Reduction of Power Envelope Fluctuations in OFDM Signals by using Neural Networks

Younes Jabrane, Víctor P. Gil Jiménez, *Member, IEEE,* Ana García Armada, *Senior Member, IEEE,* Brahim Ait Es Said, and Abdellah Ait Ouahman

***Abstract*—One of the main drawbacks of Orthogonal Frequency Division Multiplexing (OFDM) are the large fluctuations of its power envelope. In this letter, a novel and efficient scheme based on Multilayer Perceptron (MLP) Neural Networks (NN) is proposed. The NN synthesizes the Active Constellation Expansion - (ACE) technique which is able to drastically reduce envelope fluctuations. This is achieved with much lower complexity, faster convergence, and better performance compared to previously available methods.**

***Index Terms*—OFDM, PAPR, cubic metric, neural networks.**

I. Introduction

ORTHOGONAL Frequency Division Multiplexing (OFDM) is a spectrally efficient technique which is able to provide high data rates over multipath fading channels. However, its large envelope fluctuations cause a loss in energy efficiency due to the need of power back-off for the High Power Amplifiers (HPA). Some examples of previous work dealing with this problem are [1]–[5].Within these techniques, Active Constellation Expansion (ACE) [5] has aroused great interest since it provides large envelope reductions without the need for side information, so that the data rate is not compromised. Its only weakness is its slow convergence that requires a long processing time to obtain the final signal.

On the other hand, Artificial Neural Networks (ANNs) are being successfully applied across a broad range of problem domains. In fact, in [6] and [7] ANN are already proposed for envelope reductions in OFDM, implementing the Tone Injection (TI) [3] and Selective Mapping (SLM) [1] algorithms, respectively.

In this letter, a novel proposal based on ANN and ACE is presented. The idea is to train the ANN to synthesize the ACE algorithm obtaining its good results but with much lower complexity and faster convergence.

II. System Model

The classical metric used to measure power fluctuations in an OFDM signal $(\mathbf{x})$ with N sub-carriers is the PAPR (Peak to Average Power Ratio). However, as revealed recently, it does not appropriately take into account the complete distortion effect due to the non linear response of the HPA. For this reason, other metrics have been proposed such as the Cubic Metric (CM) [8], that uses higher statistics to evaluate the power de-rating factor in an HPA. It is defined by the Third Generation Partnership Project (3GPP) as

$$CM = \frac{RCM\ (dB) - RCM_{ref}\ (dB)}{K}\ dB \tag{1}$$

where RCM is the Raw Cubic Metric, that is defined for a signal $\mathbf{x}$ as

$$RCM = 20\log_{10}\left(\sqrt{E\left\{\left(\frac{|\mathbf{x}|}{\sqrt{E\left\{(\mathbf{x})\right\}}}\right)^3\right\}}\right)\ dB\ , \tag{2}$$

RCM_{ref} is the reference RCM which for OFDM takes the value 1.52 dB, and K is 1.56. This CM will be used in this paper.

The ACE method modifies and expands the constellation points within an allowable region which does not affect the demodulation slicer, and thus, it does not need side information. By using these new degrees of freedom, multi-carrier signals with arbitrarily low power peaks relative to the mean can be obtained. In [5] different algorithms to achieve PAPR reduction through ACE are provided. In this paper, the Approximate Gradient-Project (AGP) will be used since the utility of ACE is to obtain the training samples, and therefore ensuring the convergence is more important than its speed.

III. Artificial Neural Network (ANN)

The ANN consists of many non-linear computational elements operating in parallel and arranged in patterns resembling those of biological neurons. The neurons are interconnected into networks via weighted connections, which are usually adapted through a learning process in order to obtain a certain performance. Neurons are grouped into layers depending on their degree of connection to the outside world. The neurons are usually characterized by an internal threshold and by the type of their activation function. The neurons in the input layer are fed with input data. Each neuron sums all the inputs and transfers the data, according to an activation function, to all the elements in the next layer. However, each neuron receives a different signal due to the different weights associated to the connections between the neurons. The output of each neuron in the output layer is compared to the desired output. The difference between the desired output and the obtained one constitutes an error that is fed back to the network so that the weights are adjusted in such a way that the error is minimized. After repeating this training procedure several times, the network is *trained* and it can be used with non previously shown input data (denoted as test data).

Among the different ANN types and architectures [9], the multilayer feedforward network is an important class. It consists of a set of sensory units that constitute the input layer, one or more hidden layers of neurons, and an output

This work has been partly funded by the projects MULTI-ADAPTIVE (TEC2008-06327-C03-02), COMONSENS (CSD2008-00010), and the AECI Program of Research Cooperation with Morocco.

Y. Jabrane, B. es Said, and A. A. Ouhman are with the Team of Telecommunications and Informatics Networks, University Cadi Ayyad.

V. P. Gil Jiménez and A. García Armada are with the Department of Signal Theory and Communications, University Carlos III de Madrid (e-mail: vgil@tsc.uc3m.es).

layer. The input signal propagates through the network in a forward direction on a layer-by-layer basis. These ANNs are commonly referred to as Multi Layer Perceptrons (MLPs) and they can be trained with different learning algorithms [10]. MLPs with the Levenberg-Marquardt algorithm [11] have been selected in this letter to obtain signals with low envelope fluctuations.

IV. Time and Time-Frequency NN Schemes

A. Time-domain Neural Network

Our first proposal to reduce its envelope fluctuations is based on the time-domain OFDM signal. We train our ANN by using the signals with low envelope fluctuations obtained by the ACE-AGP algorithm. This way, this ANN learns what are the characteristics for a signal with low envelope fluctuations. We first decompose into real and imaginary parts the time-domain original signal to reduce the complexity. The training process is detailed here:

1) Use the original time-domain data $\mathbf{x}$ as an input to the ACE-AGP algorithm to obtain $\mathbf{x}^{AGP}$, *i.e.*, a signal with reduced envelope fluctuations.
2) Split $\mathbf{x}$ and $\mathbf{x}^{AGP}$ into two sets, namely, the training set $\mathbf{x}^{tr}$, $\mathbf{x}^{AGP,tr}$ and the test set, $\mathbf{x}^{ts}$, $\mathbf{x}^{AGP,ts}$.
3) Decompose into real and imaginary parts the original data $\mathbf{x}^{tr}$ ($\mathbf{x}^{tr}_{Re}$, $\mathbf{x}^{tr}_{Im}$), and ACE-AGP output $\mathbf{x}^{AGP,tr}$ ($\mathbf{x}^{AGP,tr}_{Re}$, $\mathbf{x}^{AGP,tr}_{Im}$).
4) Obtain $\mathbf{x}^{TNN}_{Re}$ and $\mathbf{x}^{TNN}_{Im}$ by training the two models Mod^{TNN}_{Re} and Mod^{TNN}_{Im} with the pairs [$\mathbf{x}^{tr}_{Re}$, $\mathbf{x}^{AGP,tr}_{Re}$] and [$\mathbf{x}^{tr}_{Im}$, $\mathbf{x}^{AGP,tr}_{Im}$].
5) Test with the values of $\mathbf{x}^{\mathbf{ts}}$ to validate the models Mod^{TNN}_{Re} and Mod^{TNN}_{Im}.

B. Time-Frequency-domain Artificial Neural Network

The main problem with the time-domain training scheme is that the ANN is not able to learn which regions in the constellation are allowed and which ones are forbidden (see Fig. 2). Thus, a second ANN working on the frequency-domain is proposed to be concatenated to the time-domain scheme.

1) Apply DFT on $\mathbf{x}^{TNN}_{Re}$ and $\mathbf{x}^{TNN}_{Im}$ to obtain the frequency-domain signal $\mathbf{X}^{TNN}$.
2) Split in training samples $\mathbf{X}^{TNN,tr}$ and test samples $\mathbf{X}^{TNN,ts}$.
3) Separate the training samples $\mathbf{X}^{TNN,tr}$ in the four constellation regions in order to train eight NNs. We will divide the signal in two sets: 1st set concerning real parts and the 2nd set concerning the imaginary parts, as it can be seen in Fig. 1.
4) Train the first set of NNs by $\Re e(\mathbf{X}^{TNN,tr})$ to generate $Mod^{TFNN}_{Re,1q}$, $Mod^{TFNN}_{Re,2q}$, $Mod^{TFNN}_{Re,3q}$ and $Mod^{TFNN}_{Re,4q}$ for each quadrant.
5) Train the second set of NNs by $\Im m(\mathbf{X}^{TNN,tr})$ to generate $Mod^{TFNN}_{Im,1q}$, $Mod^{TFNN}_{Im,2q}$, $Mod^{TFNN}_{Im,3q}$ and $Mod^{TFNN}_{Im,4q}$ for each quadrant.
6) Test with the values of $\mathbf{X}^{TNN,ts}$ to validate the models.

C. Complexity Analysis

Once the system has been off-line trained, the operating implementation of the proposed schemes is as simple as introducing the input signal into the generated system model, which performs basically integer multiplications and additions, to obtain the desired signal. The designed NN are simple with two layers and two neurons per layer with triangular activation function. Its complexity thus, in terms of number of integer multiplications, is $16 \times N$, while the number of integer additions is $12 \times N$. In the Table I, a complexity comparison for different schemes and algorithms is summarized. For completeness, the PTS and SLM algorithms have also been included, where U and M are the number of blocks and sequences, respectively. Several conclusions can be extracted from the table. The first one is that our proposals are much less complex than the other schemes, especially because they do not need several (I)FFT operations (the scheme only needs one). Besides, since our proposal use integer operations, the operations are even simpler[1] than the other schemes (only complex operations are needed for the DFT). Indeed, in a realistic implementation, the operations can be parallelized and it is possible to perform in parallel the real and imaginary part for each quadrant. The difference in complexity is especially relevant when comparing against ACE-SGP.

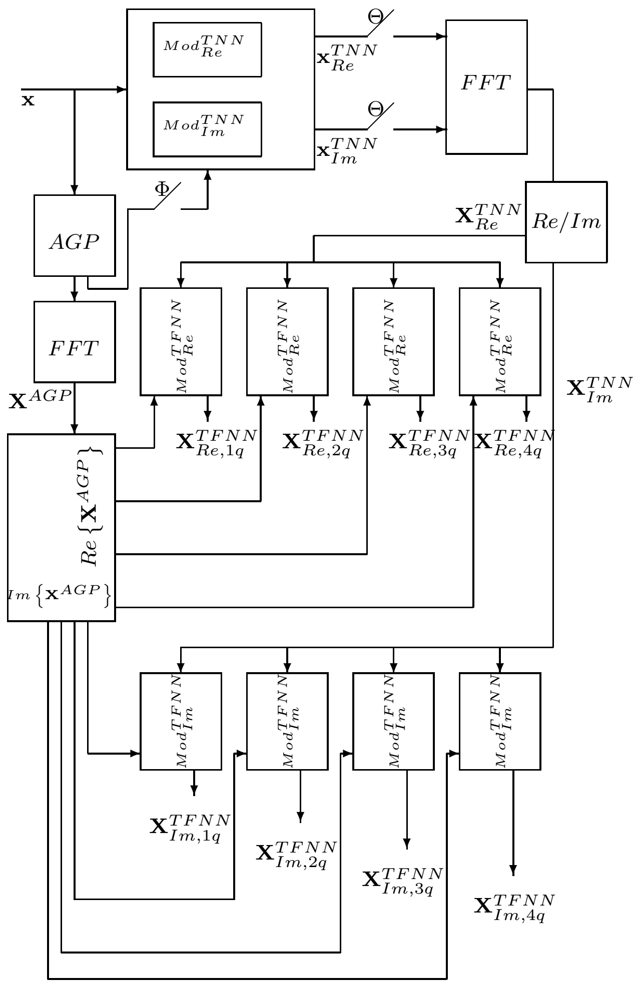
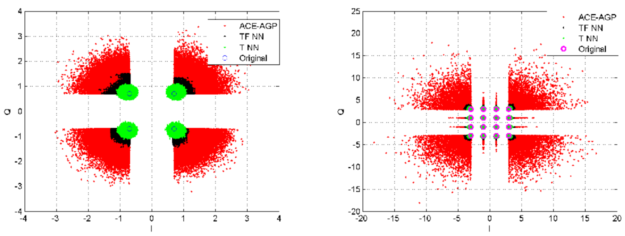


Fig. 1. For Time-domain ANN training: Switch Φ *on* and Θ *off*. For Frequency-domain ANN training: Switch Φ *off* and Θ *on*.

[1]Typically, integer additions are half complex than complex additions whereas integer multiplications are four times simpler than complex multiplications.

TABLE I
COMPLEXITY SUMMARY AND COMPARISON. N = 1024. SLM (U = 16), PTS (U = 8, M = 64) AND ACE-SGP ($N_{iter} = 50$)

	SLM	PTS	ACE-SGP	NN
(I)FFT	16	64	100	1
Complex Mult	98304	40960	204800	5120
Complex Adds	163840	1003520	409600	10240
Integer Mult.	-	-	-	16384
Integer Adds	-	-	-	12288
Check operations	16	64	50	-



(a) QPSK (b) 16QAM

Fig. 2. Constellations for different schemes.

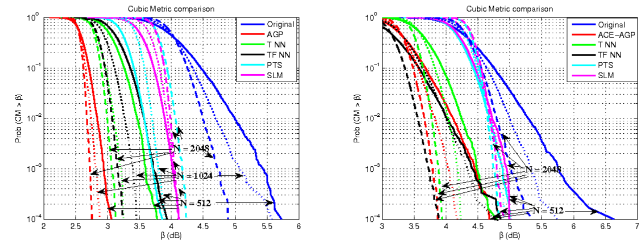
V. RESULTS

The following results have been obtained by using Monte Carlo simulations with $50,000$ randomly generated QPSK and 16-QAM modulated OFDM symbols for $N = 512, 1024$ and 2048 sub-carriers. The training experiments in Time-domain and Time-Frequency-domain have been carried out with 70 $\%$ data for training and 30 $\%$ data for testing. For the ACE-AGP, the maximum number of iterations was fixed to be $2,000$ ($i.e.$, a large number of iterations to guarantee its best performance), whereas for the PTS [2] and SLM [1], the number of phases and blocks was the typical $2(1, -1)$ and 16, respectively. In Fig. 3, we can observe that the loss in performance of the proposed TF ANN scheme with respect to the Time-domain ANN system is below 0.2 dB in terms of CM. Both proposed ANN schemes outperform the PTS and SLM algorithms, and moreover, without the complexity and convergence time of PTS or SLM. Regarding the performance loss in terms of CM, it is less than 0.3 dB with respect to the ACE-AGP for $N = 512$ sub-carriers and 0.9 dB for $N = 2048$. Similar results are obtained when applying to 16-QAM. Besides, the scheme works better for larger number of sub-carriers, which is interesting since a larger number of sub-carriers implies a larger probability of experiencing more accentuated envelope fluctuations.

In Fig. 4, the BER for the proposed ANN systems, the original signal and the ACE-AGP are presented. It can be seen that the TF ANN outperforms the other schemes and it is close to the original system. The reason for this behavior is that the proposed TF ANN concentrates the energy (average constellation energy 1.17 for QPSK) even more than ACE-AGP (1.2) - see Fig. 2 - and besides, most of the constellation points are moved from their original position. Thus, only a few symbols will experience an effective lower SNR.

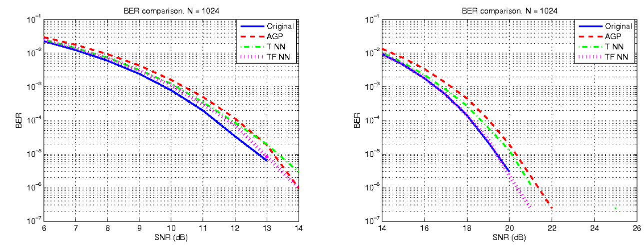
VI. CONCLUSION

In this letter a novel algorithm for reducing the envelope fluctuations in OFDM signals has been described. Simulation results show that, once trained, the proposed scheme based on ANNs, is able to directly obtain signals with low envelope fluctuations at a very low complexity. This scheme, that is valid for any number of sub-carriers, achieves a better performance than previous work. Interestingly, the obtained improvement of the envelope behavior is better for large number of sub-carriers.



(a) QPSK (b) 16-QAM

Fig. 3. Cubic Metric comparison. $N = 512$ (solid), $N = 1024$ (dotted) and $N = 2048$ (dashed). Oversampling factor is 4.



(a) QPSK (b) 16-QAM

Fig. 4. BER comparison for different schemes.

REFERENCES

[1] M. Breiling, S. H. Muller-Weinfurtner, and J. B. Huber, "SLM peak-power reduction without explicit side information," *IEEE Commun. Lett.*, vol. 5, no. 6, pp. 239–241, June 2001.
[2] J.-C. Chen, "Partial transmit sequences for peak-to-average power ratio reduction of OFDM signals with the cross-entropy method," *IEEE Signal Process. Lett.*, vol. 16, no. 6, pp. 545–548, June 2009.
[3] J. Tellado, "Peak-to-average power reduction," Ph.D. dissertation, Stanford University, Sep. 1999.
[4] V. P. G. Jiménez, M. J. F.-G. García, M. P. S. Fernández, and A. G. Armada, "Efficient implementation of complementary Golay sequences for PAR reduction and forward error correction in OFDM-based WLAN systems," *AEÜ - Inter. J. Electronics and Commun.*, vol. 62, no. 9, pp. 683–694, Oct. 2008.
[5] B. S. Krongold and D. L. Jones, "PAR reduction in OFDM via active constellation extention," *IEEE Trans. Broadcast.*, vol. 49, no. 3, pp. 258–268, Sep. 2003.
[6] M. Ohta, Y. Ueda, and K. Tamashita, "PAPR reduction of OFDM signal by neural networks without side information and its FPGA implementation," *Trans. Inst. Elect. Engnr. Jpn.*, vol. 126, pp. 1296–1303, Nov. 2006.
[7] I. Sohn, "RBF neural network based SLM peak-to-average power ratio reduction in OFDM systems," *ETRI J.*, vol. 29, no. 3, pp. 402–404, June 2007.
[8] TDoc R1-060023, "Cubic metric in 3GPP-LTE," 3GPP TSG RAN WG1, Tech. Rep., Jan. 2006.
[9] M. Ibnkahla, "Applications of neural networks to digital communications—a survey," *Signal Process.*, vol. 80, no. 7, pp. 1185–1215, 2000.
[10] E. Castillo, B. Guijarro-Berdiñas, O. Fontenla-Romero, and A. Alonso-Betanzos, "A very fast learning method for neural networks based on sensitivity analysis," *J. Machine Learning Research*, vol. 7, pp. 1159–1182, Sep. 2006.
[11] D. W. Marquardt, "An algorithm for least-squares estimation of nonlinear parameters," *J. Soc. Ind. Appl. Math.*, vol. 11, no. 2, pp. 431–441, 1963.