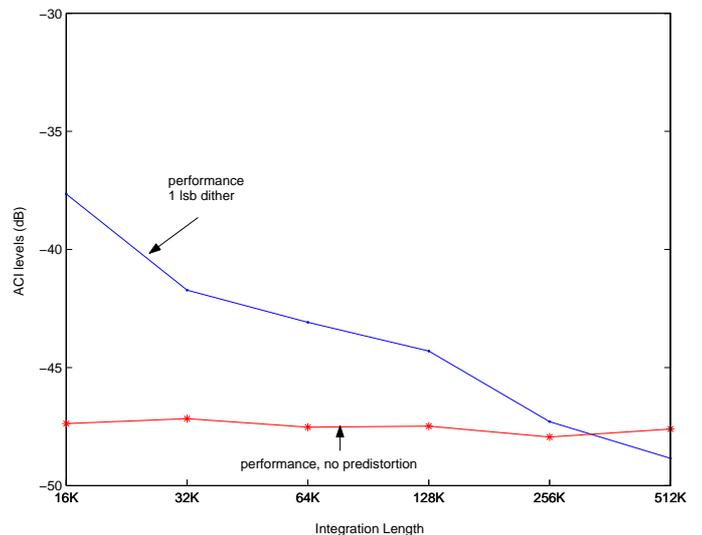
In figure 4, the ACI levels in case of 2 bit quantization, including dithering, are presented for different Integration Lengths.

Without dithering, the ACI levels for 2 bit quantization were around 0 dB (see figure 3). By adding the dither signal and increasing the Integration Length by a factor 2, the ACI levels have been reduced to -37.6 dB. By further increasing the Integration Length, the ACI levels are reduced as well. By using enough data, suppression of ACI levels can be achieved, even when using 2 bit quanization in the feedback path.

VI. Conclusion

In this paper, it is shown that the crosscorrelation predistorter, using orthogonal polynomials, gives a suppression of power in the adjacent channels (ACI levels) over a large input-power range. Low-resolution quantization in the feedback path (less than 5 bits) results in increased ACI levels because the quantization error is correlated with the signal before quantization. This correlation can be cancelled by adding dither noise before quantization. For two bit quantization, adding a dither signal improves performance. To further increase performance, the Integration Length can be increased.

References

[1] F. Zavosh, M. Thomas, C. Thron, T. Hall, D. Artusi, D. Anderson, D. Ngo, and D. Runton, "Digital predistortion techniques for RF power amplifiers with CDMA applications," *Microwave Journal*, Oct. 1999.

[2] J. Kim and K. Konstantinou, "Digital predistortion of wideband signals based on power amplifier model with memory," *Electronics Letters*, vol. 37, no. 23, Nov. 2001, pp. 1417-1418.

[3] R. Raich, H. Qian, and G.T. Zhou, "Digital baseband predistortion of nonlinear power amplifiers using orthogonal polynomials," *Proc. ICASSP2003*, vol. 6, Apr. 6-10, 2003, pp. VI-689 - VI-692

[4] A.B.J. Kokkeler, "A Crosscorrelation Predistorter using Memory Polynomials," *Proc. ISCAS2004*, vol. 3, May 2004, pp. III 345-348.

[5] A.B.J. Kokkeler, " Performance Analysis of Crosscorrelation Predistorters," *Proc. ECWT 2004*, Oct. 2004, pp. 325-328.

[6] J. K. Cavers, "Amplifier linearization using a digital predistorter with fast adaptation and low memory requirements", *IEEE Trans. Veh. Technol.*, vol. 39, no. 4, Nov. 1990, pp. 374-383.

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