Peak Shrinking and Interpolating Technique for reducing Peak to Average Power Ratio

Somayeh Mohammady, Ronan Farrell, David Malone, and John Dooley

Department of Electronic Engineering, National University of Ireland Maynooth (NUIM), Ireland

Abstract—**This paper discusses topic of high Peak to Average Power Ratio (PAPR) in Orthogonal Frequency Division Multiplexing (OFDM) based signals. An innovative technique is proposed to reduce PAPR without compromising Error Vector Mapping (EVM). The proposed technique is named Peak Shrinking and Interpolating (PSI), and results show no more than 1% EVM when the technique is operated. The hardware resource consumption of PSI technique is analyzed and compared with the simplest Crest Factor Reduction (CFR) technique known as Clipping and Filtering (CF). This analysis, together with EVM performance, and PAPR reduction performance indicates that PSI technique can be an outstanding scheme for existing and future technologies such as Long-Term Evolution (LTE) and 5th Generation of cellular mobile communications (5G), resulting in more efficient Power Amplifier (PA) operation.**

Keywords—ACPR, EVM, OFDM, PAPR, PSI

I. INTRODUCTION

The problem of high Peak to Average Power Ratio (PAPR) occurs when the Inverse Fast Fourier Transform (IFFT) is performed in order to provide the in band multi-carrier property for signals such as Orthogonal Frequency Division Multiplexing (OFDM), which gives them robustness against multipath fading and leads to low cost receivers compared to single carrier signals, i.e. Frequency Division Multiplexing (FDM) signals used in traditional Global System for Mobile Communications (GSM) systems. Here the IFFT transformation is presented as:

$$S(t) = \sum_{k=0}^{K-1} \tilde{s}(t) e^{\frac{j2\pi kn}{K}}, n = 0, 1, \dots, K-1 \qquad (1)$$

Where K represents the length of the IFFT, $\tilde{s}(t)$ represents the frequency domain signal in base band, and $S(t)$ represents the time domain signal after the IFFT process. As it can be observed from (1), the result of a summation of random sinusoids results in two phenomena known as constructive and destructive combination of peaks and troughs. The constructive combination leads to the creation of very high peaks, and the destructive combination reduces the average of the signal, in many cases as low as zero. According to the definition of Peak to Average Power Ratio (PAPR) indicated in (2), the ratio between the peaks or "the maximum power of the signal" to the average of the signal will be high, and applying PAPR reduction techniques also known as Crest Factor Reduction (CFR) becomes necessary, where $PAPR = \sqrt{Crest\ Factor}$. For a typical OFDM signal with IFFT length of 256, the PAPR is about 12dB. This causes the Power Amplifier (PA) to operate in its nonlinear region. The *PAPR(S(t))* is calculated as follows:

$$PAPR(S(t)) = 10 \log_{10} \frac{Peak(P(t))}{Average(P(t))} \qquad (2)$$

where $Peak(P(t))$ is the maximum or peak power of signal *S(t)*, and $Average(P(t))$ is the average or mean power of signal *S(t)*, and they are measured by next equations:

$$Peak(P(t)) = \max(|S(t)^2|) \qquad (3)$$

$$Average(P(t)) = \frac{1}{K}(\sum_{k=0}^{K-1}|S_k(t)^2|) \qquad (4)$$

where $k=0, 1, \dots, K$-1. In this work the PSI algorithm is applied to two types of OFDM signals, $K= 1024$ and 2048, and the results are compared using PAPR, Error Vector Mapping (EVM), and computational complexity measurements.
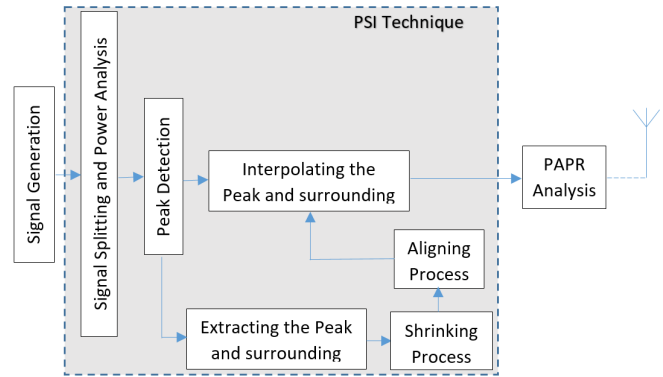

Fig. 1. Block diagram of proposed PSI technique

As shown in block diagram of the proposed PSI technique in Fig. 1, when the signal is generated, the PSI technique splits the signal into Number of Slices (*NS*) as displayed in Fig. 2:
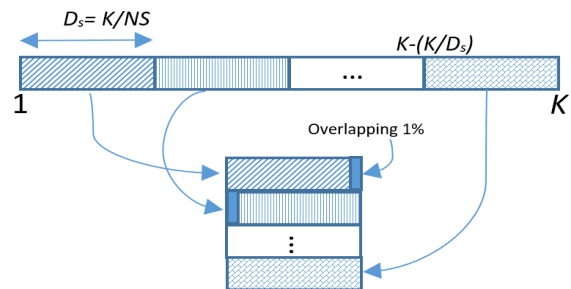

Fig. 2. Signal splitting procedure in PSI algorithm

Where Dimension of each Slice (D_s), is equal to *K/NS*. If slicing process is carried out on the signal instead of signal power, then the power of each row must be measured. It should be noted that the overlapping indicated in Fig. 2, is a useful approach in order to ensure that all peaks are detected, and 1% oversampling is suggested, however, in this work for simplicity

it is not considered, and overlapping=0%. The next process is to detect power peak for each slice, it should be noted that one slice is processed at the time and the memory is reused in order to minimize the number of bits required for implementation.

II. DETECTION OF PEAK AND ITS SURROUNDING

Peak detection can be simply implemented with maximum detection algorithm or threshold comparison algorithm, however, the recognition of the surrounding process is essential in order to ensure that border points match after the shrinking process is complete. The following algorithm is proposed to detect and extract the peak and surrounding of each peak:

```
Algorithm for defining the borders:

    SUR=2;
    P_index= index of the peak;
    L_index=Left side border;
    R_index=Right side border;

    L_index=P_index - SUR;

    if L_index ≤ 0 then;
        L_index=1;
    else
        L_index=P_index-SUR;
    end

    if R_index ≤ N then;
        R_index=P_index+SUR;
    else
        R_index=N;
    end
```

where SUrrounding Region (SUR) indicates the dimension of extraction, P_index, L_index, and R_index, show index or address of the peak, index of the border to the left, and index of the border to the right respectively. When SUR=2, it means that two data samples at the left side and two data samples at the right side of the peak are considered for extraction, therefore, in total five samples are extracted for each existing peak unless the peak is very close to the beginning of the signal or very close to the end of the signal, then the extraction will be limited to the available points. The shrinking process can be implemented as a multiplication of a single data sample point in Shrinking Ratio (SR), and the shrinking processes presented in Fig. 3, 4, and 5 are performed with SR=0.5. Table 1 presents results for SR=0.4, 0.5, and 0.6, where 0.5 performs best. It is obvious that SR values closer to 1 must result in smaller improvements to PAPR. Similarly, if we reduce SR to be close to zero then EVM performance will be degraded. However, as seen in Fig. 4, 5, and explained in next section, when borders need smoothing, SR_{new}=0.7. It should also be noted that N=$IF\times K$, and the Interpolation Factor is shown with IF, and K is the length of the signal. As mentioned before, finding the peak can be implemented by a modest power analysis and power comparison technique, however, ensuring that the surrounding points of a

peak match the rest of the data when interpolation of points is critical in order to avoid big gaps between data points.

A. Algorithm for Matching/ Smoothing out the border points:

An example peak before and after shrinking process is displayed in an example presented in Fig. 3. It can be understood that the P_index in this example is 83, assuming that SUR = 2, results the L_index=81, and R_index=85. When comparing the power different between border points, the difference between power of the point located at L_index, and the sample point located at $L\_index-1$, named D_R is about 0.001 which is less than the predefined Maximum Dimension (D_{max}) is 0.005.
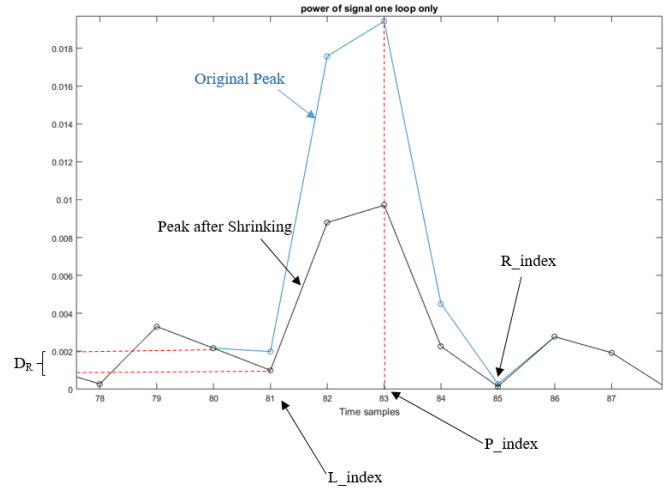

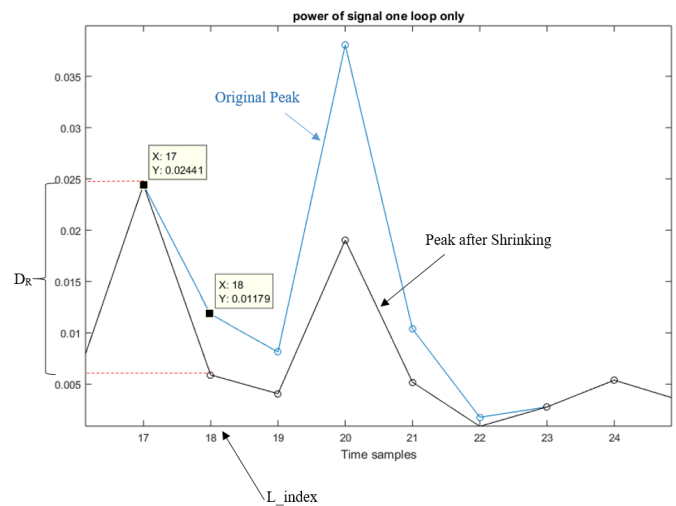Fig. 3.   Example of a peak border does not need matching/smoothing

Fig. 4.   Example of a peak border needs matching/smoothing

Hence, the smoothing process for borders of this example peak is not necessary, and it can be directly interpolated into the main signal without causing problematic distortion. The peak presented in Fig. 4 shows a case that the D_R is larger than D_{max}, and smoothing process is necessary in order to avoid distortion. As it can be seen from Fig. 4, D_R is 0.012, which is greater than

D_{max}. Therefore, the power of the signal at point L_index should be replaced with a new power point version with $\overline{SR}_{new}$=0.7.
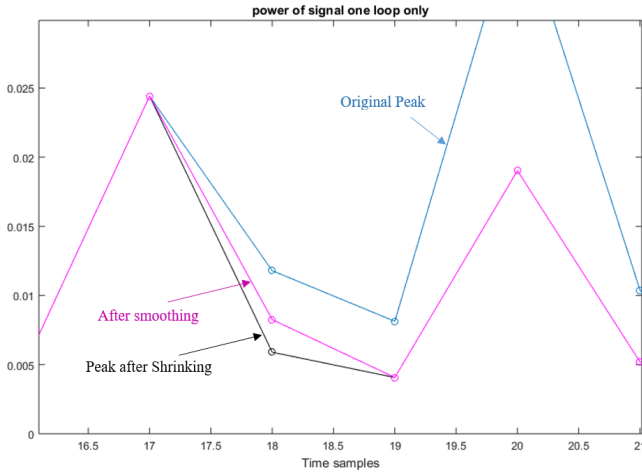

Fig. 5. Example of a peak border after matching/smoothing out

The smoothing algorithm may be operated for more multiple iterations to make sure that D_R is within the desired range of D_{max}. It should be noted that smoothing border points is only practical when both left and right indexes are within the frame of the slice, and as mentioned earlier with an overlapping slicing approach the peak detection and smoothing process can be both optimized ensuring that all high peaks are processed. As stated before, if the peak is located at the very beginning of the signal ($L\_index = 0, or\ 1$) or at the very end of signal ($R\_index = N - 1, or\ N$) smoothing borders algorithm does not function, as the comparable points are not obtainable. In order to avoid this situation, overlapped slicing is suggested.
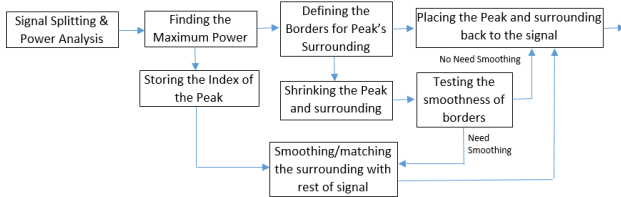

Fig. 6. The PSI iteration loop block diagram

Algorithm for matching/smoothing out the borders (after shrinking):

D= Absolute distance between power of borders' power;
D_L= Left side border point distance between peak surrounding and the rest of the signal
D_R= Right side border point distance between peak surrounding and the rest of the signal
D_{max}= Maximum value D can have=0.005 (normalized power);

if $D_L \geq D_{max}$ **then**;
    Border point is shrank with $SR/2$ instead of SR;
else
end

if $D_R \geq D_{max}$ **then**;
    Border point is shrank with $SR/2$ instead of SR;
else
end

B. Error Vector Mapping (EVM) Performance:

In order to validate the signal modification is not causing degradation on the vector, the EVM must be calculated as:

$$EVM = \sqrt{\frac{\frac{1}{K}\sum_{n=0}^{K-1}(I_{Ideal}-I_{Modified})^2+(Q_{Ideal}-Q_{Modified})^2}{|P_{max}|}} \times 100$$

(7)

Where $(I_{Ideal} - I_{Modified})^2 + (Q_{Ideal} - Q_{Modified})^2$ is the error between the modified signal ($I_{Modified}/Q_{Modified}$), and the ideal signal (I_{Ideal}/Q_{Ideal}). It should be noted that for the ideal signal, the EVM would be 0%, and according to LTE standard [2], the practical EVM for QPSK modulated signal should be kept below 17.5%, for existing technologies. Table 1 presents the EVM percentage measurement for 10^3 random OQPSK modulated signal. It is observed that for example when K=512, NS= 16, the iteration number of the PSI algorithm (RP=5), and SR=0.5, the EVM varies between EVM_{min}=0.1174%, and EVM_{max}=0.1652%, with average value of EVM_{mean}=0.1397%. These EVM results make PSI an applicable CFR technique. Various length of IFFT, different values of RP, and SR are tested, and the EVM_{max} in each case is measured and presented in Table 1. It is experienced that the maximum EVM is measured when K=512, RP=10, SR=0.5, and NS=16. It is also perceived that the EVM_{max} has the lowest value when K=2048, RP=5, SR=0.5, and increasing NS does not make any difference.

TABLE I.   THE EVM MEASUREMENT FOR PSI TECHNIQUE

Length of the IFFT (K)	EVM_{max} RP=5 SR=0.5 NS=16	EVM_{max} RP=5 SR=0.5 NS=32	EVM_{max} RP=10 SR=0.5 NS=16	EVM_{max} RP=10 SR=0.5 NS=32	EVM_{max} RP=10 SR=0.4 NS=16	EVM_{max} RP=10 SR=0.6 NS=16
512	0.1652 %	0.1652 %	0.2035 %	**0.2405 %**	0.2192 %	0.1857 %
1024	0.0756 %	0.0756 %	0.0932 %	0.1143 %	0.1053 %	0.0840 %
2048	**0.0345 %**	**0.0345 %**	0.0435 %	0.0542 %	0.0486 %	0.0381 %

III. PAPR REDUCTION PERFORMANCE OF PSI TECHNIQUE

The signals tested in Figures 3, 4, and 5 were all based on an OFDM signal with IFFT length of 1024, however here the performance is compared for different types of OFDM signal (K=512, 1024, and 2048). The simulation is performed using Matlab tools, and it is expected to see higher PAPR for K=2048 without PSI algorithm, compared to K=1024, and 512, due to the higher probability of the presence of high peaks. The common method to measure the PAPR reduction performance is through Complementary Cumulative Distribution Function (CCDF), as:

$$CCDF = Probability(PAPR(S(t)) > PAPR_{th})$$ (8)

where $PAPR_{th}$ is a predefined value, and therefore, the CCDF defines the probability of the peak value exceeding a predefined threshold. When the modulation scheme is Offset Quadrature Shift Key (OQPSK), K=2048, NS=64, RP=10, and SR=0.5. The time domain signal power, $P(t)$ with and without the PSI technique is captured and presented in Fig. 7. It can be seen that the $P(t)$ after PSI indicated with circle marked plot has lower peaks compared to unmarked plot representing the $P(t)$ without PSI technique.
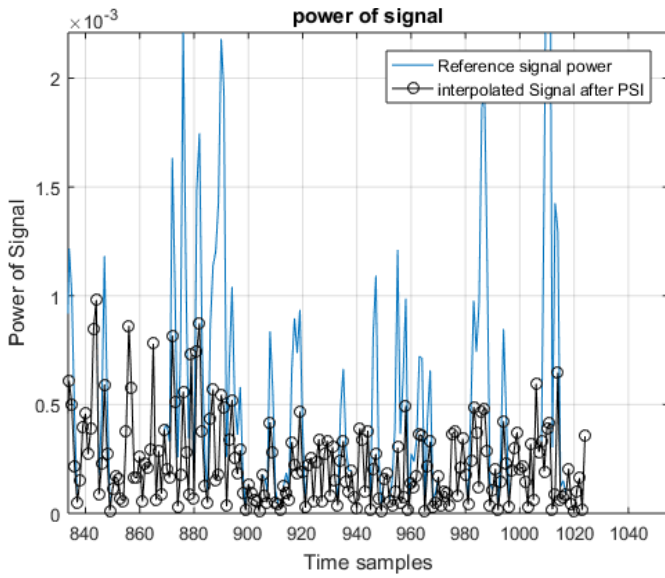
Fig. 7. Signal power before and after PSI technique

Fig. 9. Signal spectrum before and after PSI technique for K=1024.

The PAPR reduction result presented by CCDF in Fig. 8, is achieved for K=1024, and 2048, and in both cases, the PSI technique reduces the probability of PAPR by about 4.7dB at CCDF of 10^{-4}, and this aligns with IEEE standards [3]. Adjacent Channel Power Ratio (ACPR), or Adjacent Power Leakage Ratio (ACLR) is critical parameter to consider for signal modification, and as shown in spectrum of signal presented in Fig. 9, it can be seen that the main spectrum after process overlapps with original spectrum, and there is no leakage to the signal spectrum adjacent channels. This test was performed with oversampling the signal by factor of two, NS=16, SR=0.9, and RP=10.
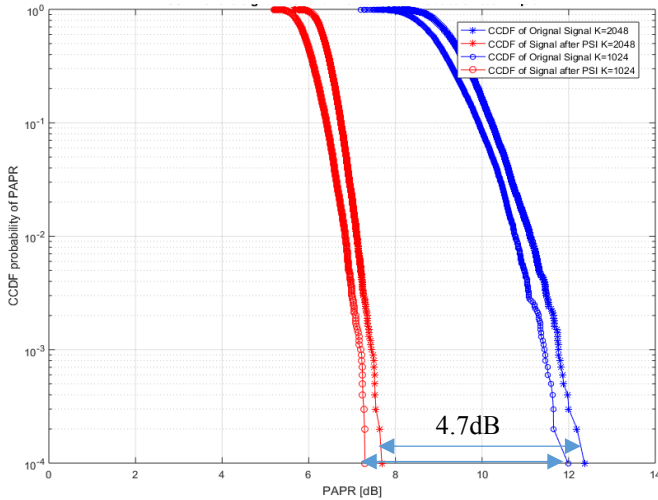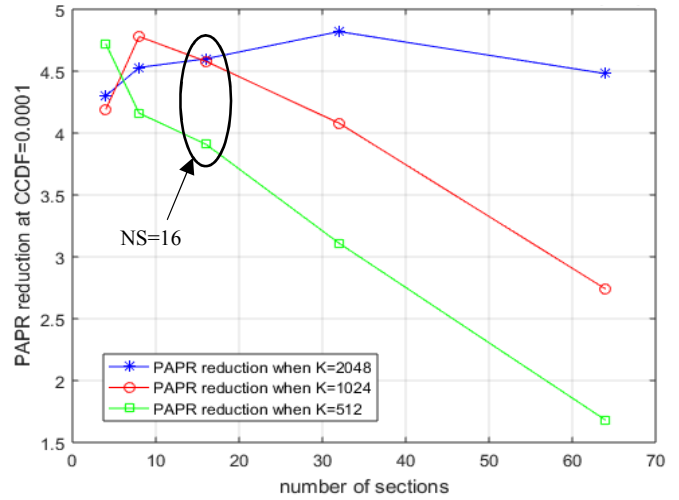

Fig. 10. PAPR reduction performance variation with NS

Fig. 10 shows how value of NS effects the PAPR reduction performance in three conditions of K=512, 1024, and 2048. It can be observed that increasing the value of NS does not necessarily improve the performance and 16 sections can be considered as an optimum value for NS, for all K. It should be noted that, higher the value of RP is employed, more effective process is achieved, but it increases the computational complexity. The PAPR reduction is measured and compared for different signals as presented in Fig. 11. It is seen that since PAPR reduction of over 4dB is acceptable for IEEE standard, not more than 10 iterations is required in all cases. It should be noted that 10 iterations is performed only on small portion of the signal and therefore, the hardware resource involved is minimal. In terms of EVM performance, as presented in Table 1, changing NS, RP, and SR has very small effect on EVM percentage. More discussions are now presented in computational complexity comparison.

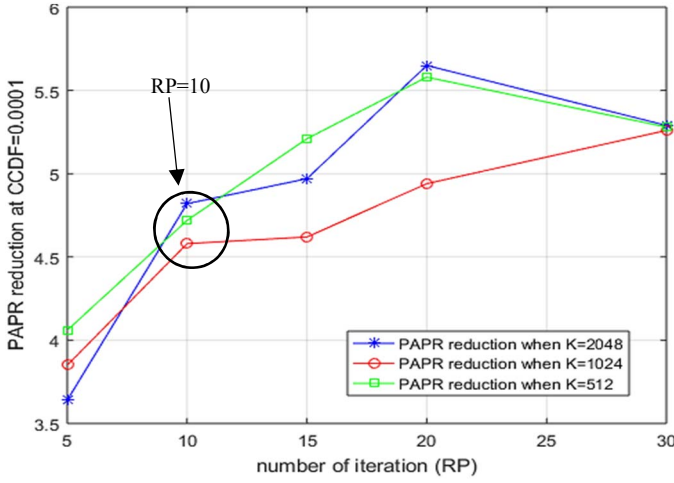Fig. 8. Signal power before and after PSI for K=1024, and K=2048.

Fig. 11. PAPR reduction performance variation with *RP*

IV. COMPUTATIONAL COMPLEXITY ANALYSIS OF PSI AND COMPARING IT WITH CF TECHNIQUE

As demonstrated in this paper, the main contribution of the proposed PSI technique is its simplicity and practicality in the case of a time domain signal without need of any additional filter and with minimal effect on EVM performance. Computational complexity is a trusted measure to analyze and compare the practicality of CFR techniques [4], and it indicates the number of additions and multiplications required for implementing an algorithm on hardware. Here, for simplicity only the number of multiplications are considered for both cases. However, it should be noted that each complex multiplication requires additions to be implemented. Furthermore, the number of bits occupied throughout the algorithm is also analyzed. In order to have a reasonable comparison, the simplest technique of Clipping and Filtering (CF) with only one loop repetition is considered as the CF technique is the most practical technique in order to reduce the PAPR [5]. A simplified CF block diagram is shown in Fig. 11, in order to explain the computational complexity involved in CF technique and comparing it with the proposed PSI technique. The main motivation for designing PSI technique is to compete with CF since it is the only practical algorithm used in GC5328 Texas Instrument Chip [7]. There many outperforming CFR techniques published, however they are not considered practical because of their complications. As presented in Fig. 11, in order to filter the noise caused by the clipping process on the signal, the signal has to be transformed in the frequency domain, and back to time domain, which requires FFT, and IFFT [5]. The number of multiplications required for CF can be determined from the following equation:

$$n_{mul,CF} = n_{mul,Clipping} + n_{mul,FFT}$$
$$+ n_{mul,Filtering} + n_{mul,IFFT} \quad (9)$$

Where $n_{mul,Clipping}$ represents the number of multiplications required for clipping which includes peak detection, defining threshold, and at least one if/else loop for each iteration.
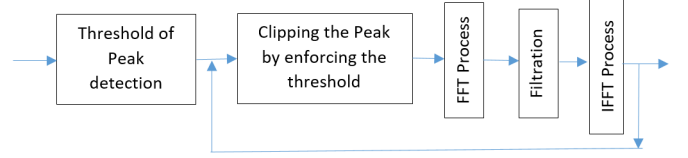

Fig. 12. Block diagram of CF technique

The $n_{mul,Filtering}$ represents the number of multiplications required for a fileting process, and the $n_{mul,FFT}$, and $n_{mul,IFFT}$ are the number of multiplications required for performing FFT and IFFT process, consequently, which can be computed as:

$$n_{mul,IFFT} = K \log k \quad (10)$$

When K=1024, $n_{mul,IFFT} = 3072$. The $n_{mul,Filtering}$ is equivalent of a convolution of two signals in frequency domain or multiplication of two signals in time domain represented as:

$$n_{mul,Filtering} = K^2 \quad (11)$$

When K=1024, $n_{mul,Filtering} = 1048576$. As stated earlier it is also useful to analyze the number of bits required for storing any data made in the process. As storing any complex number requires $\log_2 n$ bits for its absolute and $\log_2 n$ for its angle, thus, for any complex number with length of n, at least $2\log_2 n$ bits is required. The total number of bits for the CF technique is $2K\log_2 n$, due to the filter's length. When K=1024, total of 20480 bits are required. In the proposed PSI technique, as per the block diagram presented by Fig. 1, and 6, the total number of multiplications can be determined from following equation:

$$n_{mul,PSI} = n_{mul,Peak} + n_{mul,Boarders}$$
$$+ n_{mul,Extracting} + n_{mul,Shrinking} + n_{mul,Smoothing} \quad (12)$$

The $n_{mul,Peak}$ is the number of multiplications required for detecting the peak, and/or defining a threshold, which is very similar to the CF technique. The $n_{mul,Boarders}$ is the number of multiplications required for defining the borders, which includes two multiplications for calculating *L_index*, and *R_index*. Defining the borders also requires two bits of memory for storing the indexes, however it does not have to be multiplied by the number of PSI loops since they can be reused for each iteration. The $n_{mul,Extracting}$ is number of multiplications required for extracting a peak and its surrounding, and it does not involve any computation, however it requires 5 bits of memory for each peak and its surrounding. For the example shown in Fig. 2, the extracted peak can be presented as:

$$\text{Extracted Peak}$$
$$= [P(t)_{L_{index}} \; P(t)_{L_{index}+1} \; P(t)_{P_{index}} \; P(t)_{R_{index}-1} \; P(t)_{R_{index}}]$$
$$= [P(t)_{81} \; P(t)_{82} \; P(t)_{83} P(t)_{84} P(t)_{85}]$$
$$= [0.002 \; 0.018 \; 0.02 \; 0.004 \; 0] \quad (13)$$

Where *P(t)* is power of the signal *S(t)* as explained before, and it can be determined from $P(t) = |S(t)^2|$. It should be noted that the shrinking process involves the most computational complexity since it has to multiply all the five

samples by a constant, and therefore, $n_{mul,Shrinking} = 5$. This process also involves one bit for storing the value being multiplied into the signal power. For smoothing process, the computational complexity involves two if/then loops and it is equivalent to two subtractions and two multiplications. It should be noted that in general the total number of bits required for each PSI loop can be represented as:

$$n_{bit,Boarders} = 2bits \times RP \qquad (14)$$

Where RP is the number of PSI loops, and for RP=10, $n_{mul,Boarders} = 20bits$. Table 2 presents a computational complexity comparison between CF, and PSI technique, obtainable from (9) to (13).

TABLE II.   THE COMPUTATIONAL COMPLEXITY COMPARISON

The CFR Technique (K=1024)	Number of required multiplications (n_{mul})	Number of required bits (n_{bit})
Clipping and filtering (**CF**)	> 1,000,000	1,024
Peak Shrinking And Interpolation (**PSI**)	1,440	80

As stated before, outstanding complexity reduction obtained from PSI compared to CF technique is due to the elimination of filtering process, and also the slicing process, and overall design of the PSI technique allows optimum hardware usage.

V.   CONCLUSION

A novel technique is introduced here for PAPR reduction of baseband signal and its practicality and its simplicity makes it a potential approach for OFDM based signals such as those used in 4G, and 5G. Simulations show more than 4dB reduction of PAPR at CCDF of 0.0001, when length of IFFT is 2048, and the algorithm is repeated only 10 times. This improvement is achieved with less than 1% degradation in EVM.

REFERENCES

[1] S. Mohammady, R. M. Sidek, P. Varahram, M. N. Hamidon and N. Sulaiman, "FPGA implementation of the proposed DSI-SLM scheme for PAPR reduction in OFDM systems," *The 17th Asia Pacific Conference on Communications*, Sabah, 2011, pp. 484-487.

[2] LTE; Evolved Universal Terrestrial Radio Access (E-UTRA); Base Station (BS) radio transmission and reception (3GPP TS 36.104 version 13.7.0 Release 13), page 49.

[3] P. Varahram, S. Mohammady, B. M. Ali, N. Sulaiman, Power Efficiency in Broadband Wireless Communications, CRC Press, 2014, pp. 97-100.

[4] J. Hou, J. Ge and J. Li, "Peak-to-Average Power Ratio Reduction of OFDM Signals Using PTS Scheme With Low Computational Complexity," in IEEE Transactions on Broadcasting, vol. 57, no. 1, pp. 143-148, March 2011.

[5] Luqing Wang and C. Tellambura, "A simplified clipping and filtering technique for PAR reduction in OFDM systems," in IEEE Signal Processing Letters, vol. 12, no. 6, pp. 453-456, June 2005.

[6] Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, "Electron spectroscopy studies on magneto-optical media and plastic substrate interface," IEEE Transl. J. Magn. Japan, vol. 2, pp. 740–741, August 1987 [Digests 9th Annual Conf. Magnetics Japan, p. 301, 1982].

[7] Texas Instruments, "Low-Power Wideband Digital Predistortion Transmit Processor" GC5328 datasheet, October 2009