Predistorter for Power Amplifier using Flower Pollination Algorithm

Beena Jacob[1], Nisha Markose[2] and Shinu S Kurian[3]

[1, 2, 3]*Assistant Professor, Department of Computer Application, MA College of Engineering, Kothamangalam, Kerala, India*

***Abstract*- In wireless communication system, memory high power amplifier is an indispensable component but due to its presence in the transmitter section nonlinearity is originated in the system. In this project an efficient predistorter is designed using flower pollination algorithm for mitigating the distortions caused by the power amplifier. The wiener model which exhibits true output saturation characteristics is used to represent memory power amplifier model. In identification process, wiener power amplifier parameters are estimated using flower pollination algorithm. Thus obtained parameters are used for implementing hammerstein predistorter. The effectiveness of designed predistorter is evaluated by simulation of models and obtained results have been presented.**

***Index Terms*- Hammerstein model; flower pollination algorithm; memory high power amplifier; predistorter; wiener model.**

I. INTRODUCTION

Power Amplifier (PA) is an unavoidable component of wireless communication which is used at the end of the transmitter in order to produce a signal with a power suitable for transmission through an antenna. The PA performs power amplification of the input signal, but it introduces nonlinearity to the system. Nonlinearity occurs due to the AM-AM and AM-PM characteristics of the power amplifier. Power amplifier also causes distortion of signal due its memory effect, which cannot be neglected at higher bandwidth. Therefore an accurate linearization technique is needed for compensating problems caused by the power amplifier. Presently there are various methods for the linearization of the power amplifier. Among the various linearization techniques, digital baseband predistorter is considered to be the most effective because it has relatively good performance and low implementation cost. There are two types of predistorter designs one is indirect-learning [1] based PD designs in which a post-inverse polynomial filter is identified for the memory power amplifier to be compensated and then copy the post-inverse polynomial filter to form the PD and other is the direct-learning [7] based PD designs, in this input-output relation of the memory power amplifier is identified using a polynomial model and then implement a polynomial PD which directly invert the resulting polynomial HPA model.

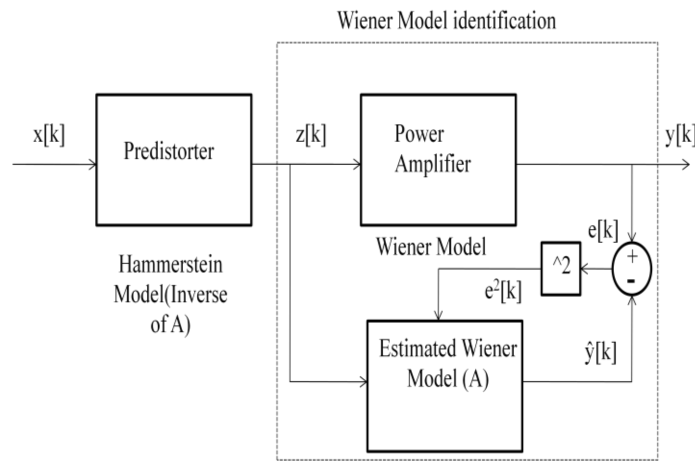

Fig. 1. Block diagram of proposed work.

In this paper, the proposed predistorter is based on direct learning technique. The memory high power amplifier is modeled using wiener model with true output saturation characteristics. Then the model optimization is done using flower pollination algorithm ([3] and [6]) which is developed based on the flower pollination process of flowering plants. From the results obtained from model optimization an algebraic predistorter [4] is implemented. Lastly the performance of designed predistorter is evaluated by stimulating models in MATLAB. Fig.1 represents the block diagram of work flow.

II. POWER AMPLIFIER MODEL

There are various models ([2],[5] and [9]) for modeling power amplifier. In this work, power amplifier is modeled by wiener model which has linear filter followed by memoryless nonlinearity. Memory effect of power amplifier is represented by linear filter with memory length of three and travelling wave tube nonlinearity is used to represent the nonlinearity of power amplifier. The linear filter coefficients have been represented as $h^T=[h_0\ h_1\ h_3]$ and nonlinearity of power amplifier is given as $t^T=[\alpha_a\ \beta_a\ \alpha_\phi\ \beta_\phi]$. Eq. (1) and (2) represent amplitude and phase distortion caused by memoryless nonlinearity.

$$A(r) = \begin{cases} \frac{\alpha_a r}{1+\beta_a r^2}, & 0 \le r \le r_{sat} \\ A_{max}, & r > r_{sat} \end{cases} \quad (1)$$

$$\phi(r) = \frac{\alpha_\phi r^2}{1+\beta_\phi r^2} \quad (2)$$

In above equations r is the amplitude of input QAM signal, r_{sat} is the saturating input and A_{max} is the saturation amplitude of output. The expression for saturation input and output saturation is given as

$$r_{sat} = \frac{1}{\sqrt{\beta_a}} \quad (3)$$

$$A_{max} = \frac{\alpha_a}{2\sqrt{\beta_a}} \quad (4)$$

The wiener power amplifier model is implemented with h^T=[0.7692 0.1538 0.0769] and nonlinearity coefficients t^T=[2.1587 1.15 4.0 2.1].Figure 2 and Figure 3 show power amplifier distortion for input back-off values of 5 dB and 10 dB respectively.

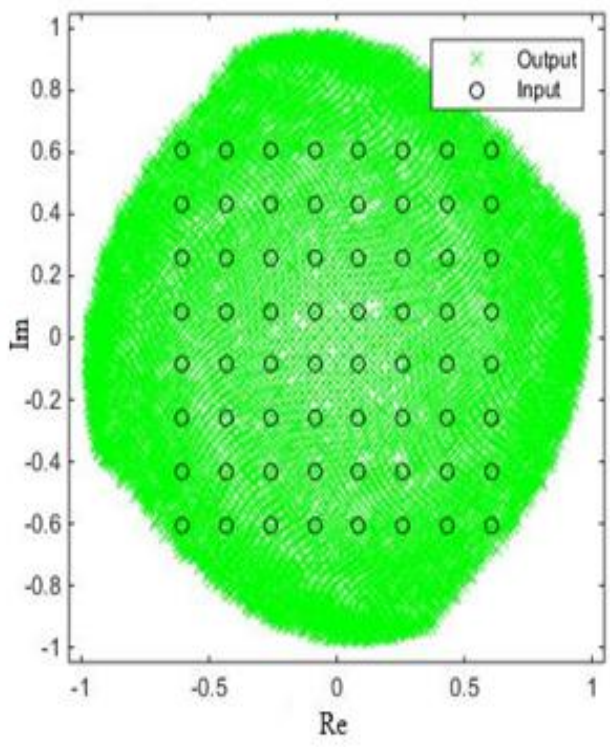

Fig. 2. Output from Wiener Model with IBO 5 dB. 'x' represents output y(n) and 'o' represents input 64-QAM signal x(n).

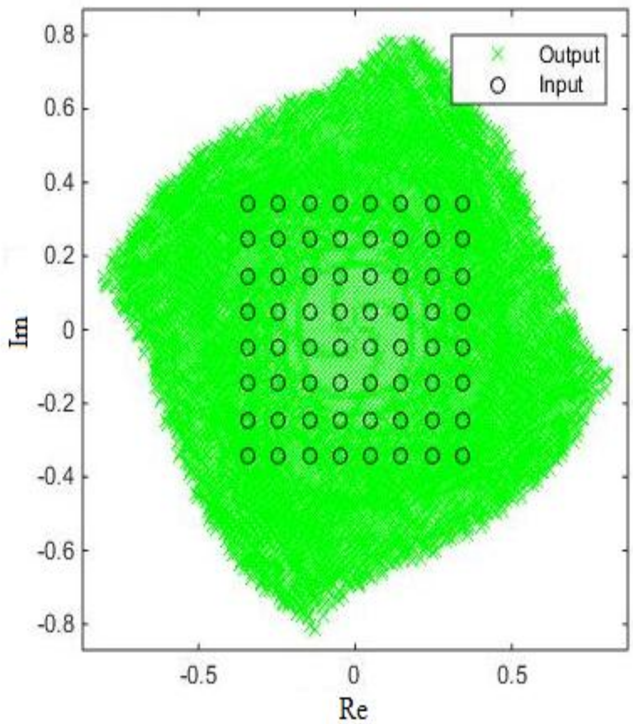

Fig. 3. Output from Wiener Model with IBO 10 dB. 'x' represents output y(n) and 'o' represents input 64-QAM signal x(n)..

III. MODEL IDENTIFICATION

For the identification purpose normalized 64-QAM signal is generated and applied to power amplifier model to create training data set {x(n),y(n)}, where x(n) is the input QAM and y(n) is output from the model. The true parameter of memory high power amplifier is estimated using training data. The true parameter vector is defined as $\theta=[\theta_1\ \theta_2\ \dots\ \theta_{N_\theta}]^T$, where N_θ represents the total number of parameter to be estimated. The training data input x(n) is applied to the model and measured output y(n) is corrupted by the small noise. The output from the estimated wiener model is denoted by $\hat{y}(n)$. Consider the error between the desired output y(n) and the model output $\hat{y}(n)$ as $e(n)=y(n)-\hat{y}(n)$, thus mean square error cost function can be given by

$$J(\tilde{\theta}) = \frac{1}{K}\sum_{k=1}^{K} |e(k)|^2 \quad (5)$$

The true parameter vector θ is estimated by defining the solution of the following optimization problem

$$\hat{\theta} = \arg\min_{\tilde{\theta}\in\Theta} J(\tilde{\theta}) \quad (6)$$

where Θ is the search space for parameter vector. The true parameter θ is an element of the search space. The cost function Eq. (5) is a nonlinear function and has local minima. The above challenging identification problem is solved using

Flower Pollination algorithm (FPA) ([3] and [6]). The main characteristics of this algorithm can be summarized as

- Biotic and cross-pollination is considered as global pollination process with pollen carrying pollinators performing Levy flights.
- Abiotic and self-pollination are considered as local pollination.
- Flower constancy can be considered as the reproduction probability is proportional to the similarity of two flowers involved.
- Local pollination and global pollination is controlled by a switch probability p ε [0, 1]. Due to the physical proximity and other factors such as wind, local pollination can have a significant fraction p in the overall pollination activities.

For solving the optimization (5) using FPA algorithm population of flower or pollen gametes (N), number of iteration (Iter), search space (Θ) and switching probability (p) has to be determined in order to find a solution $\hat{\theta}$. In order to get better solution search space should be small enough. The updating equations for global and local pollination are given in Eq. (7) and (8) respectively.

$$\theta_i^{t+1} = \theta_i^t + L * (\theta_i^t - g_*) \quad (7)$$

$$\theta_i^{t+1} = \theta_i^t + \varepsilon * (\theta_j^t - \theta_k^t) \quad (8)$$

where as θ_i^t represents the solution vector at iteration t, g_* is the current best solution obtained among all solutions at the current iteration, θ_j^t and θ_k^t are selected from same population if ε is uniformly distributed in [0 1] and L is the step size drawn from the Levy distribution. The expression for L is given as

$$L = 0.01 * \frac{\mathrm{randn}(1, N_\theta) * \sigma}{\mathrm{abs}(\mathrm{randn}(1, N_\theta))^{\frac{1}{\beta}}} \quad (9)$$

where randn(1,N_θ) gives vector of size N_θ with elements are normally distributed pseudorandom number, β=1.5 [3] and

$$\sigma = \frac{\Gamma(1+\beta) * \sin\frac{\pi\beta}{2}}{\Gamma\left(\frac{1+\beta}{2}\right) * \beta * 2^{\left(\frac{\beta-1}{2}\right)}}^{1/\beta} \quad (10)$$

Γ represents the gamma function. The detailed pseudo code for FPA can be given as

- FPA Initialization

Specify the population size (N), number of iteration (iter), switching probability (p), upper bound and lower bound limits.
Define the cost function.
Randomly initialize the initial solutions.
Find the current best solution (g_*).

- FPA Generation

```
while(t<iter)
        for (i=1:N)
          if rand<p
                 Calculate L using (9) and update the
                 solution using (7)
            else
              Draw ε from [0,1].
              Randomly select two solutions j and k.
                 Update the solution using (8)
        end if
      Calculate the cost function for new solutions
      If new solutions produce better result, then update
      them in population
     end for
Find the g*
 end while
```

- FPA Termination

The best solution is g_* after iteration.

IV. PREDISTORTER DESIGN

Hammerstein model [4] (memoryless nonlinearity followed by linear filter) is used as the predistorter, as it represents the inverse form of the wiener model. More precisely, linear filter of predistorter is made in such a way that it invert linear filter of identified wiener power amplifier model and inverse nonlinearity of estimated wiener is implemented for the nonlinearity of predistorter. Consider the transfer function of the Hammerstein Predistorter's linear filter is to be

$$G(z) = z^{-\tau} \sum_{i=0}^{N_h} g_i z^{-i} \quad (11)$$

where g_i represents the linear filter coefficient and τ is the delay. If H(z) is the transfer function of linear filter of wiener model and is a minimum phase filter then τ=0. The filter coefficient of predistorter can be obtained by solving the linear equations derived from the

The memoryless nonlinearity of the predistorter should compensate the amplitude and phase predistortion caused by the power amplifier nonlinearity, as described in Eq. (1) and (2). Let r(n) denotes the amplitude of the input signal x(n). Consider the amplitude gain function of the predistorter's nonlinearity is to be P(r), which means that the amplitude predistortion function of this memoryless nonlinearity is the product of input amplitude (r) and the amplitude gain function of predistorter, and the corresponding phase predistortion function by Ω(r). Noting (1), the required correction equation for the amplitude predistortion function to meet is

$$A(r.P(r)) = r, \quad \text{for } r.P(r) \leq r_{sat} \quad (13)$$

.

Expanding (13) by using Eq. (1) and solving it gives two solutions, and the smaller solution is taken as the required amplitude gain function, also when r>A_{max} it is impossible to obtain A(r.P(r))=r, so for this case P(r) is set to one. Thus, appropriate gain function is

$$P(r) = \begin{cases} \frac{\alpha_a - \sqrt{\alpha_a{}^2 - 4\beta_a r^2}}{2\beta_a r^2}, & r \leq A_{max} \\ 1 & r > A_{max} \end{cases} \quad (14)$$

Next the correction equation for the predistorter phase function is obtained from Eq. (2), which is given as

The expression for the predistorter phase distortion is

$$\Omega(r) = -\phi(r.P(r)) = -\frac{\alpha_\emptyset (r.P(r))^2}{1 + \beta_\emptyset (r.P(r))^2} \quad (16)$$

V. SIMULATION RESULTS

The 64 QAM signal is used as input data and the wiener power amplifier model was implemented with linear filter coefficients h^T=[0.7692 0.1538 0.0769] and nonlinearity coefficients t^T=[2.1587 1.15 4.0 2.1]. All the work in this project is done by using MATLAB.

5.1. Identification Results

The training data set consists of 500 samples of normalized 64 QAM data. In FPA population size N=20, p=0.8 and number of iteration iter=100 are used. Also noise with standard deviation 0.0 and 0.01 was added at output during identification with IBO=5, 10, 15db. The results were averaged over 100 runs. The parameter vector for the estimated wiener power amplifier obtained by FPA is h^T=[0.7669 0.1537 0.0765] and t^T=[2.1667 1.1945 4.004 2.05].

5.2. Performance of Predistorter

Predistorter was implemented by using estimated wiener power amplifier model. The length of the linear filter hammerstein predistorter was set to 8. The performance of the designed predistorters were evaluated using following mean square error metric given below,

$$MSE = 10\log_{10}\left(\frac{1}{K_{test}}\sum_{k=1}^{K_{test}} |x(n) - (n)|^2\right) \quad (17)$$

where K_{test} represents the total number of the test data, x(k) was the input signal and y(k) was the output of the combined predistorter and memory high power amplifier system.

For evaluating the performance of predistorter K_{test} =2×10^5 samples of 64 QAM data were passed through the combined predistorter and wiener power amplifier. The mean square error metric (MSE) was computed for the both predistorters by observing input and output data. The resulting MSE for the both predistorter is plotted as a function of IBO, which is shown in Fig. 4.

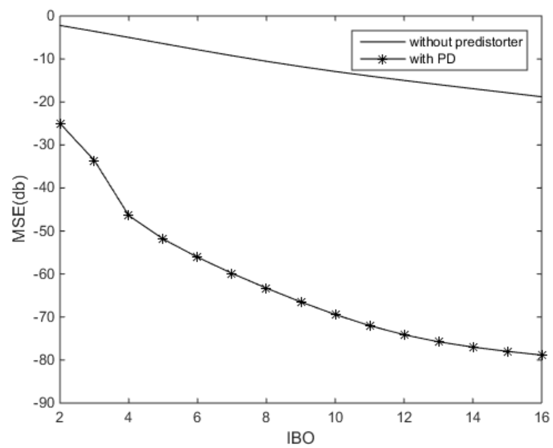

Fig. 4. MSE Vs IBO performance of designed predistorter, where wiener power amplifier is using h^T=[0.7692 0.1538 0.0769] and t^T=[2.1587 1.15 4.0 2.1].

The constellation diagram of output signal after the combined predistorter and wiener power amplifier model is shown Fig. 5 for the IBO=5db. It can be seen that the predistorters almost completely cancel out the nonlinearity caused by power amplifier model.

For evaluating bit error rate generated 64 QAM signal is applied to power amplifier model with and without predistorter and then output from power model is transmitted through an AWGN channel and at receiver BER is determined. Fig. 6. shows the obtained BER plot.

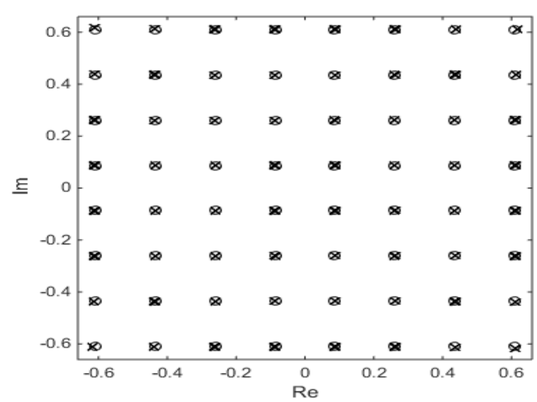

Fig. 5. Output after combined Predistorter and Wiener Model with IBO=5 dB. 'x' represents output y(n) and 'o' represents input 64-QAM signal x(n).

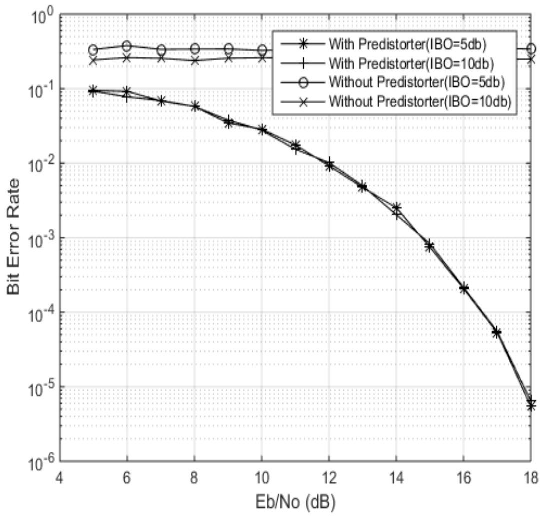

Fig. 6. BER Vs SNR performance

VI. CONCLUSION

Predistorter is designed to compensate the distortions caused by wiener memory power amplifier which exhibit true output saturation characteristics. For identification purpose FPA algorithm has been employed to estimate an accurate memory power amplifier model, based on which predistorter solution can be directly obtained. The effectiveness of the predistorters designed has been illustrated by simulation results. In particular, it has been shown that this novel digital PD is capable of successfully compensating serious nonlinear distortions caused by the PA operating into the output saturation region.

REFERENCES

[1] Biyi Lin, Yide Wang Bruno Feuvrie and Qingyun Dai , "A Novel Power Amplifier Linearization Technique based on Pre-distortion Principle and Wiener model," IJAST Vol. 22, September, 2010.

[2] Bipin P R,"Artificial bee colony for pre-distortion of nonlinear power amplifiers," IJEIR 2013.

[3] Xin-She Yang, "Flower Pollination Algorithm for Global optimization," in Unconvectional Computation and Natural Computation 2012, Lecture Notes in Computer Science, Vol. 7445, pp. 240-249 (2012).

[4] Y. Wang, L. Xie, Z. Wang, H. Chen, and K. Wang, "An efficient algebraic predistorter for compensating the nonlinearity of memory amplifiers," in *Proc. 2010 IEEE Int. Conf. Communications, Circuits and Systems*, Chengdu, China, Jul. 28–30, 2010, pp. 800–804.

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

[6] Xin-She Yang, Mehmet Karamanoglu, Xingshi He, "Multi-objective Flower Algorithm for Optimization", ICCS 2013, Elsevier.

[7] D. Zhou and V. E. DeBrunner, "Novel adaptive nonlinear predistorters based on the direct learning algorithm," IEEE Trans. Signal Processing, vol. 55, no. 1, pp. 120–133, Jan. 2007.

[8] A. A. M. Saleh, "Frequency-independent and frequency-dependent nonlinear models of TWT amplifiers," *IEEE Trans. Communications*, vol. COM-29, no. 11, pp. 1715–1720, Nov. 1981.

[9] Joel Goodman, Matthew Herman, Bradley Bond and Benjamin Miller," A Log-Frequency Approach to the Identification of the Wiener–Hammerstein Model" IEEE signal processing letters, vol. 16, no. 10, october 2009.