CHALMERS

OFDM Peak-to-Average-Power-Ratio Reduction in WiMAX Systems

AUTHOR: SEYRAN KHADEMI

EXAMINER: MATS VIBERG

SUPERVISORS: THOMAS ERIKSSON, THOMAS SVANTESSON

Communication Systems and Information Theory Group
Department of Signals and Systems
Chalmers University of Technology
Goteborg, Sweden, 2010
Report Number: EX036/2010

Acknowledgement

This master thesis is defined by ArrayComm company in U.S, California, San Jose. The project is a collaboration between the company and Chalmers.

I would like to show my gratitude to certain people:

Dr. Thomas Svantesson from ArrayComm
-for giving me this opportunity and also your great guidance and help during the project.

Professor Mats Viberg
-for introducing me to the right people, This work would not have been possible without your support and knowledge.

Professor Thomas Eriksson
-for kind contribution in this thesis and sharing excellent experience, your clarifying comments lead me to have more realistic perception of the problem.

Seyran Khademi
Goteborg, August 2010

Contents

Chapter 1

Introduction

Wireless communications is , by many measure, the fastest growing segment of the communications industry and the broadband wireless at the confluence of the most remarkable growth stories of telecommunication industry in recent years. WiMAX (Worldwide Interoperability for Microwave Access) is an emerging broadband wireless technology based on the IEEE 802.16 standard. The core 802.16 specification is a standard for broadband wireless access systems operating at radio frequencies between 10 GHz and 66 GHz. Data rates of around 40 Mbps will be available for fixed users and 15 Mbps for mobile users, with a range of several kilometers. Many laptop manufacturers incorporate WiMAX in their products to enable broadband internet connectivity. WiMAX will compete with wireless LANs (Local Area Networks), 3G (3th Generation) cellular services, and possibly wire line services like cable and DSL (Digital Subscriber Line). The ability of WiMAX to challenge or supplant these systems will depend on its relative performance and cost, which remain to be seen [1].

Orthogonal frequency division multiplexing (OFDM) is a multicarrier modulation technique with bandwidth-efficient signaling schemes for use in high data rate communication systems. This technique received a lot of attention especially in the field of wireless communication because of its robustness to the multi-path fading and it has already been adopted as the standard transmission technique in the wireless LAN systems and the terrestrial digital broadcasting systems. The WiMAX physical layer (PHY) is based on OFDM modulation. The OFDM system has a high peak-to-average power ratio (PAPR) that can cause unwanted saturation in the power amplifiers, leading to in-band distortion and out-of-band radiation.

In this thesis, we introduce a novel solution to the PAPR problem in the WiMAX systems using the potential capabilities in WiMAX standard as well as studying the well-known PAPR reduction techniques. Furthermore, the complexity evaluation and comparison between the existing approaches and the new algorithm has been discussed.

1.1 What Is Peak to Average Power Ratio (PAPR)?

Peak to average power ratio is a signal property that is calculated by dividing the peak power amplitude of the waveform by the RMS value of it, a dimensionless quantity which is expressed in decibels (dB). In digital transmission when the waveform is represented as signal samples, the PAPR is defined as

$$PAPR = \frac{\max(|S[n]|^2)}{E\{|S[n]|^2\}}, \quad 0 \leq n \leq N-1 \,, \tag{1.1}$$

Where $S[n]$ represents the signal samples, $\max(|S[n]|^2)$ denotes the maximum instantaneous power and $E\{|S[n]|^2\}$ is the average power of the signal, and $E\{.\}$ is the expected value operation.[8]

1.1.1 The PAPR of a Multi-Carrier Signal

A multi-carrier signal is the sum of many independent signals modulated to sub-channels of equal bandwidth. OFDM is a special case of multi-carrier transmission, where a single data stream is transmitted over a number of lower rate subcarriers. The subcarrier frequencies are chosen to be orthogonal to each other. The main advantages of OFDM are its increased robustness against frequency selective channels and also efficient use of available bandwidth. Furthermore, the orthogonality allows for efficient modulator and demodulator implementation using the FFT (Fast Fourier Transform) algorithm on the receiver side, and inverse FFT on the sender side. In the transmitter, initially the binary input data are mapped into QAM (Quadrative Amplitude Modulation) symbols and then the IFFT block is used to modulate the symbol sequence, the N-point IFFT output is given by

$$x_k = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} X_n \, e^{j \frac{2\pi nk}{N}}, \quad 0 \le k \le N-1 \, , \tag{1.2}$$

where X_n denotes the data symbols; according to (1.1) and (1.2), the PAPR for OFDM signal x_k is

$$PAPR = \frac{\max\limits_{0 \le k \le N-1} \left(|x_k|^2 \right)}{E\{|x_k|^2\}} \, . \tag{1.3}$$

Since the OFDM symbol is the summation of N different QAM symbols, it can have high peaks when phases in (1.2) are accumulated constructively [3].

1.1.2 The CCDF of the PAPR

The cumulative distribution function (CDF) of the PAPR is one of the most frequently used performance measures for PAPR reduction techniques. The complementary CDF (CCDF) of the PAPR denotes the probability that the PAPR of a data block exceeds a given threshold and is expressed as $CCDF = 1 - CDF$. When the number of subcarriers N is large, from the central limit theorem, the real and imaginary parts of the time domain signal samples fallow Gaussian distributions, each with a mean of zero and variance of $\sigma^2 = 0.5$. Hence, the amplitude of a multi-carrier signal has Rayleigh distribution, while the power distribution is exponentially distributed with mean $2\sigma^2$, (1.4). The important thing to note is that the output amplitude and hence the power are random so the PAPR is not a deterministic quantity [4].

$$|x_k| = \sqrt{(\Re\{x_k\})^2 + (\Im\{x_k\})^2}, \quad |x_k|^2 = (\Re\{x_k\})^2 + (\Im\{x_k\})^2 \, . \tag{1.4}$$

It can be proved that the maximum possible value for the PAPR is N, which occurs when all the subcarriers add up constructively at a single point. However, for independent binary inputs, the probability of maximum peak value occurring in the order of 2^{-N}. The distribution of OFDM PAPR has been studied by many researchers, among them Nee and Wild [5] introduced a simple accurate approximation of the CCDF of large N(≥ 64).

$$F_c(N, \varepsilon_{max}) = 1 - G(N, \varepsilon_{max}) = 1 - F(N, \varepsilon_{max})^{\beta N} = 1 - (1 - exp(-\frac{\varepsilon_{max}}{2\sigma^2}))^{\beta N} \, , \tag{1.5}$$

Where ε_{max} is the peak power level and β is a pseudo approximation of the over sampling factor, which is given empirically as $\beta = 2.8$. Note that the PAPR is $\frac{\varepsilon_{max}}{2\sigma^2}$ and $F(N, \varepsilon_{max})$ is the CDF of a single Rayleigh-distributed subcarrier with parameter σ^2.
The basic idea behind this approximation is that unlike a Nyquist-sampled signal, the samples of an over sampled OFDM signal are correlated, making it difficult to derive an exact peak distribution. There are many attempts to derive more accurate distribution of PAPR, see [6] for more results on this topic.

1.2 Why We Need to Reduce the PAPR?

Non-linear devices such as high power amplifiers (HPA) and digital to analog converters (DAC) exist in almost all communication links and demand for data transmission over longer ranges. At the same time higher power efficiency of the amplifiers, require the amplifier to operate in a more non-linear region, In general, there is a trade of between linearity and efficiency.

In single-carrier modulation the signal amplitude is somehow deterministic, except for the pulse shaping filter effect, so the operating point in the amplifier can be determined accurately without destructive non-linear impairments. But in the multi-carrier systems like OFDM, the envelope of the time domain signal will change with different data symbols. Accordingly, the input power amplitude will change with a noticeable variance in specified operating point and the non-linearity effect causes distortion.
Distortion acts as noise for the receiver, and also the signal constellation rotates due to phase conversion. Moreover, the out-of-band distortion of subcarriers is the result of non-linearity impairments, which causes cross talk since the subcarriers are not orthogonal any more [3].

To estimate the distortion which is caused by non-linearity, it is desired to have a measure of the signal to show its sensitivity to non-linearity. A well known measure for the multi-carrier signals is peak to average power ratio (PAPR). The higher the PAPR, the more fluctuation in the signal amplitude, so the operating point in the amplifier needs to be set far enough from saturation point and this input back off [1] reduces the efficiency.

1.3 Existing Approaches for PAPR Reduction

As in everyday life, we must pay some costs for PAPR reduction. There are many factors that should be considered before a specific PAPR reduction technique is chosen. These factors include PAPR reduction capability, power increase in transmit signal, BER increase at the receiver, loss in data rate, computational complexity increase, and so on [7]. Next we briefly discuss each of them and corresponding known PAPR reduction methods. You can find more details about these techniques in Section 2.1.

PAPR Reduction Capability Careful attention must be paid to the fact that some techniques result in other harmful effects. For example, the *Amplitude Clipping* technique in Section 2.1.1 on Page 10 clearly removes the time domain signal peaks, but results in in-band distortion and out-of-band radiation.

Power Increase in Transmit Signal Some techniques require a power increase in the transmit signal after using PAPR reduction techniques. For example, *Tone Reservation* (TR) in Section 2.1.4 on Page12 requires more signal power because some of its power must be used for the peak reduction carriers.
Tone Injection (TI) in Section 2.1.4 uses a set of equivalent constellation points for an original constellation point to reduce PAPR. Since all the equivalent constellation points require more power than the original constellation point, the transmit signal will have more power after applying TI.

BER Increase at the Receiver Other techniques may have an increase in BER at the receiver if the transmit signal power is fixed or equivalently may require larger transmit signal power to maintain the BER after applying the PAPR reduction technique. For example, the BER after applying *Active Constellation Extension* in Section 2.1.5 on Page 12 will be degraded if the transmit signal power is fixed. In some techniques such as *SLM* in Section 2.1.3 on Page 11, the entire data block may be lost if the side information is received in error. This may also increase the BER at the receiver.

[1] Input back-off defined as $10 \log 10 \frac{P_{sat}}{P_{in}}$, which P_{sat} is the power that saturates the amplifier, and P_{in} is the input power. By increasing the input back-off, performance of the system improves. However larger back-off means less output power and as a result the efficiency of the amplifier with respect to the output power will be reduced.

Loss in Data Rate Some techniques require the data rate to be reduced, the block coding technique needs a portion of information symbols to be dedicated to control the PAPR. In *SLM* and *PTS*, the data rate is reduced due to the side information used to inform the receiver of what has been done in the transmitter. In these techniques the side information may be received in error unless some form of protection such as channel coding is employed. When channel coding is used, the loss in data rate due to side information is increased further [7].

Computational Complexity Techniques such as PTS in Section 2.1.6 on Page 13 find a solution for the PAPR reduced signal by using many iterations and performing exhaustive search, so it costs a lot to implement it in hardware. This report will propose a low complexity version of PTS which use more intellectual method to find optimum phase for PAPR minimization.

1.4 PAPR Problem in WiMAX

The WiMAX physical layer (PHY) is based on OFDM, however fixed and mobile version of WiMAX have slightly different implementation of the OFDM. Fixed WiMAX (IEEE 802.16) uses 256 IFFT block size and mobile WiMAX (IEEE 802.16$e-2005$)[2] uses scalable OFDMA. Hence WiMAX hardware equipments is vulnerable to non-linearity caused by high PAPR, more details about data structure in WiMAX can be found in Section 2.2.

In this thesis, we address PAPR problem in WiMAX base stations which transmit a larger number of OFDM subcarriers (1024) comparing the subscriber device which transmits smaller set of OFDM blocks, however the proposed algorithm is applicable in user devices as well, with some modifications.
Although it is up to the WiMAX vendors to implement PAPR reduction scheme, there is an option of PAPR reduction in the 802.16 standard. It is not part of the WiMAX profile and it is not widely used currently. clipping is a common technique but as we discussed it, it has serious drawbacks. For more information about clipping in WiMAX amplifiers, see [4]. WiMAX mobile devices (MS) have already been designed and are commercially available. Subscribers expect the providers to facilitate adaptive equipments which work simply with the user gadget. In other words, the MS devices should not be redesigned for different base stations, so the PAPR reduction algorithms is required to be implemented as a precoding technique at base station (BS).

1.5 Proposed Approach

Here we propose a novel technique to mitigate the high PAPR problem in WiMAX OFDMA systems. The proposed method is based on the WiMAX data structure in base stations, and it takes advantage of the idea of antenna beamforming weights in down link which is a part of the WiMAX standard. These are unique properties of the project:

- It preserves the transmitted power by adjusting only the phase of the beamforming weights per cluster.
- No extra side information regarding the phase change needs to be transmitted due to the property of dedicated pilots.
- Not sending the phase coefficients allows for arbitrary phase shifts instead of a quantized set such as used for PTS.
- A novel search algorithm based on gradient optimization to find the optimum cluster weights phase shifts.

[2]Although the scalable OFDMA scheme is referred to as mobile WiMAX, it can be used in fixed, nomadic and mobile application.

The whole 1024 OFDMA data block is divided into a number of disjoint block,where each sub-block is chosen to contain exactly one data cluster (each cluster has a separate phase weight). After doing IFFT for each sub block, it is multiplied by a phase coefficient so that the summation of the IFFT of sub-block, minimize the PAPR of the whole transmitted OFDM symbol. Up to here this is equivalent to conventional PTS, but the main strength of our technique is in the phase coefficient selection algorithm.

Each data cluster will be processed in its receiver subscriber device, and the channel estimation is performed to compensate for channel imperfection (channel equalization). Hence, changing the phase of a cluster at the transmitter, is just a different channel in the receiver perspective and will be inverted by the channel equalization procedure. So there is no need of sending explicit information to receiver about phase factors and the complete PAPR reduction process can be done at transmitter. As the matter of fact, this is how beamforming is performed in WiMAX, using dedicated pilot subcarriers in data unit, that are known symbols and are beamformed together with original data.

To summarize, phase coefficients are chosen in the continuous interval between $[0, 2\pi]$ with no constraints. That is the main difference between this method and conventional PTS, which chose the phase factors among a discrete phase set. Another important feature here is the unique algorithm to find optimum phase coefficients. Instead of doing exhaustive search, which is impossible here due to infinite number of phase sets, a novel optimization algorithm has been deployed. By formulating the problem as a minimax optimization problem it is solved by sequential quadratic programming approach which is developed for our specific application with high reliability and quite low complexity. One can read more about this topic in Section 3.2.

Chapter 2

PAPR Reduction

2.1 Review of PAPR Reduction Methods

PAPR reduction is a well-known signal processing topic in multi-carrier transmission and large number of techniques appeared in the literature during the past decades. These techniques include amplitude clipping and filtering, coding, tone reservation (TR) and tone injection (TI), active constellation extension (ACE) and multiple signal representation methods such as partial transmit sequence (PTS), selected mapping (SLM) and interleaving. The existing approaches are different from each other in terms of requirements, and most of them enforce various restriction to the system. Therefore, careful attention must be paid to choose a proper technique for each specific communication system. In this section we focus more closely on the PAPR reduction techniques for multi-carrier transmission. In order to evaluate the performance of these techniques, we need to look at the application and existing restrictions of a communication system.

2.1.1 Amplitude Clipping and Filtering

The simplest technique for PAPR reduction may be amplitude clipping. Amplitude clipping limits the peak envelop of the input signal to a predetermined value,

$$B(x) = \begin{cases} x & |x| \leq A \\ Ae^{j\phi(x)} & |x| > A \end{cases}$$

where $\phi(x)$ is the phase of x and A is the cutting threshold. The noise caused by amplitude clipping falls both in-band and out-of-band. In-band distortion can not be reduced by filtering and results in an error performance degradation, while out-of-band radiation reduces the spectral efficiency. Filtering the out-of-band distortion after clipping cause some peak regrowth and to avoid this undesirable effect, repeated clipping and filtering can be used [8][9][10].

2.1.2 Coding

Coding can also be used to reduce the PAPR, a simple idea is to select codewords that minimize or reduce the PAPR for transmission. This idea is illustrated in the following example; The PAPR for all possible data blocks for an OFDM signal with four subcarriers and binary phase shift keying (BPSK) modulation is shown in Figure 2.1. It can be seen from this table that among 2^3 possible status, four data blocks result in a PAPR of 6.0dB, and another four data blocks result in a PAPR of 3.7dB. It is clear that we could reduce PAPR by avoiding transmitting those sequences. This can be done by block coding the data such that the 3-bit data word is mapped onto a 4-bit codeword such that the set of permissible sequences does not contain those that result in high PAPR. The PAPR of the resulting signal is 2.3dB, a reduction of 3.7dB from that without block coding. However, this approach suffers from the need to perform an exhaustive search to find

the best codes and to store large lookup tables for encoding and decoding, especially for a large number of subcarriers [11][12].

Data block X	PAPR (dB)	Data block X	PAPR (dB)
[1 , 1 , 1 , 1]	6.0	[-1,-1 , 1 , 1]	3.7
[1 , 1 , 1 ,-1]	2.3	[-1,-1 , 1 ,-1]	2.3
[1 , 1 ,-1 , 1]	2.3	[-1, 1 , 1 , 1]	2.3
[1 , 1 ,-1 ,-1]	3.7	[-1, 1 , 1 ,-1]	3.7
[1 ,-1 , 1 , 1]	2.3	[-1, 1 ,-1 , 1]	6.0
[1 ,-1 , 1 ,-1]	6.0	[-1, 1 ,-1 ,-1]	2.3
[1 ,-1 ,-1 , 1]	3.7	[-1,-1 ,-1 , 1]	2.3
[1 ,-1 ,-1 ,-1]	2.3	[-1,-1 ,-1 ,-1]	6.0

Figure 2.1: An example of block coding for an OFDM symbol with four subcarriers.

2.1.3 Selected Mapping(SLM)

In the SLM technique, the transmitter generates a set of sufficiently different candidate data blocks, all representing the same information as the original data block, and selects the most favorable for transmission [13][14]. A block diagram of the SLM technique is shown in Figure 2.2.
Each data block is multiplied by U different phase sequences, each of length N, $B^{(u)} = [b_{u,0}, b_{u,1}, \cdots, b_{u,N-1}]^T$ $u = 1, 2, \cdots, U$, resulting in U modified data blocks. To include the unmodified data block in the set of modified data blocks, we set $B^{(1)}$ as the all-one vector of length N.

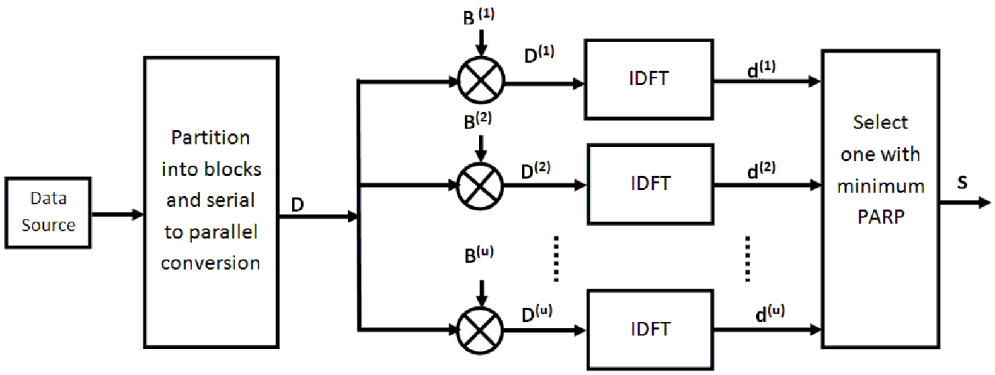

Figure 2.2: Block diagram of the SLM technique with U different signal representation and phase weights to produce a minimized PAPR signal.

Let us denote the modified data block for the u_{th} phase sequence $D^{(u)} = [D_0 b_{u,0}, D_1 b_{u,1}, \cdots, D_{N-1} b_{u,N-1}]^T$ $u = 1, 2, \cdots, U$, After applying SLM to X, the multi-carrier signal becomes

$$d_k^{(u)} = \sum_{n=0}^{N-1} D_n . b_{u,n} . e^{j \frac{2\pi nk}{N}} \quad 0 \leq k \leq N-1. \tag{2.1}$$

Among the modified data blocks $D^{(u)}$, the one with the lowest PAPR is selected for transmission. Information about the selected phase sequence need to be transmitted to the receiver as side information. At the receiver, the reverse operation is performed to recover the original data block. For implementation, the SLM technique needs U times IDFT operations, and the number of required side information bits is

$\log_2 U$ for each data block.

This approach is applicable with all types of modulation and any number of subcarriers. The amount of PAPR reduction for SLM depends on the number of phase sequences U and the design of the phase sequences. There are extensions of SLM technique which does not need explicit information to be sent to the receiver. For further information see the reference [18].

2.1.4 Tone Reservation and Tone Injection

"Tone reservation (TR) and tone interjection (TI), explained below, are two efficient techniques to reduce the PAPR of a multi-carrier signal. These methods are based on inserting data-block-dependent subcarriers into reserved or unused band in such a way that the PAPR is minimized for the whole block. This time domain signal can be easily computed at the transmitter and stripped off at the receiver" [7].

In the TR technique, the transmitter does not send data on a small portion of subcarriers that are optimized for PAPR reduction. The objective is to find the time domain signal to be added to the original time domain signal d such that the PAPR is reduced [15]. When a frequency domain vector $C = [C_0, C_1, \cdots, C_{N-1}]^T$ is added to D, the new time domain signal can be illustrated as $d + c = \text{IDFT}\{D + C\}$, where c is the time domain signal due to C. The data block D and peak reduction vector C must lie in disjoint frequency subspaces i.e, $D_n = 0, n \in \{i_1, i_2, \cdots, i_L\}$ and $C_n = 0, n \notin \{i_1, i_2, \cdots, i_L\}$) in TR technique. Peak Reduction Carriers (PRCs) are referred to the L non-zero positions in C, since the subcarriers are orthogonal, these additional signals cause no distortion on the subcarriers which carry the information. To find the value of $C_n, n \in \{i_1, i_2, \cdots, i_L\}$, a convex optimization problem needs to be solved that can easily be arranged as a linear programming (LP) problem.

"The basic idea in tone TI is to increase the constellation size so that each of the points in the original basic constellation can be mapped into several equivalent points in the expanded constellation [15]. Since each symbol in a data block can be mapped into one of several equivalent constellation points, these extra degrees of freedom can be exploited for PAPR reduction. This method is called tone injection because substituting a point in the basic constellation for a new point in the larger constellation is equivalent to injecting a tone of the appropriate frequency and phase in the multi-carrier signal " [7].

Assume that M-ary square quadrature amplitude modulation (QAM) is used as a modulation scheme and the minimum distance between constellation points is l. Accordingly, the real part and the imaginary part of D_n are R_n and I_n respectively, and they can take values $\pm l/2, \pm 3l/2, \cdots, \pm(\sqrt{M} - 1)l/2$ where $\sqrt{M}$ is equal to the number of levels per dimension. Assume that $D_n = l/2 + j3l/2$, modifying the real and/or imaginary part of D_n could reduce the PAPR of the transmit signal. Since the receiver should decode D_n correctly, D_n must be changed by an amount that can be estimated at the receiver. A simple case would be to transmit $D_n = D_n + pQ + jqQ$, where p and q are any integer values and Q is a positive real number known at the receiver [7].

The TI technique is similar to TR but could be more complicated since the injected signal takes the same frequency band as the data. It also results in a power increase on the transmit signal due to the inserted signal power. These techniques do not always comply with the standard, for example in IEEE 802.16d the spectrum loss required by this method is not acceptable, but it is suited for other implementations including IEEE 802.16e.

2.1.5 Active Constellation Extension (ACE)

Another family of methods altering the QAM constellation to reduce high signal peaks is ACE [16]. This technique allows the outer points in the QAM constellation to be moved dynamically within the quarter planes outside their nominal values. The inner points within the constellation are not modified to keep the

distance between different symbol levels. Thus the BER does not increase and the decision boundaries are preserved in the receiver. The transmitted power is increased due to expanding the outer symbols, while the BER will decrease slightly at the receiver. In [17], the performance of this method is analyzed and the theoretical gains promised in the literature is compared with those achieved in a real-world situation. The usefulness of this scheme is rather restricted for a modulation with a large constellation size. It is possible to combine the TR and ACE techniques to make the convergence of TR much faster.

Active constellation extension (ACE) is a PAPR reduction technique similar to TI. The main idea of this scheme is easily explained in the case of a multi-carrier signal with QPSK modulation in each subcarrier. In each subcarrier there are four possible constellation points that lie in each quadrant in the complex plane and are equidistant from the real and and imaginary axes. Assuming white Gaussian noise, the maximum likelihood decision regions are the four quadrants bounded by the axes; thus, a received data symbol is decided according to the quadrant in which the symbol is observed. Any point that is farther from the decision boundaries than the nominal constellation point (in the proper quadrant) will offer increased margin,which guarantees a lower BER [7].

"We can therefore allow modification of constellation points within the quarter-plane outside of the nominal constellation point with no degradation in performance. If adjusted intelligently, a combination of these additional signals can be used to partially cancel time domain peaks in the transmit signal. The ACE idea can be applied to other constellations as well, such as QAM and MPSK constellations, because data points that lie on the outer boundaries of the constellations have room for increased margin without degrading the error probability for other data symbols" [16].

2.1.6 Partial Transmit Sequence (PTS)

In the PTS technique, an input data block of N symbols is partitioned into disjoint sub-blocks. The subcarriers in each sub-block are weighted by a phase factor for that sub-block. The phase factors are selected such that the PAPR of the combined signal is minimized. Figure 2.3 shows the block diagram of the PTS technique [20].

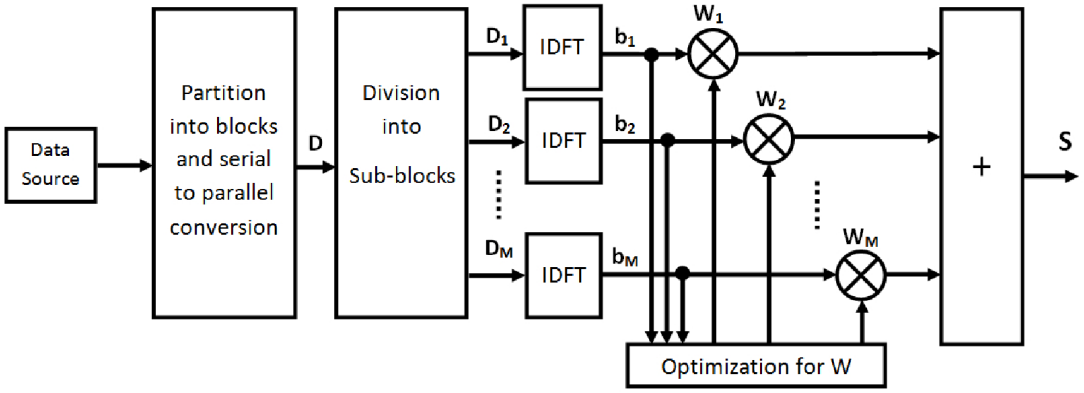

Figure 2.3: Block diagram of the PTS technique with M disjoint sub-blocks and phase weights to produce a minimized PAPR signal, quantized phase weights W are selected by exhaustive search among possible combinations.

In the conventional PTS, the input data block D is partitioned into M disjoint sub-blocks $D_m = [D_{m,0}, D_{m,1}, \cdots, D_{m,N-1}]^T$, $m = 1, 2, \cdots, M$, such that $\sum_{m=1}^{M} D_m = D$, and the sub-blocks are com-

bined to minimize the PAPR in the time domain. The L-times over-sampled time domain signal of D_m is obtained by taking an IDFT of length NL on D_m concatenated with $(L-1)N$ zeros, and is denoted by $b_m = [b_{m,0}, b_{m,1}, \cdots, b_{m,LN-1}]^T$, $m = 1, 2, \cdots, M$; these are called the partial transmit sequences. Complex phase factors, $W_m = e^{j\phi m}$, $m = 1, 2, \cdots, M$ are introduced to combine the PTSs which are represented as a vector $W = [W_1, W_2, \cdots, W_M]^T$ in the block diagram. The time domain signal after combination is given by

$$s(n) = \sum_{m=1}^{M} W_m b_m(n)\,. \tag{2.2}$$

The objective is to find a set of phase factors that minimize the PAPR. In general, the selection of the phase factors is limited to a set with a finite number of elements to reduce the search complexity. The set of possible phase factors is written as $P = e^{\frac{j2\pi l}{K}}$ $l = 0, 1, \cdots, K-1$, where K is the number of allowed phases. The first phase weight is set to 1 without any loss of performance, so a search for choosing the best one is performed over the $(M-1)$ remaining places. The complexity increases exponentially with the number of sub-blocks M, since K^{M-1} possible phase vectors are searched to find the optimum set of phases. Also, PTS needs M times IDFT operations for each data block, and the number of required side information bits is $\log_2(K^{M-1})$ to send to the receiver.

The amount of PAPR reduction depends on the number of sub blocks M and the number of allowed phase factors K. Another factor that may affect the PAPR reduction performance in PTS is the sub-block partitioning, which is the method of division of the subcarriers into multiple disjoint sub-blocks. There are three kinds of sub-block partitioning schemes: adjacent, interleaved, and pseudo-random partitioning. Among them, pseudo-random partitioning has been found to be the best choice.

The PTS technique works with an arbitrary number of subcarriers and any modulation scheme. As mentioned above, the ordinary PTS technique has exponentially increasing search complexity. To reduce the search complexity, various techniques have been suggested. For example,iterations for updating the set of phase factors are stopped once the PAPR drops below a pre-set threshold. These methods achieve significant reduction in search complexity with marginal PAPR performance degradation [19].

Here, we show a simple example of the PTS technique for an OFDM system with eight subcarriers that are divided into four sub blocks. The phase factors are selected in $P = \{+1, -1\}$. Figure 2.4 shows the adjacent sub-block partitioning for a data block D of length 8. The original data block X has a PAPR of 6.5dB and there are $8 = 2^{(4-1)}$ way to combine sub blocks with fixed $b_1 = 1$ and among them $[W_1, W_2, W_3, W_4]^T = [1, -1, -1, -1]^T$ achieves the lowest PAPR.

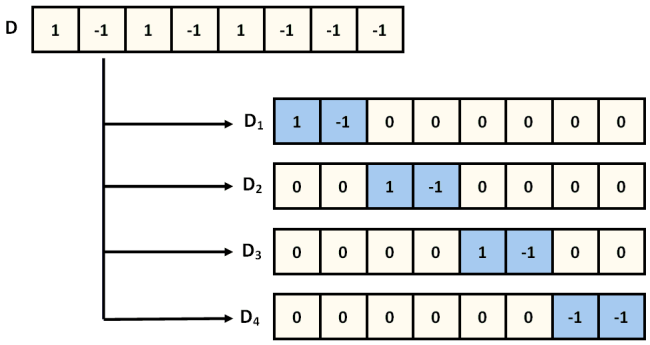

Figure 2.4: Example of adjacent sub block partitioning in PTS

The modified data block whose PAPR is 2.2dB, resulting in a 4.3dB reduction. In this case, the number

of required IDFT operations is 4 and the amount of side information is 3 bits. The side information must be transmitted to the receiver to recover the original data block. One way to do this is to transmit these side information bits with a separate channel other than the data channel. It is also possible to include the side information within the data block; however, this results in a data rate loss [7].

2.2 Formulation of the PAPR Problem in WiMAX Systems

In this Section we try to look at the WiMAX configuration and data structure, the capabilities in the standard for PAPR reduction purposes is studied as well as practical restrictions. Also a novel technique is introduced to mitigate the PAPR exploiting the potentials in standard and the problem is formulated as an optimization approach.

WiMAX allows DL beamforming using separate weights per cluster. Both data and pilots are beam formed so from the receiver perspective, beamforming just results in a different channel and this is compensated for in channel equalization. In the simplest form the new channel frequency response, H will be calculated by observing the known pilot subcarriers, and the whole received data unit is multiplied with the inverse of the channel frequency response, H^{-1}. So the receiver does not need to know the transmitted weights and the BER and transmitted power does not change.

2.2.1 System Model

Consider an OFDM system, where the data is represented in the frequency domain. The time domain signal $s(n)$, $n = 1, 2, \ldots, N$, where N denotes the FFT size is calculated from the frequency domain symbols $D(k)$ using an IFFT as [32].

$$s(n) = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} D(k) e^{\frac{j2\pi kn}{N}} \ . \tag{2.3}$$

Note that the frequency domain signal $D(k)$ typically belong to QAM constellations. In the case of WiMAX; QPSK, 16QAM and 64QAM constellations are used. The metric that will be used to measure the peaks in the time-domain signal is the peak-to-average-power-ratio (PAPR) metric defined as

$$PAPR = \frac{\max\limits_{0 \leq n \leq N-1} |s_n|^2}{E\{|s_n|^2\}}. \tag{2.4}$$

Although not explicitly written in (2.4), it is well known that oversampling is required to accurately capture the peaks. In this paper, an oversampling of four times is used.

The WiMAX protocol defines several different DL transmission modes, of which the DL-PUSC mode is the most widely used and is the focus here. The minimum unit of scheduling a transmission is a sub-channel, which here spans multiple clusters. One cluster spans 14 subcarriers over two OFDM symbols containing four pilots and 24 data symbols which is illustrated in Figure 2.5. For a 10MHz system there are a total of 60 clusters. A sub-channel is spread over eight or twelve clusters of which only two or three data subcarriers from each cluster are used. The sub-channel carries 48 data symbols. For example, logical sub-channel zero uses two data subcarriers from twelve clusters over two OFDM symbols to reach 48 data symbols.

To extract frequency diversity, the WiMAX protocol specifies that the clusters in a sub-channel are spread out across the band, i.e. a distributed permutation. The WiMAX standard further specifies two main modes of transmitting pilots; Common pilots and dedicated pilots. Here, dedicated pilots allow per-cluster beamforming since channel estimation is performed per-cluster while for common pilots channel estimation across the whole band is allowed. Beamforming in this context is defined as sending the same

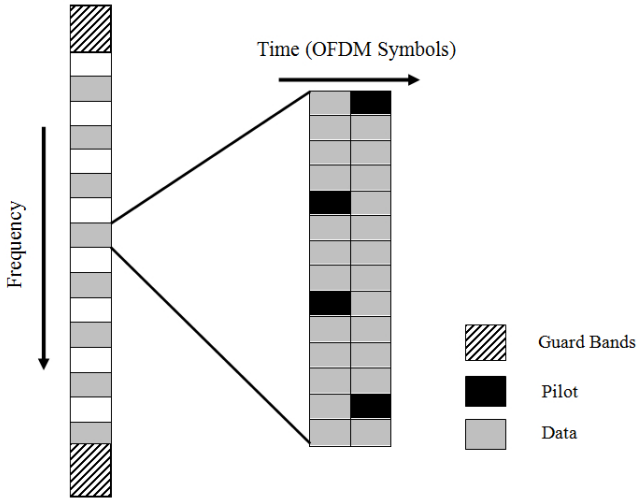

Figure 2.5: Structure of DL-PUSC permutation in WiMAX, where the transmission bandwidth is divided into 60 clusters of 14 sub-carriers over two symbols each.

message from different antennas but using different weights [1]. The transmitted signal from a single antenna s_s at tone n can be written as

$$s_s(n) = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} D(k) W_s(k) e^{\frac{2\pi kn}{N}}. \tag{2.5}$$

where $W_s(k)$ denotes the beamforming weight on subcarrier k. Note that we only consider phase shifts, so $W_s(k) = e^{j\phi(k)}$. Since the channel is estimated using the pilots in each cluster, the beamforming weights need to be constant over each cluster but can change from cluster to cluster, i.e. $W_s(k_0) = W_s(k_0 + 1) = \ldots = W_s(k_0 + 13)$, where k_0 denotes the first subcarrier in a particular cluster.

The presentation so far has ignored several practical details, such as guard bands which are inserted to reduce spectral leakage. In WiMAX a number of subcarriers in the beginning and the end of the available bandwidth do not carry any signal, leaving N_{usable} subcarriers that carries data and pilots. Although this number depends on bandwidth and transmission modes, weights that are constant across each cluster are simply applied to only the N_{usable} subcarriers.

2.2.2 Proposed Technique

Cluster weights can be used to decrease the PAPR of the OFDM symbol, that is choosing weights which minimize the PAPR. The easiest way to do this would be the exhaustive search among a combination of weight vectors containing a set of phase, for example ± 1, in this case the whole search space for aforementioned data unit will be 2^{60} which takes huge amount of computation. Here we introduce a realistic algorithm to find the optimum weights.

The technique exploits dedicated pilots for beamforming, which is a feature common in next generation wireless systems. For example, in several 4G systems such as WiMAX [4] beamforming weights are not explicitly announced but instead both pilots and data are beamformed using the same weights. In the WiMAX

[1] In MIMO techniques, the available antenna elements can be used to adjust the strength of transmitted or received signals based on their direction. This focusing of energy is achieved by choosing appropriate weights for each antenna element with a certain criteria [4].

downlink (DL), weights are applied in units of clusters (14 subcarriers), and in the uplink (UL) in units of tiles (four subcarriers). Multiplying the beamforming weights with a different complex scalar for each cluster or tile does not change the beamforming performance or the bit-error-rate, but it can substantially reduce PAPR.

Finally, we are ready to formulate the optimization problem of finding the weights that minimize the PAPR as

$$W_s = \arg\min_{W_s} \max_n \left| \sum_{k=0}^{N-1} D(k) W_s(k) e^{\frac{2\pi k n}{N}} \right|^2 . \tag{2.6}$$

Note that for a 10 MHz WiMAX system, there are 60 unique phase shifts in W_s, and these 60 phases are sought in the interval $[0, 2\pi)$.

In essence, the technique is similar to partial-transmit-sequence (PTS), but without the drawbacks of requiring side-information which would make it impossible to apply in existing communication standards such as WiMAX[2]. Further, using unit norm complex scalars does not increase the overall transmit power which is another drawback of many PAPR reduction techniques [7].

The PAPR reduction technique proposed here is transparent to the receiver and thus does not require any modification to existing receivers and wireless standards. It does not increase the required transmit power nor does it increase the bit-error-rate. Finally, it does not require any side information that reduces throughput and requires standard support. These advantages makes it a very attractive technique to reduce PAPR.

Since the IFFT is a linear operation, the phase weights can be multiplied either before the IFFT blocks or after it and the result will be the same, anyhow it would be easier in terms of implementation complexity to apply phase coefficient after IFFT, this would be exactly the same approach as PTS which has been explained with descriptive figure in Section 2.1.6.
Here we define Matrix B which is 1024×60 matrix, contains summation of IFFT weights within a cluster and should be calculated efficiently.Ordinary calculation costs 60 IFFT blocks of size 1024 which means $60 \times (1024/2)\log_2(1024) \approx 3 \times 10^5$ complex multiplies , this can be reduced effectively by some interleaving and Cooley-Tukey FFT algorithm , which is studied in Section 5.1.
The transmitted sequence $\mathbf{s}$ is illustrated as a multiplication of matrices $\mathbf{B}$ and $\boldsymbol{\phi}$ in (2.7). Number of columns (60), shows the number of disjoint sub blocks and each sub block is multiplied with a separate phase weight.

$$\mathbf{s} = \begin{bmatrix} b_{1,1} & b_{1,2} & \cdots & b_{1,M} \\ b_{2,1} & b_{2,2} & \cdots & b_{2,M} \\ b_{3,1} & b_{3,2} & \cdots & b_{3,M} \\ \vdots & \vdots & \ddots & \vdots \\ b_{LN,1} & b_{LN,2} & \cdots & b_{LN,M} \end{bmatrix} \begin{bmatrix} e^{j\phi_1} \\ e^{j\phi_2} \\ e^{j\phi_3} \\ \vdots \\ e^{j\phi_M} \end{bmatrix} . \tag{2.7}$$

Here, we rewrite the optimization problem to find the optimum phase set $\boldsymbol{\phi}$ as

$$\boldsymbol{\phi} = \arg\min_{\phi_m} \max_n |s(n)|^2 . \tag{2.8}$$

Where

$$s(n) = \sum_{m=1}^{M} b_{n,m} e^{j\phi_m} . \tag{2.9}$$

[2]One technique of inserting a PAPR reducing sequence is part of the IEEE 802.16e standard. It is activated using the PAPR Reduction/Sounding Zone/Safety Zone Allocation IE. Using this technique reduces the throughput since it requires sending additional PAPR bits. It is also not part of the WiMAX profiles so it is likely not supported by the majority of handsets.

The $s(n)$s are complex values and ϕ_ns are continuous phases between $[0, 2\pi)$. Substituting $b_{n,m} = R_{n,m} + jI_{n,m}$ and $e^{j\phi m} = \cos\phi_m + j\sin\phi_m$ in (2.9) and taking the square of $|s(n)|$ results in (2.10), when $R_{n,m}$ and $I_{n,m}$ stands for $\Re\{b_{n,m}\}$ and $\Im\{b_{n,m}\}$ respectively. This is a very important equation, which shows the square of the norm or the power of output subcarriers that are transmitted; a multi-variable cost function to be minimized when the largest $|s(n)|$ specifies the PAPR of the system.

$$\begin{aligned} |s(n)|^2 \quad &= \Big((R_{n,1}\,\cos\phi_1 + \cdots + R_{n,M}\,\cos\phi_M) - (I_{n,1}\,\sin\phi_1 + \cdots + I_{n,M}\,\sin\phi_M)\Big)^2 \\ &+ \Big((R_{n,1}\,\sin\phi_1 + \cdots + R_{n,M}\,\sin\phi_M) + (I_{n,1}\,\cos\phi_1 + \cdots + I_{n,M}\,\cos\phi_M)\Big)^2 \end{aligned} \tag{2.10}$$

Both PAPR reduction and beamforming can be done together; for instance, if we have a beamforming vector $\mathbf{v} = [v_1, v_2, v_3, v_4]^T$ for a four antenna system we can apply our PAPR reduction scheme by using a modified vector $\mathbf{w} = e^{j\phi}\mathbf{v}$ since the beamforming performance is not affected by a multiplication of a complex scalar on all elements of $\mathbf{v}$. Note that if the channel is $\mathbf{h}(4\times 1)$, the BF gain is $|\mathbf{v}^H\mathbf{h}|^2 = |\mathbf{w}^H\mathbf{h}|^2$.

In the following chapter we introduce algorithms to solve the above optimization problem. A revolutionary optimization technique is used to determine the phase coefficients and also a gradient base optimization approach is proposed which is shown to be more promising for this problem.

Chapter 3

Phase Optimization

Now we want to find the best phase coefficients which are multiplied by PTS sub-blocks and reduce the overall peak to average power ratio when sub-blocks are summed up together.
Two main approaches have been followed in this thesis to solve this optimization problem:

1. Evolutionary algorithm (Particle Swarm Optimization)

2. Minimax algorithm (Sequential Quadratic programming)

3.1 Particle Swarm Optimization

When there is no explicit knowledge about the gradient of the optimization problem, numerical optimization could be a choice. Particle swarm optimization (PSO) is a subset of evolutionary algorithm like Genetic algorithms [1] which can be performed to derive the solution. PSO optimizes a problem by keeping a population of possible solutions called particles and moving these particles around in the search-space according to a criteria. The particles activity are directed according to the best found positions in the search-space, which are continually updated as better positions are found by the particles.

There are two parameters to characterize each particle; position and velocity which is explained below. The algorithm evaluates particles with the fitness value which is the objective function of the particles. A swarm with N particles randomly is generated in solution space, and in case of D dimensional, in the t^{th} iteration, the i^{th} particles position vector and velocity vector are $\mathbf{x}_i^t = (x_{i1}^t, x_{i2}^t, \cdots, x_{iD}^t)$ and $\mathbf{v}_i^t = (v_{i1}^t, v_{i2}^t, \cdots, v_{iD}^t)$, where $\mathbf{x}_{id}^t \in U$, and U denotes the domain of objective function.

After multiple iterations, the optimum solution will be searched. In each iteration, particle updates itself through tracking two best positions: the first one is called local best position and represented as $\mathbf{p}_i = (p_{i1}, p_{i2}, \cdots, p_{iD})$; the position vector of the best solution this particle has achieved so far. The other one is called global best position and represented as $\mathbf{p}_g = (p_{g1}, p_{g2}, \cdots, p_{gD})$, this is the best position obtained so far by any particle in the whole swarm. The update of position and velocity for each particle described as,

$$\mathbf{v}_{id} = w_i\, \mathbf{v}_{id} + c_1\, r_1\, (\mathbf{p}_{id} - \mathbf{x}_{id}) + c_2\, r_2 (\mathbf{p}_{gd} - \mathbf{x}_{id}). \tag{3.1}$$

$$\mathbf{x}_{id} = \mathbf{x}_{id} + \mathbf{v}_{id}. \tag{3.2}$$

[1] Genetic algorithms are a particular class of evolutionary algorithms (EA) that use techniques inspired by evolutionary biology such as inheritance, mutation, selection, and crossover.

Where $i = 1, 2, \cdots, N$ and $d = 1, 2, \cdots, D$. The acceleration terms are c_1 and c_2. The constants, r_1 and r_2 are uniform distribution random number in range of [0,1]; w_i is inertia factor. Parameters c_1 , c_2 and w_i are all assigned before system simulation.

In our case, D represents the length of vector $\mathbf{x}$, contains of phase weights or the number of PTS sub-blocks. The initial particles number is N, which build the swarm and the objective function or fitness value is shown by Y, which is the PAPR of the OFDM symbol which is specified in (1.3).

3.1.1 PSO Pseudocode

According to the above explanation, this is an instruction to implement PSO with N particles for a D dimensional problem.
Step 0 Initialization of the particle swarm:

- Generate N different[2] $\mathbf{x}_i^1$, which means N different phase vectors of the length of PTS sub-blocks;
- Initialize the velocity $\mathbf{v}_i^1$ by zeros, note the size of $\mathbf{v}$ and $\mathbf{x}$ are the same;
- Compute fitness value for each particle;
- Initialize the particles local best position with their initial position $\mathbf{x}_i^1$;
- Initialize global best position as the minimum of current particles local best position.

Step 1 Update particles (the $t + 1_{\text{th}}$ iteration):

- Update the velocity according to (3.1)
- Update the position according to (3.2)
- Update the local best position as: $\mathbf{p}_{id} = \underset{k=1,2,\cdots,t+1}{\arg\min} \mathrm{Y}(\mathbf{x}_{\mathrm{id}}^{\mathrm{k}})$;
- Update the global best position as: $\mathbf{p}_{gd} = \underset{i=1,2,\cdots,N}{\arg\min} \mathrm{Y}(\mathbf{p}_{\mathrm{id}})$.

Step1.5 [3] Filtration of active particles; Assuming that $Y_b(n)$ denotes the fitness of the local best position for the active particles, while $n = 1, 2, \cdots, N$, N denotes number of active particle at present.

- Compute truncation threshold :$T = \min(Y_b(n)) + \big(\max(Y_b(n)) - \min(Y_b(n))\big) \times R$, $R \in [0, 1]$ denotes truncation ratio.
- Filtration of active particles, the active particles in the next iteration are: $\mathbf{x} = \{\mathbf{x}_i | Y_b(\mathbf{x}_i) < T\}$

Step3 Compare the current iteration number t with the maximum iteration number M ,
if $t < M$ **then**
Go back to step2
else
Stop iteration[4]
end if

[2] This can be done by randomly generate a matrix of size $N \times D$ or can be a selection between possible phase sets in conventional PTS with limited phase weights

[3] In order to reduce the computational complexity, the number of initial particles can be eliminated during the algorithm and those who are not close enough to the fitness will no longer participate in the process and the algorithm goes on with smaller number of particles.

[4] Other satisfaction criteria can be replaced in this step, for example, if the Y drops a certain value.

3.1.2 Parameter Tuning

Adjusting the parameters for the algorithm affects the convergence of the problem to the global solution and one can achieve the result much quicker, if the coefficients in (3.2) are selected properly. An efficient way of tuning the PSO parameters for different optimization problems and settings was presented by Pedersen et al. [26] [27]. The technique for tuning PSO parameters is referred to as meta-optimization because another optimization method is used in an overlaying manner to tune the PSO parameters.

The PSO method to find the optimum phase set works well for our PTS schema when the number of sub-blocks are limited (up to 8), but the complexity of algorithm grows exponentially when the number of sub-blocks increases, in a way that, it is almost impossible to implement it in a practical case when the algorithm is running in real time communication, and this is the major motivation to search for a better algorithm. One can see the simulation results and more details on complexity calculation in Chapter 4 and 5.

3.2 Minimax Approach

In this section, a new algorithm is introduced to solve the minimization problem in (2.8). As can be seen in Section 2.2 the multi-variable objective function for problem can be defined explicitly as (2.10). To emphasis on the role of objective function, the $|s(n)|^2$ is replaced with $f_n(\boldsymbol{\phi})$ as expressed in (3.3).

$$\begin{aligned} f_n(\boldsymbol{\phi}) \quad &= \underbrace{\Big((R_{n,1}\,\cos\phi_1 + \cdots + R_{n,M}\,\cos\phi_M) - (I_{n,1}\,\sin\phi_1 + \cdots + I_{n,M}\,\sin\phi_M)\Big)^2}_{A} \\ &+ \underbrace{\Big((R_{n,1}\,\sin\phi_1 + \cdots + R_{n,M}\,\sin\phi_M) + (I_{n,1}\,\cos\phi_1 + \cdots + I_{n,M}\,\cos\phi_M)\Big)^2}_{B} \end{aligned} \tag{3.3}$$

Clearly, the multi-variable objective function is continuous and differentiable over $[0, 2\pi)$, so its gradient can be derived analytically and this is a key property to develop a solution. Knowing the gradient of the objective function, the problem can be solved using wide range of gradient based optimization methods. The gradient of $|s(n)|^2$ as a function of phase vector $\boldsymbol{\phi} = [\phi_1, \phi_2, \cdots, \phi_M]$ is defined as the vector $\nabla f_n = [\frac{\partial f_n}{\partial \phi_1}, \frac{\partial f_n}{\partial \phi_2}, \cdots, \frac{\partial f_n}{\partial \phi_M}]^T$.

$$\frac{\partial f_n(\boldsymbol{\phi})}{\partial \phi_m} = -2A\Big(R_{n,m}\sin\phi_m + I_{n,m}\cos\phi_m\Big) + 2B\Big(R_{n,m}\cos\phi_m - I_{n,m}\sin\phi_m\Big) \tag{3.4}$$

The Jacobian matrix is defined in (3.5) where M is the number of sub-blocks and LN the length of the vector $\mathbf{s}$ (oversampled OFDM symbol). The n_{th} row of this matrix is the gradient of the $f_n(\boldsymbol{\phi})$.

$$\mathbf{J} = \begin{bmatrix} \frac{\partial f_1}{\partial \phi_1} & \frac{\partial f_1}{\partial \phi_2} & \cdots & \frac{\partial f_1}{\partial \phi_M} \\ \frac{\partial f_2}{\partial \phi_1} & \frac{\partial f_2}{\partial \phi_2} & \cdots & \frac{\partial f_2}{\partial \phi_M} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_{LN}}{\partial \phi_1} & \frac{\partial f_{LN}}{\partial \phi_2} & \cdots & \frac{\partial f_{LN}}{\partial \phi_M} \end{bmatrix}. \tag{3.5}$$

The elements of Jacobian matrix is expressed in (3.4) .

3.2.1 Minimax Problem

The minimax optimization minimizes the largest value in a a set of multi-variable functions. An initial estimate of the solution is made to start with, and the algorithm proceeds by moving towards the minimum;

this is generally defined as,

$$\begin{array}{cc} \text{minimize} & \max\{f_n(\boldsymbol{\phi})\} \\ \boldsymbol{\phi} & 1 \leq n \leq N \end{array} \tag{3.6}$$

To eliminate the PAPR, the objective of the optimization problem is to minimize the greatest value of $|s(n)|^2$ in (2.10) which is analogous to $\max\{f_n(\boldsymbol{\phi})\}$ in (3.6). Here, we reformulate the problem into an equivalent Non-linear Linear Programming problem as (3.7), in order to solve it using sequential quadratic programming (SQP) technique.

$$\begin{array}{cc} \text{minimize} & f(\boldsymbol{\phi}) \\ \boldsymbol{\phi} & \\ \text{subject to} & \\ f_n(\boldsymbol{\phi}) \leq f(\boldsymbol{\phi}) & \text{Nonequality Constraints} \end{array} \tag{3.7}$$

In agreement with this new setting, the objective function $f(\boldsymbol{\phi})$ is the maximum of $f_n(\boldsymbol{\phi})$, or equivalently it is the greatest IFFT sample in the whole OFDM sequence which characterizes the PAPR value. The remaining samples are appended as additional constraints, in the form of $f_n(\boldsymbol{\phi}) \leq f(\boldsymbol{\phi})$. In fact, the $f(\boldsymbol{\phi})$ is minimized over $\boldsymbol{\phi}$ using SQP, and the additional constraints are considered because we do not want other f_ns pop out when the maximum value is being minimized. In this way, the whole OFDM sequence is kept smaller than the value that is being minimized during iterations.

3.2.2 Sequential Quadratic Programming (SQP)

SQP is one of the most popular and robust algorithms for non-linear constraint optimization. Here, it is particularly modified and simplified for the phase optimization problem of PAPR reduction, but the basic configuration is the same as general SQP. The algorithm proceeds based on solving a set of subproblems created to minimize a quadratic model of the objective, subject to a linearization of the constraints. For example, a version of SQP has been implemented and tested by Schittkowski[40] that surpass every other methods in terms of efficiency, accuracy, and percentage of successful solutions, over an extensive number of sample problems. An overview of the SQP method can be found in [32, 33, 34].

The Kuhn-Tucker (KT) equations are necessary conditions for optimality for a constrained optimization problem. If the problem is a convex programming problem, then the KT equations are both necessary and sufficient for a global solution point [35]. For this specific PAPR problem, the largest sample should be kept as small as possible and there is no predetermined value in the problem formulation to be preserved during the algorithm, so equality constraints do not exist here. The KT equations for the phase optimization problem are stated as the following expression where λ_ns are the Lagrange multipliers of the constraints.

$$\begin{array}{c} \nabla f(\boldsymbol{\phi}) + \sum_{n=1}^{N} \lambda_n . \nabla f_n(\boldsymbol{\phi}) = 0, \\ \lambda_n \ \geq \ 0. \end{array} \tag{3.8}$$

These equations used to form the quasi Newton updating step in 3.11 by accumulating second order information of KT criteria and also checking for optimality during iterations.

The SQP implementation consists of two loops; the phase solution is updated at each iteration in major loop with k as the counter, while itself contains an inner QP loop to solve for optimum search direction $\mathbf{d}_k$.

Major loop to find $\boldsymbol{\phi}$ which minimize the $f(\boldsymbol{\phi})$:
while $k <$maximum number of iterations **do**
  $\boldsymbol{\phi}_{k+1} = \boldsymbol{\phi}_k + \mathbf{d}_k$,
  QP loop to determine $\mathbf{d}_k$ for major loop:
  while optimal $\mathbf{d}_k$ found **do**
    $\mathbf{d}_{l+1} = \mathbf{d}_l + \alpha \mathbf{d}_l$,
  end while

end while

The step length α is determined within the QP iterations which is distinguished from major iterations by index l as the counter.

The Hessian of the Lagrange function is required to form the quadratic objective function in (3.12). This Hessian matrix of second derivative does not need to be evaluated directly and is approximated at each major iteration using a quasi Newton updating method, when the Hessian matrix is estimated using the information specified by gradient evaluations. The Broyden Fletcher Goldfarb Shanno (BFGS) is one of the most attractive members of quasi Newton methods , and a common part in non-linear optimization; it approximates the second derivative of the objective function using (3.9).

Quasi Newton methods are a generalization of the secant method to find the root of the first derivative for multidimensional problems [36]. Convergence of the multi-variable function $f(\boldsymbol{\phi})$ can be observed dynamically by evaluating the norm of the gradient $|\nabla f(\boldsymbol{\phi})|$. Practically, the first Hessian can be initialized with an identity matrix ($\mathbf{H}_0 = \mathbf{I}$), so that the first step is equivalent to a gradient descent, while further steps are gradually refined by $\mathbf{H}_k$, which is the approximation to the Hessian [37]. The updating formula for the Hessian matrix $\mathbf{H}$ in each major iteration is given by,

$$\mathbf{H}_{k+1} = \mathbf{H}_k + \frac{\mathbf{q}_k\,\mathbf{q}_k^T}{\mathbf{q}_k^T\,\mathbf{s}_k} - \frac{{\mathbf{H}_k}^T\,\mathbf{H}_k}{\mathbf{s}_k^T\,\mathbf{H}_k\,\mathbf{s}_k}. \tag{3.9}$$

where H is $M \times M$ matrix and λ_n is the Lagrange multipliers of the objective function $f(\boldsymbol{\phi})$.

$$\begin{array}{rcl} \mathbf{q}_k & = & \nabla f(\boldsymbol{\phi}_{k+1}) + \sum_{n=1}^{N} \lambda_n . \nabla f_n(\boldsymbol{\phi}_{k+1}) \\ & - & \nabla f(\boldsymbol{\phi}_k) + \sum_{n=1}^{N} \lambda_n . \nabla f_n(\boldsymbol{\phi}_k). \end{array} \tag{3.10}$$

$$\mathbf{s}_k = \boldsymbol{\phi}_{k+1} - \boldsymbol{\phi}_k. \tag{3.11}$$

The Lagrange multipliers (according to (3.8)) is non-zero and positive for active set constraints, and zero for others. The $\nabla f_n(\boldsymbol{\phi}_k)$ is the gradient of n_{th} constraints at the k_{th} major iteration. The Hessian is maintained positive definite at the solution point provided $\mathbf{q}_k^T\,\mathbf{s}_k$ is positive at each update. Powell [41] proposed to modify q_k on an element-by-element basis so that $\mathbf{q}_k^T\,\mathbf{s}_k > 0$ [5].

After the above update at each major iteration, a QP problem is solved to find the step length $\mathbf{d}_k$, which minimizes the SQP objective function $f(\boldsymbol{\phi})$. In fact, the complex nonlinear problem in (3.7) is broken down to several convex optimization sub problems of the form (3.12), which can be solved with known programming techniques. The quadratic objective function is referred to, as $q(\mathbf{d})$ below.

$$\begin{array}{c} \text{minimize} \\ \mathbf{d} \in \Re^n \\ \text{subject to} \\ \nabla f_n(\boldsymbol{\phi}_k)^T \mathbf{d} + f_n(\boldsymbol{\phi}_k) \leq 0 \end{array} \qquad q(\mathbf{d}) = \tfrac{1}{2}\mathbf{d}^T\,\mathbf{H}_k\,\mathbf{d} + \nabla f(\boldsymbol{\phi}_k)^T\,\mathbf{d} \tag{3.12}$$

We generally refer to the constraints of the QP sub-problem as $\mathbf{G}(\mathbf{d}) = \mathbf{A}\,\mathbf{d} - \mathbf{a}$, where $\nabla f_n(\boldsymbol{\phi}_k)^T$ and $-f_n(\boldsymbol{\phi}_k)$ are the n_{th} row and element of the matrix $\mathbf{A}$ and vector $\mathbf{a}$ respectively.

The $q(\mathbf{d})$ is supposed to reflect the local properties of the original objective function and the main reason to use quadratic function is that such problems are easy to solve and yet mimics the nonlinear behavior of

[5] "The general aim of this modification is to distort the elements of $\mathbf{q}_k$, which contribute to a positive definite update, as little as possible. Therefore, in the initial phase of the modification, the most negative element of $\mathbf{q}_k^T\mathbf{s}_k$ is repeatedly halved. This procedure is continued until $\mathbf{q}_k^T\,\mathbf{s}_k$ is greater than or equal to a small negative tolerance. If, after this procedure, $\mathbf{q}_k^T\,\mathbf{s}_k$ is still not positive, modify $\mathbf{q}_k$ by adding a vector $\mathbf{v}$ multiplied by a constant scalar w, and increase $\mathbf{w}$ systematically until $\mathbf{q}_k^T\,\mathbf{s}_k$ becomes positive" see [26].

the initial problem. The reasonable choice for the objective function is the local quadratic approximation of $f(\boldsymbol{\phi}_k)$ at the current solution point and the obvious option for the constraints is the linearization of current constraints in original problem around $\boldsymbol{\phi}_k$ to form a convex optimization problem. This is illustrated in (3.12) when $f_n(\boldsymbol{\phi}_k)^T$ is the matrix $\mathbf{J}$ in (3.5). In the next section we try to explain the QP algorithm which is solved iteratively by updating the initial solution, and since the notation of variables might be tricky, we sort them here.

- $\mathbf{d}_k$ is a search direction in the major loop while $\acute{\mathbf{d}}_l$ is the search direction in the QP loop.
- k is used as an iteration counter in the major loop and l is the counter in the QP loop.
- $\boldsymbol{\phi}_k$ is the minimization variable in the major loop, it is the phase vector in this problem.
- $\mathbf{d}_l$ is the minimization variable in the QP problem.
- $f_n(\boldsymbol{\phi}_k)$ is the n_{th} constraint of the original minimax problem at a solution point $\boldsymbol{\phi}_k$.
- $\mathbf{G}(\mathbf{d}_l) = \mathbf{A}\,\mathbf{d}_l - \mathbf{a}$ is the matrix represents the constraint of the QP sub-problem at a solution point $\mathbf{d}_l$ and $g_n(\mathbf{d}_l)$ is the n_{th} constraint.

3.2.3 Quadratic Programming (QP)

In a quadratic programming (QP) problem, a multi-variable quadratic function is maximized or minimized, subject to a set of linear constraints on these variables. Basically, the quadratic programming problem can be formulated as; minimizing $f(\mathbf{x}) = 1/2\,\mathbf{x}^T\,\mathbf{C}\,\mathbf{x} + \mathbf{c}^T\,\mathbf{x}$ with respect to $\mathbf{x}$, with linear constraints $\mathbf{A}\mathbf{x} \leq \mathbf{a}$, which shows that every element of the vector $\mathbf{A}\mathbf{x}$ is less than or equal to the corresponding element of the vector $\mathbf{a}$.

The quadratic program has a global minimizer if there exists some feasible vector $\mathbf{x}$ satisfying the constraints, provided that $f(\mathbf{x})$ is bounded in constraints on the feasible region; this is true when the matrix $\mathbf{C}$ is positive definite. Naturally, the quadratic objective function $f(\mathbf{x})$ is convex, so as long as the constraints are linear we can conclude the problem has a feasible solution and a unique global minimizer. If $\mathbf{C}$ is zero, then the problem becomes a linear programming [31].

A variety of methods are commonly used for solving a QP problem; the Active Set Strategy has been applied in the phase optimization algorithm. We will see how this method is suitable for problems with a large number of constraints. In general, the active set strategy includes an objective function to optimize and a set of constraints which is defined as $g_1(\mathbf{d}) \leq 0, g_2(\mathbf{d}) \leq 0, \cdots, g_n(\mathbf{d}) \leq 0$ here. That is a collection of all $\mathbf{d}$ which introduce a feasible region to search for the optimal solution. Given a point $\mathbf{d}$ in the feasible region, a constraint $g_n(\mathbf{d}) \leq 0$ called active at $\mathbf{d}$ if $g_n(\mathbf{d}) = 0$ and inactive at $\mathbf{d}$ if $g_n(\mathbf{d}) < 0$ [6]. The active set at $\mathbf{d}$ is made up of those constraints $g_n(\mathbf{d})$ that are active at the current solution point.

The active set specifies which constraints will particularly control the final result of the optimization, so they are very important in the optimization. For example, in quadratic programming as the solution is not necessarily on one of the edges of the bounding polygon, specification of the active set creates a subset of inequalities to search the solution within [29][38][39]. As a result, the complexity of the search is reduced effectively. That is why non-linearly constrained problems can often be solved in fewer iterations than unconstrained problems using SQP; because of the limits on the feasible area.

In the phase optimization problem, the QP subproblem is solved to find the $\mathbf{d}_k$ vector which is used to form a new $\boldsymbol{\phi}$ vector in the k_{th} major iteration, $\boldsymbol{\phi}_{k+1} = \boldsymbol{\phi}_k + \mathbf{d}_k$. The matrix $\mathbf{Q}$ in the general problem is replaced with the positive definite Hessian in (3.12) so as discussed before, the QP sub-problem is a convex

[6] Equality constraints are always active but there is no equality constraints in this phase optimization problem.

optimization problem and has a unique global minimizer. This has been tested practically in the simulation results, when the $\mathbf{d}_k$ which minimizes a QP problem with specific setting is always identical, regardless of the initial guess.

The QP subproblem is solved by iterations when at each step the solution is given by $\mathbf{d}_{l+1} = \mathbf{d}_l + \alpha \acute{\mathbf{d}}_l$. An active set constraints at l_{th} iteration $\acute{A}_l$ is used to set a basis for a search direction $\acute{\mathbf{d}}_l$. This constitutes an estimate of the constraint boundaries at the solution point, and it is updated at each QP iteration. When a new constraint joins the active set, the dimension of the search space is reduced as expected.

The $\acute{\mathbf{d}}_l$ is the notation for the variable in the QP iteration; it is different from $\mathbf{d}_k$ in the major iteration of the SQP, but it has the same role which shows the direction to move towards the minimum. The search direction $\acute{\mathbf{d}}_l$ in each QP iteration is remaining on any active constraint boundaries while it is calculated to minimize the quadratic objective function. The possible subspace for $\acute{\mathbf{d}}_l$ is built from a basis $\mathbf{Z}_l$, whose columns are orthogonal to the active set $\acute{\mathbf{A}}_l$, $\acute{\mathbf{A}}_l \mathbf{Z}_l = 0$. Therefore, any linear combination of the $\mathbf{Z}_l$ columns constitutes a search direction, which is assured to remain on the boundaries of the active constraints.

The $\mathbf{Z}_l$ matrix is formed from the last $M - P$ columns of the QR decomposition of the matrix $\acute{\mathbf{A}}_l^T$ (3.13) and is given by; $\mathbf{Z}_l = \mathbf{Q}[:, P+1 : M]$. Here, P is the number of active constraints and M shows the number of design parameters in the optimization problem, which is the number of sub-blocks in the PAPR problem.

$$\mathbf{Q}^T \acute{\mathbf{A}}_l^T = \begin{bmatrix} R \\ 0 \end{bmatrix} . \tag{3.13}$$

The active constraints must be linearly independent, so the maximum number of possible independent equations is equal to the number of design variables, in other words $P < M$. For more details see [45].

Finally, there exists two possible situations when the search is terminated in QP subproblem and the minimum is found; either the step length in (3.14) is 1 or the optimum $\mathbf{d}_l$ is sought in the current subspace whose Lagrange multipliers are all positive.

3.2.4 SQP Pseudocode

Here, a psudocode is provided for the SQP implementation and we will refer to it in the complexity evaluation section. As discussed in the previous parts, the algorithm consists of two loops.
Step 0 Initialization of the variables before starting the SQP algorithm

- An extra element (slack variable) is appended to the variables so $\boldsymbol{\phi} = [\phi_0, \phi_1, \phi_2, \cdots, \phi_M]$. The objective function is defined as $f(\boldsymbol{\phi}) = \phi_M$ and is initialized with zero, other elements can be any random guess.
- The initial Hessian is an identity matrix $\mathbf{H}_0 = \mathbf{I}$, and the gradient of the objective function is $\nabla f(\boldsymbol{\phi}_K)^T = [0, 0, \cdots, 1]$.

Step1 Enter the major loop and repeat until the defined maximum number of iterations is exceeded.

- Calculate the objective function and constraints according to (2.10)
- Calculate the Jacobian matrix (3.4)
- Update the Hessian based on (3.9) and make sure it is positive definite.
- Call the QP algorithm to find $\mathbf{d}_k$

Step2 Initialization of the variables before starting the QP iterations,

- Find a feasible starting point for $\mathbf{d}_0 = [d_0^0, d_0^1, \cdots, d_0^M]$ and $\acute{\mathbf{d}}_0 = [\acute{d}_0^0, \acute{d}_0^1, \cdots, \acute{d}_0^M]$;
  Check that the constraints in the initial working set[7] are not dependent, otherwise find a new initial

[7] When it is not the first major iteration, the active set is not empty.

point $\mathbf{d}_0$ which satisfies this initial working set.

Calculate the initial constraints $\mathbf{A}\,\mathbf{d}_0 - \mathbf{a}$,

if $\max(constraints) > \epsilon$ **then**

The constraints are violated and the new $\mathbf{d}_0$ needs to be searched

end if

- Initialize the $\mathbf{Q}$, $\mathbf{R}$ and $\mathbf{Z}$ and compute initial projected gradient $\nabla q(\mathbf{d}_0)$ and initial search direction $\acute{\mathbf{d}}_0$

Step3 Enter the QP loop and repeat until the minimum is found

- Find the distance we can move in the search direction before a constraint is violated,

  $\mathbf{gsd} = \mathbf{A}\acute{\mathbf{d}}_l$ (Gradient with respect to the search direction)

  ind = find ($\text{gsd}_n >$ threshold)

  if isempty(ind) **then**

  Set the distance to the nearest constraint as zero and put $\alpha = 1$

  else

  Find the distance to the nearest constrain as follows

$$\alpha = \min_{1 \le n \le N} \left\{ \frac{-(A_n\,\mathbf{d}_l - a_n)}{A_n\,\acute{\mathbf{d}}_l} \right\}. \tag{3.14}$$

  Add the constraint A_i [8] to the active set $\acute{\mathbf{A}}_l$

  Decompose the active set as (3.13)

  Compute the subspace $\mathbf{Z}_l = \mathbf{Q}[:, P+1 : M]$

  end if

- Update $\mathbf{d}_{l+1} = \mathbf{d}_l + \alpha\acute{\mathbf{d}}_l$

- Calculate the gradient objective at this point $\nabla q(\mathbf{d}_l)$

- Check if the current solution is optimal[9]

  if $\alpha = 1$ || length($\acute{\mathbf{A}}_l$) $= M$ **then**

  Calculate the λ of active set by solving

$$-\,\mathbf{R}_l\,\lambda_l = (\mathbf{Q}_l^T * \nabla q(\mathbf{d}_l))\,. \tag{3.15}$$

  end if

  if all $\lambda_i > 0$ **then**

  return $\mathbf{d}_k$

  else

  Remove the constraints with $\lambda_i < 0$

  end if

- Compute the QP search direction according to the Newton step criteria,

$$\acute{\mathbf{d}}_l = -\mathbf{Z}_l \left((\mathbf{Z}_l^T\,\mathbf{H}_k\,\mathbf{Z}_l) \setminus (\mathbf{Z}_l^T \nabla\, q(\mathbf{d}_l)) \right), \tag{3.16}$$

  Where the $(\mathbf{Z}_l^T\,\mathbf{H}_k\,\mathbf{Z}_l)$ is projected Hessian, see A on Page 44.

[8] Where i is the index of minimum in (3.14) which indicates the active constraint to be added.

[9] The term "length" indicates the number of rows in $\mathbf{A}_l$ or equivalently the number of active constraints.

Step4 Update the solution $\boldsymbol{\phi}$ for the k_{th} iteration; $\boldsymbol{\phi}_{k+1} = \boldsymbol{\phi}_k + \mathbf{d}_k$ and go back to Step 1

To summarize, The SQP algorithm as its name indicates is one of the iterative optimization techniques. The initial solution of the problem moves towards the directional derivative which is found by solving the QP subproblem. In the QP algorithm the solution sought by taking the Newton step within subspace indicated by active constraints, that is the constraints define the boundary of solution space. The minimum is found when the gradient of the quadratic objective function became zero. The quadratic function is defined in a way that shows the direction of the greatest decrease in the original objective function at current solution point.

Chapter 4

Simulation Results

In this part we explore Matlab simulation results of discussed methods in the previous chapters, here most results represented as CCDF of PAPR which studied in Section 1.1.2. We consider 1024 OFDM with 840 useful data subcarriers and 64 QAM modulation which is oversampled by 4 times. Pilot subcarriers are set as known value 1 in the whole simulations.

4.1 PSO Simulation Results

As reviewed before, the PSO is one of the representation of evolutionary algorithms that is used to find the optimum phase weights in the PAPR reduction problem, but it is expensive to implement when the number of sub-blocks are large. The parameters for PSO simulation in Section 3.1 adjusted for the best result by empirical observation; c_1 and c_2 are 0.25 and r_1 and r_2 set to 1.

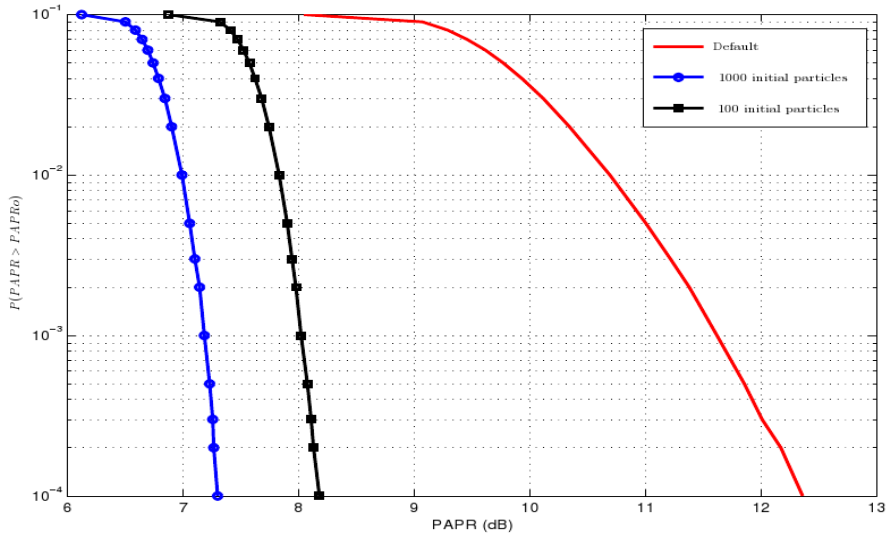

Figure 4.1: PAPR reduction performance for PSO algorithm and 60 clusters

PAPR reduction performance of PSO with different initial particles, is shown in Figure 4.1 with CCDF plots. The default curve shows the original PAPR of the system with no signal processing algorithm. By employing the PSO algorithm to find the appropriate phase weights of PTS sub-blocks, PAPR reduction gain of 4.1dB achieved for 100 initial particles at probability of 10^{-4}. In Section 5.2.1, we will see that

the complexity grows linearly with the number of initial particles, but in terms of performance the PAPR decreased about 0.7dB when the initial particles are increased for 10 times.

4.2 SQP Simulation Results

The proposed PAPR reduction technique for an OFDMA system with 1024 subcarriers and 64 QAM modulation is simulated for a WiMAX data structure as shown in Figure 2.5. There is an illustration of the SQP implementation to see the progress in the algorithm in each major iteration, it gives an impression of how many QP iterations are needed to form one major step. This is a brief description of Matlab implementation; assume that there are 60 sub-blocks so the phase vector is $\mathbf{x} = [x_0, x_1, x_2, \cdots, x_{59}]$, now an extra element is appended so there is $\mathbf{x} = [x_0, x_1, x_2, \cdots, x_{60}]$ and the objective function is defined as $f(\mathbf{x}) = x_{60}$, the x_{60} initialized with zero and that is why it is equal to zero in the 0_{th} iteration in Figure.4.2.

In the QP subproblem the objective is $q(\mathbf{d}) = 1/2\,\mathbf{d}^T\,\mathbf{H}_k\,\mathbf{d} + \nabla f(\mathbf{x}_K)^T\,\mathbf{d}$, and the initial feasible point in the first iteration is set to $\mathbf{d}_0 = [0, 0, \cdots, \max(F_i(\mathbf{x}_0))]$. So $\nabla f(\mathbf{x}_K)^T = [0, 0, \cdots, 1]$ and $\nabla f(\mathbf{x}_K)^T\,\mathbf{d} = \max(F_i(\mathbf{x}_0))$ which is the maximum value, since $1/2\,\mathbf{d}^T\,\mathbf{H}_k\,\mathbf{d}$ is zero then $q(\mathbf{d}_0) = \max(G_i(\mathbf{x}_0))$. Once $\mathbf{d}_k$ which minimize $q(\mathbf{d})$ is found, then $\mathbf{x}$ in the SQP problem is updated as $\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{d}_k$ so $f(\mathbf{x}) = x_{60}$ as can be noticed in below process.

Iter	F-count	Objective value	Max constraint	Number of QP iters	Directional derivative	Procedure
0	1	0	80.3263			
1	3	41.94	1.977	40	41.9	
2	5	35.56	1.111	110	-6.39	Hessian modified
3	7	33.77	0.1087	94	-1.79	Hessian modified
4	9	32.3	0.204	126	-1.47	Hessian modified
5	11	31.58	-0.0196	151	-0.719	Hessian modified twice
6	13	30.8	0.1585	131	-0.783	Hessian modified
7	15	30.35	0.1387	217	-0.444	Hessian modified
8	17	30.12	0.03673	156	-0.232	Hessian modified
9	19	29.97	0.01005	69	-0.155	Hessian modified
10	21	29.9	0.01257	22	-0.0706	Hessian modified

Figure 4.2: SQP algorithm progress

In the 0_{th} iteration, there is no QP solving and it handles the problem formulation as minimax so the constraints are evaluated at initial point and the maximum is pointed. From 1_{th} iteration, the QP subproblem is solved each time and the number of inner iterations is given in `Number of QP iters` column.
`Objective value` indicates the objective function (maximum) in the original problem evaluated at the current solution. As can be noticed, it is reduced in each iteration, this is the value which gives the PAPR of system;[1]. $\texttt{Objective value}_k = \texttt{Objective value}_{k-1} + \texttt{Directional derivative}_k$

To have a better perception of the PAPR cost function, a 3D plot is provided in Figure 4.3, which illustrates the variation of the PAPR, or equivalently the maximum amplitude of one OFDM symbol partitioned into two disjoint sub-blocks, versus two phase coefficients. Predictably, two sub-blocks can not do much for the PAPR reduction purpose and this is just to give a visual impression of the cost function to be minimized in the SQP optimization algorithm.

As can be seen, there are many local minima which have slightly different levels; that is one of the promising properties of this optimization problem because reaching a local minimum satisfies the PAPR

[1]Since in our proposed technique, the phase of clusters are changed and not the amplitude of signal, therefore the average of the OFDM symbol (denominator in (1.3)) remains intact during the PAPR reduction process and the largest subcarrier defines the PAPR value.

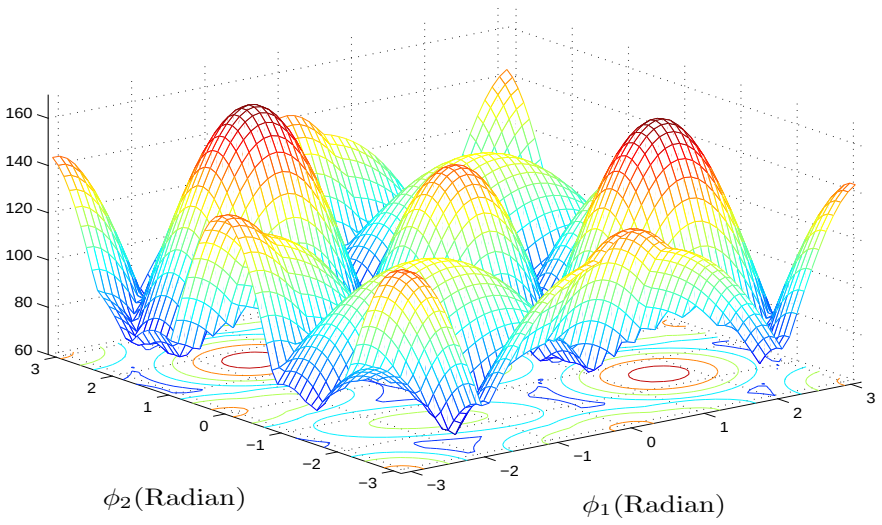

Figure 4.3: The 3 dimensional power cost function for an arbitrary OFDM symbol with 2 phase coefficients.

reduction aim even though the global minimum is not found. As a result, the performance of the proposed algorithm is relatively insensitive to the initialization of the optimization.

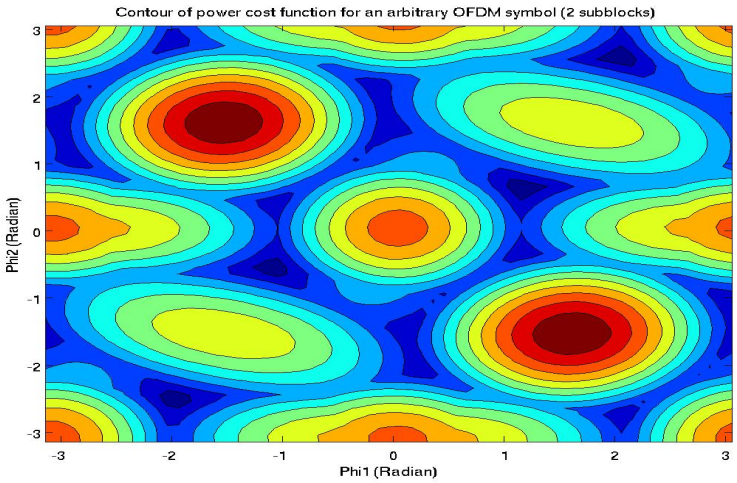

Figure 4.4: The contour plane for 3D cost function for an arbitrary OFDM symbol with 2 phase coefficients, dark blue areas are minimums.

4.2.1 Performance of Different Algorithms

The performance of four different optimization techniques is illustrated in Figure 4.5 by CCDF curves. Once the Jacobian of the cost function is defined, the optimization problem can be treated with different optimization methods.

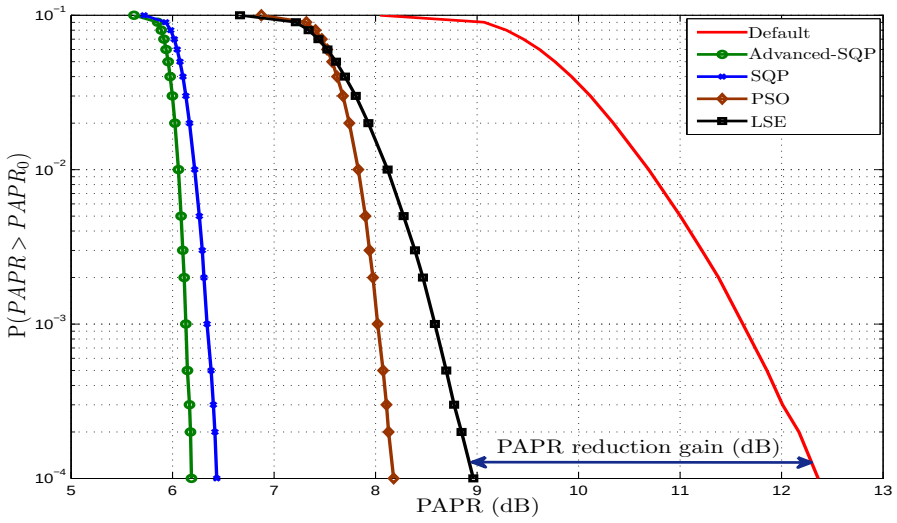

Figure 4.5: The comparison between CCDF curves of different PAPR reduction algorithms, advanced-SQP outperforms other methods with 6.2dB gain while the LSE gives the least PAPR reduction gain of 3.4dB.

The SQP is the best solution for the problem in terms of PAPR reduction performance, but the Least Square Error (LSE) approach can also be used to reduce the peak amplitude of the signal with much less complexity. However, the performance is not as good as the SQP algorithm but still comparable with existing PAPR reduction techniques [7].

The LSE algorithm minimizes the objective function $f(x) = (f_1(x))^2 + (f_2(x))^2 + \cdots + (f_N(x))^2$, which is the sum of the OFDM subcarriers amplitudes [2]. The components try to be equal to minimize the sum, so the large samples are pushed to a specific level while the smaller ones become larger.

As discussed befor, one of the examined optimization methods to search the phase coefficients in PTS is Particle Swarm Optimization (PSO) [25]. This is a method for performing numerical optimization when there is no definite knowledge of the gradient of the problem to be optimized. The achieved gain for PSO is better than LSE, but it is expensive to implement especially when the number of sub-blocks is large.

If the search for the global minimum can be performed in each OFDM symbol then the CCDF curve improves to some degree. in our test, each OFDM symbol has been processed 100 times with different initial guesses and the one with the smallest PAPR is selected. The result in Figure 4.5 (Advanced SQP) shows an overall improvement of about 0.5dB. In this case, the PAPR of the system can roughly be considered as a deterministic value since the CCDF curve is almost vertical.

4.2.2 Parameters in Performance

Figure 4.6 shows the performance of the SQP algorithm at the point $\Pr\{\text{PAPR} > \text{PAPR}_0\} = 10^{-4}$ for 10000 random OFDM symbol with 64 QAM modulation versus different number of major iterations. The vertical axis represents the PAPR reduction gain in dB, which is the difference between the original CCDF curve and the processed signal curve at the specified probability, as indicated in Figure 4.5. As noticed here, most of the job is done in the first iteration and for more than 10 iterations the progress tends to be slower as it reaches a local minimum.

[2] This is the simplest scenario, but other modifications can be made to develop a more elaborate version of LSE.

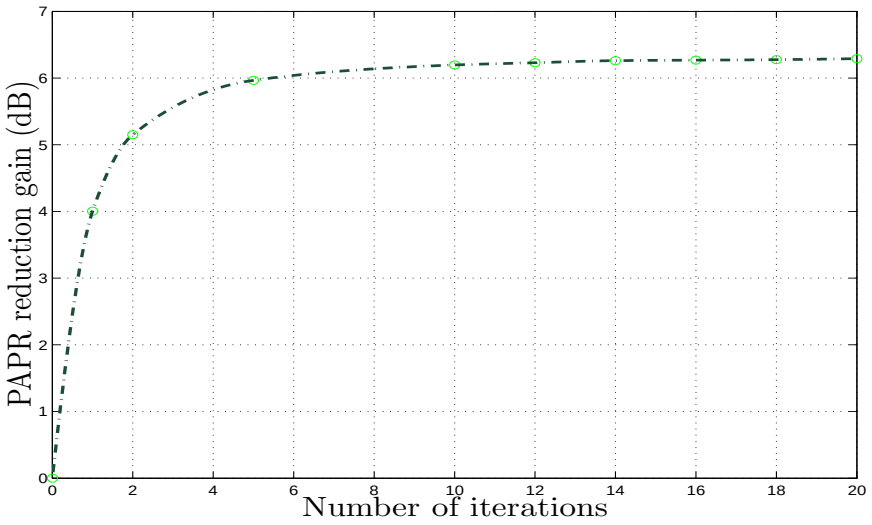

Figure 4.6: SQP PAPR reduction gain versus number of major iterations when $\Pr\{\text{PAPR} > \text{PAPR}_0\} = 10^{-4}$.

Figure 4.7 shows the PAPR reduction degradation, when the number of sub-blocks are reduced. As explained before, each cluster can be phase rotated and this will be reversed at the receiver in the channel equalization process. To bring down the complexity, the same phase coefficients are assigned to several adjacent clusters to simplify the optimization algorithm. In fact, 30 sub-blocks means 2 clusters within one sub-block and each sub-block is weighted with specific phase coefficient. In practice, there can not be 120 phase coefficients or sub-blocks, because it means that one cluster has two phase weights and this is not possible to compensate at the receiver according to the WiMAX standard. But in Figure 4.7, a 120 sub-blocks configuration is simulated to show the trend of PAPR reduction gain versus the number of sub-blocks.

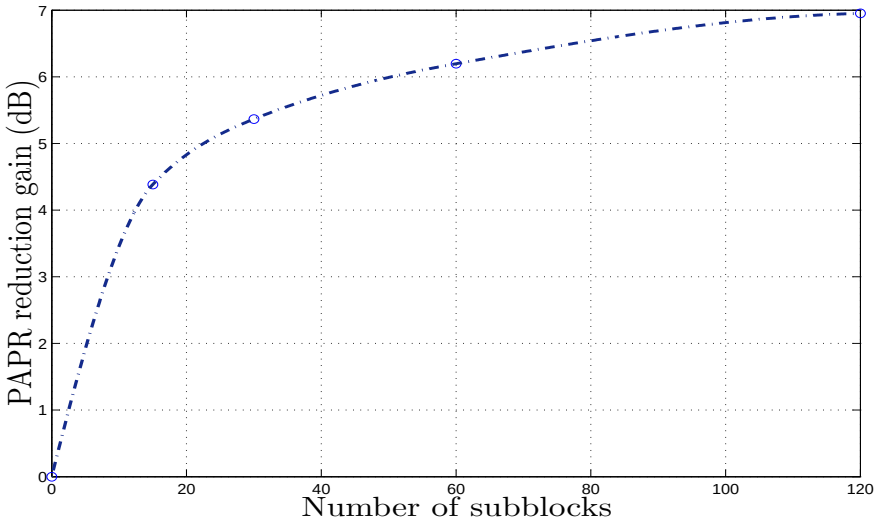

Figure 4.7: SQP PAPR reduction performance versus number of sub-blocks when $\Pr\{\text{PAPR} > \text{PAPR}_0\} = 10^{-4}$

The power of two 1024 ODFM symbol in time is illustrated in Figure 4.8, before and after the signal processing algorithm. The benefit is clear from this graph, that the back off margin can be much smaller and the misperformance discussed in [17] for the ACE method, does not exist for the SQP technique.

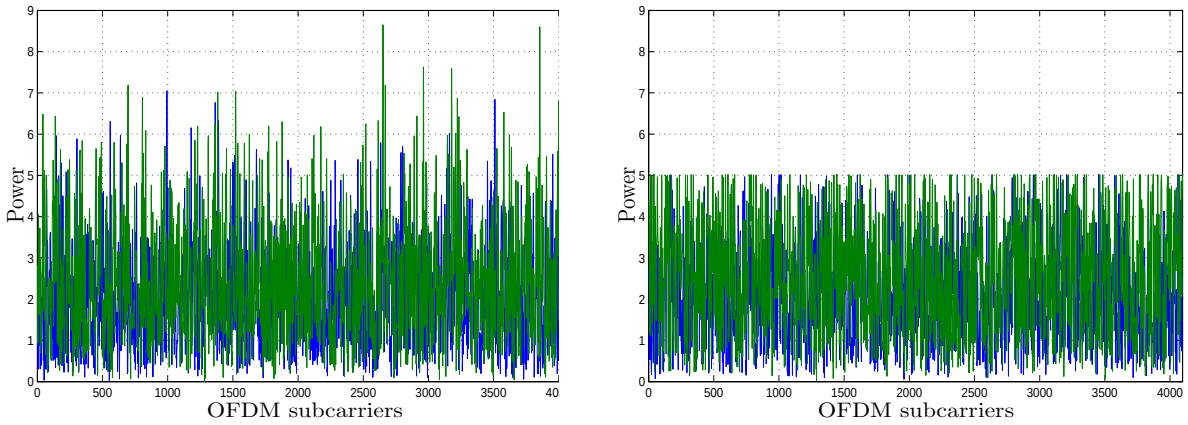

Figure 4.8: The comparison between the time domain OFDM symbol before and after the PAPR reduction procedure for 60 clusters and 10 iterations.

Finally, the PAPR reduction performance in terms of CCDF curve is not changed with different initial guesses, because the maximum of all 10000 simulated OFDM symbols defines the CCDF curve in low probability of $\Pr\{\text{PAPR} > \text{PAPR}_0\}$, and this does not depend on the initial solution. But in each OFDM symbol the minimum can be found by examination of various starting points and the performance can be improved as Figure 4.5 illustrates in advanced-SQP curve.

Chapter 5

Implementation Issues

In the first section, we introduce a method to reduce the complexity of PTS technique based on sub-block partitioning in [42], and then we try to give a measure of complexity for different techniques and compare them in terms of operations needed for each algorithm.

5.1 IFFT block complexity

We discussed the PTS technique in Section 2.1.6 and as mentioned, the complexity of PTS technique is increased exponentially with the number of sub-blocks. In PTS, each sub-block has to be modulated by IFFT independently. The number of complex addition and multiplication required to modulate a sub-block is given as,

$$n_{mul} = N/2\, log_2\, N; \qquad n_{add} = N\, log_2\, N; \tag{5.1}$$

Hence, the total computational complexity to transmit an OFDM symbol is calculated by multiplication of (5.1) by the number of sub-blocks M [44]. It can be verified that the complexity has been increased significantly comparing to the OFDM system without PTS algorithm. Reduction of computational complexity can be achieved by using Cooley-Tukey FFT algorithm.

5.1.1 Cooley-Tukey FFT algorithm

Cooley-Tukey FFT algorithm is the most common fast Fourier transform (FFT) algorithm. It re-expresses the discrete Fourier transform (DFT) of an arbitrary composite size $N = N_1 N_2$ recursively, in terms of smaller DFTs of sizes N_1 and N_2, in order to reduce the computation time to $O(N\, log\, N)$ for highly-composite N (smooth numbers).

One of the well known variant of Cooley-Tukey algorithm is a radix-2 decimation-in-time (DIT) FFT, a DFT of size N is divided into two interleaved DFTs (hence the name is "radix-2") of size $N/2$ with each recursive stage. As a result, to calculate the DFT of size N based on the original definition, N^2 complex multiplication required where the implementation of DFT based on Cooley-Tukey algorithm only needs $N/2\, log_2\, N$ multiplication, for example in DFT of size 1024, there can be saved 200 times less multiplication [44].

The radix-2 version of Cooley-Tukey is used almost every where and almost all DSPs use it for DFT implementation. This is not what we want to utilize it in the PTS to reduce the computational complexity, we are mostly interested in a general factorization of algorithm [43].More generally, Cooley-Tukey algorithms recursively calculate a DFT of a composite size $N = N_1 N_2$ as,

1. Perform $N1$ DFTs of size N_2.

2. Multiply by complex roots of unity called twiddle factors.

3. Perform N_2 DFTs of size N_1.

Therefore the DFT of size N can be calculated in three steps instead of one, and the results and complexity are identical. In other words, first apply DFT along rows and when twiddle factors multiplied by outputs, apply another DFT along columns.

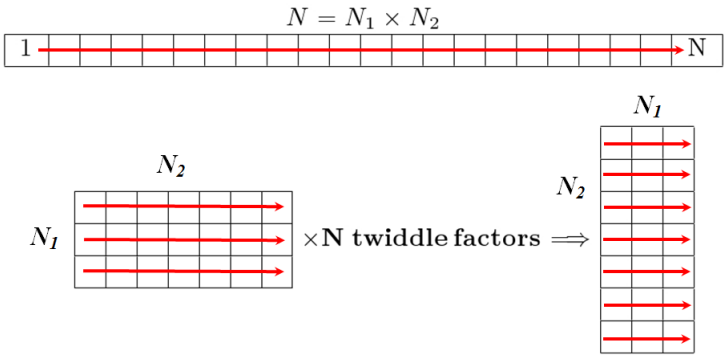

Figure 5.1: The stages to perform Cooley-Tukey algorithm with general factorization, 1D DFT of size N is equal to 2D DFT of size $N_1 \times N_2$.

The output of algorithm is formulated here to have a mathematical expression of the result; assume $n_1 = 0, 1, 2, \cdots, N_1 - 1$ and $n_2 = 0, 1, 2, \cdots, N_2 - 1$, also k_1 and k_2 respectively indicate the row and column of an arbitrary element in the table. For example, if the original DFT block has 21 elements as shown in Figure 5.1 and it is divided to $N1 \times N2 = 3 \times 7$ then the element x_{12} in the original block is $x_{1,5}$ in the table. Note that the number of rows and columns starts with 0 so for this example, $k_1 = 1$ and $k_2 = 5$.

$$\begin{aligned} X_{N_2 k_1 + k_2} &= \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} x_{N_1 n_2 + n_1} e^{-\frac{2\pi i}{N_1 N_2} \cdot (N_1 n_2 + n_1) \cdot (N_2 k_1 + k_2)} \\ &= \sum_{n_1=0}^{N_1-1} \underbrace{\left[e^{-\frac{2\pi i}{N} n_1 k_2} \right]}_{\text{twiddle factor}} \underbrace{\left(\sum_{n_2=0}^{N_2-1} x_{N_1 n_2 + n_1} e^{-\frac{2\pi i}{N_2} n_2 k_2} \right)}_{\text{DFT of size } N_2} e^{-\frac{2\pi i}{N_1} n_1 k_1}. \end{aligned} \tag{5.2}$$

5.1.2 PTS with Cooley-Tukey IFFT

As one can see in Figure 2.4, there are many zeros in PTS sub-blocks and since the IFFT operation is summation, then zeros are an ineffective part of operation and we can do better than performing 1024 IFFT on each of them. The trick is, for each PTS sub-block of size 1024, a table is built according to Figure 5.2, in a way that the data information part is in one row and other rows are zeros. The table has $M = 14$ columns and $L = 60$ rows which $60 \times 14 = 840$ are the number of useful subcarriers in WiMAX OFDM.

Now there should be some compensation for those zero values in the first frequency guard subcarriers in WiMAX OFDM, see Figure 2.5. In this case, there is another extra phase factor to multiply by IFFT outputs rather than twiddle factor, that is the IFFT output of $x_{p,q}$ in p_{th} row and q_{th} column, $W_N = e^{j2\pi/N}$ ($N = 1024,\ M = 14,\ L = 60$ and $g = 92$).

$$x_{p,q} = \frac{1}{N} W_N^g \sum_{l=0}^{L-1} \left[W_N^{lq} \right] \left(\sum_{m=0}^{M-1} X_{l,m} W_M^{mq} \right) W_L^{lp}. \tag{5.3}$$

$M \rightarrow$

$L \downarrow$

0	0	0	...	0
⋮	⋮	⋮	⋱	⋮
0	0	0	...	0
$X_{l,0}$	$X_{l,1}$	$X_{l,2}$	⋯	$X_{l,M-1}$
0	0	0	...	0
⋮	⋮	⋮	⋱	⋮
0	0	0	...	0

Figure 5.2: Except for l_{th} row which contains data subcarriers, other rows are zero. Consider $p, l = 0, 1, 2, \cdots, L-1$ and $q, m = 0, 1, 2, \cdots, M-1$.

According to the table in Figure 5.2, applying the three step Cooley-Tukey algorithm and taking IFFT from rows, gives zero result in rows filled with zeros and only l_{th} row returns non-zero output. After first step the table became,

0	0	0	...	0
⋮	⋮	⋮	⋱	⋮
0	0	0	...	0
$\acute{x}_{l,0}$	$\acute{x}_{l,1}$	$\acute{x}_{l,2}$	⋯	$\acute{x}_{l,M-1}$
0	0	0	...	0
⋮	⋮	⋮	⋱	⋮
0	0	0	...	0

Figure 5.3: The output of table in Figure 5.2 after taking IFFT in rows of table.

After twiddle factor multiplication that still does not change the null rows, in third step, IFFT is taken along columns and the final result is formulated as,

$$x_{p,q} = \frac{1}{N} \left[W_N^g\right] \left[W_N^{(lq)}\right] \left[W_L^{(lp)}\right] \acute{x}_{l,q} = \frac{1}{N} \left[W_N^{(lq+g)}\right] \left[W_L^{(lp)}\right] \acute{x}_{l,q}. \tag{5.4}$$

In this case, each sub-block can be modulated with only one M-point IFFT plus N-times complex multiplications. The number of complex computations required is,

$$n_{mul} = M/2\, log_2\, M + N; \qquad n_{add} = M\, log_2\, M; \tag{5.5}$$

Comparing this result with (5.1), the new arrangement gives and impressive computational complexity reduction. This result is very important in the new WiMAX PAPR reduction problem setting, because in the conventional PTS the number of sub-blocks are small (up to 8), but in this case there are 60 sub-blocks, so the amount of operations in IFFT block affects the total complexity of the proposed technique.

5.2 Complexity analyses

In this part, the computational complexity of the PSO and SQP algorithms are analyzed. These are methods to find the best phase coefficients in the PTS sub-blocks, so it is needed to separate the computation in the proposed technique before the optimization step which is common for both algorithms, and then compare the computational complexity only in the phase optimization part.

Here is a review of the steps taken to implement the proposed technique, N is the number of OFDM subcarriers, M is the number of PTS sub-blocks [1]

- Compute matrix $\mathbf{B}$ in (2.7) using discussed method in the previous section, requires $(M \times (\acute{M}/2) \, log_2 \acute{M} + N)$ multiplication and $(M \times \acute{M} \, log_2 \acute{M})$ addition.
- Phase optimization
- Compute $\mathbf{s}$ in (2.10) which requires $(N \times M)$ complex multiplication and addition.

So by specification of the second step, the total computational complexity would be determined. For example, in conventional PTS the optimal phase set is selected among limited number of phases, by doing exhaustive search. If we assume the number of allowed phases is K and the number of sub-blocks is M then the total possible combinations to search would be $K^{(M-1)}$ when the first phase sequence, is set to zeros.

5.2.1 PSO

Recall Section 3.1.1 for complexity computation in particle swarm optimization, the complexity depends on the number of initial particles n. Here is the complexity for each iteration of the PSO algorithm,

- Compute the PAPR of initial particles: for each particle $(N \times M)$ complex addition and multiplication required and for n particles ,there is $(n \times N \times M)$. Then for each particle, to calculate PAPR search will be done to find the maximum, among N outputs.
- Updating velocity and positions: refer to (3.1) and (3.2), total $(n \times M)$ complex multiplication and $(3 \times n \times M)$ complex addition is required for each update.

Hence if the algorithm runs for k iterations, we can formulate PSO complexity as :

$$\begin{aligned} n_{mult} &= k\, n\, (N \times M) \\ n_{add} &= k\, n\, (N \times (M+1)) \\ n_{comp} &= k\, n\, N \end{aligned} \tag{5.6}$$

5.2.2 SQP

The SQP algorithm has a quite complicated mathematical concept, and it can be implemented with different modifications. Hence, the complexity evaluation is not straightforward. The number of QP iterations is not fixed[2] and is different for each OFDM symbol; here the average number of QP iterations is considered to evaluate the complexity. For 60 sub-blocks, 1024 subcarriers and 64 QAM, the average is obtained as 80 iterations for each major SQP iteration.

Another difficulty to compute the required operation, is the length of the active set, which alters during iterations starting from 1 to at most M at the end of loop. Consequently, the size of $\mathbf{R}$ in the QR decomposition and $\mathbf{Z}$ the basis for the search subspace are not fixed during the process so the complexity can not be assessed directly for each QP iteration and some numerical estimations are necessary.

To evaluate the amount of computation needed for this technique, all steps in the psudocode are reviewed in detail and an explicit expression is given for each part. First, the complexity of the major loop is assessed in Steps 1 and 4, and then the QP loop is evaluated separately. Finally, the complexity is derived in terms of the number of sub-blocks and major iterations with some approximation and numerical analysis.

[1] Be careful that this M (like the whole literature) denotes the number of clusters or PTS sub-blocks and is different from the one in Section 5.1.2, so the M in the previous section is replaced with $\acute{M} = \frac{(N-2g)}{M}$.

[2] The QP is a convex optimization problem, so the iterations proceed till the optimum is found, but a modification of the algorithm can be used when the number of iterations are fixed.

Major loop

Steps 1 & 4

1. Objective function and constraints from (2.10):
   $4M \times N$ multiplications and the same amount of addition, N comparisons to find the maximum of constraints

2. Jacobian matrix from (3.4):
   $6M \times N$ multiplications, $4M \times N$ additions

3. Hessian update (3.9):
   $2M \times N$ multiplications, $2M \times (N+1)$ additions to calculate (3.11),
   $3(M+1)$ additions and M multiplications for matrices of size $M \times 1$ to compute $\mathbf{q}_k$ and $\mathbf{q}_k$,
   $2M$ divisions and M additions is required to update $\mathbf{H}$

4. The solution $\boldsymbol{\phi}$ is updated, which requires M additions.

QP loop

Step3

1. Gradient with respect to the search direction:
   $4M \times N$ multiplications and additions to calculate **gsd** , N comparisons to find the maximum

2. Distance to the nearest constraint from (3.14):
   $2M \times N$ multiplications and additions, N comparisons to find the minimum

3. Addition of constraint to the active set:
   Assume the active set has length $L-1$, then the new constraint is inserted and the matrix size becomes $M \times L$. To compute the QR decomposition of this matrix, $2L^2(M - L/3)$ operations are needed [30].

4. Update the solution d_l which needs M additions.

5. The gradient objective at the new solution point needs M^2 multiplications and $M^2 + 1$ additions

6. The Lagrange multipliers are obtained by solving a linear system of equations, and this impose a complexity in the order of M^3 [30].

7. Remove the constraint in case of $\lambda_i < 0$:
   Removing the constraint and recalculation of QR decomposition requires $2L^2(M - L/3)$ operations.

8. Search direction according to (3.16):
   It is a solution to a system of linear equations. The size of $\mathbf{Z}$ varies during the iterations, and starts from $M \times M$ and reduces to an $M \times 1$ matrix at the end. Accordingly, the complexity in a QP iteration can be stated as $2S^2(M + S/3)$ where S is the number of columns in $\mathbf{Z}$ at each step.

At first, the computation which is required for the major loop is obtained as $22NM + 9M + N$. Next, the amount of computation in the QP loop is divided into fixed and variable parts[3]; there are $(6M + 2)N + 2M^2 + M$ operations which are performed in parts numerated by $1, 2, 4$ and 5 in every iterations. Besides there are amount of uncertain operations in other parts which are evaluated separately.

To resolve the search direction in (3.16) two states is possible; the first M times needs $0.4167M^4 + 0.6667M^3 + 0.25M^2$ operations, which is derived by numerical analysis and polynomial fitting, and for further iterations each needs $2M$ operations. Therefore the required number of flops can be approximated as

[3]The fixed operations belong to those matrices whose sizes do not change during the iterations while there are other matrices like $\mathbf{Z}$ that has variable size and hence different complexity during iterations.

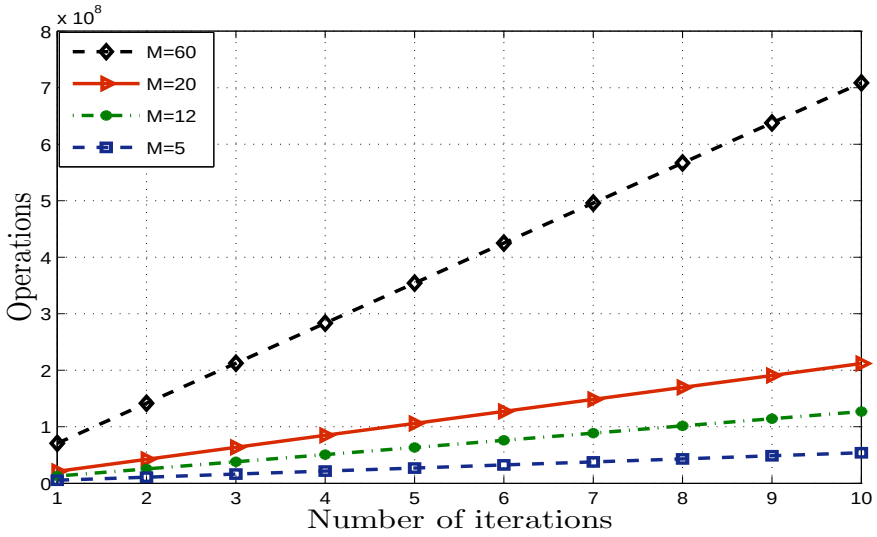

Figure 5.4: Complexity of the SQP algorithm versus the number of iterations for different number of sub-blocks.

$0.4M^3 + 0.7M^2 + 0.2M$ for each iteration. In the QR decomposition part, which is certainly done in every iterations, the procedure is the same. It means that for the first M iteration, $0.25M^4 - 0.3333M^3 + 0.0833M^2$ operations and for the extra ones $4/3M^3$ flops are done. So the amount of major computation is approximated to be $0.25M^3$ for each QP iteration by dividing the total operations over M.

With an acceptable approximation, we claim that the Lagrange multipliers calculation can be neglected in comparison with other dominant parts of computations, because it mostly appears after $M+1$ iterations; this occurs when the active constraints are full (M constraints are added to the active set), or sometimes when the exact step to the minimum is found. To sum up, the total number of operations needed for each QP iteration is roughly expressed as $0.65M^3 + 2.7M^2 + 6NM + 2N$, and the total complexity is shown in (5.7), where k and l are the number of major and QP iterations respectively.

$$\begin{aligned} O \quad = \Big(& (0.65M^3 + 2.7M^2 + 6NM + 2N) \times l \\ & + (22NM + 9M + N) \Big) \times k \,. \end{aligned} \tag{5.7}$$

The complexity in above the expression is illustrated as graphs, which show the number of operations in the SQP algorithm for two OFDM symbols in time with 1024 subcarriers. Figure 5.4 indicates the trend when the number of iterations increases. Predictably, when more sub-blocks are chosen to be phase rotated, then the complexity is raised with sharper slope versus the number of iterations because M^3 is the coefficient which dominantly defines the slope of l. Figure 5.5 shows how the complexity grows almost linearly with the number of sub-blocks for less number of iterations, while it tends to a cubic curve for larger number of iterations.

The exhaustive search approach, which is used in conventional PTS, costs significantly higher compared to the proposed algorithm. Moreover, the performance is not as good as SQP, since the phase coefficients are optimized among a quantized phase set. The whole calculation in (2.7) has to be repeated for every combination of phase vectors, and this requires $K^M \times MN$ times additions and multiplications where K is the number of allowed phases and M is the number of sub-blocks. Additionally, $K^M \times (N+1)$ comparisons are needed to find the largest sample among each produced transmit sequence, and also between all PAPRs to choose the minimum.

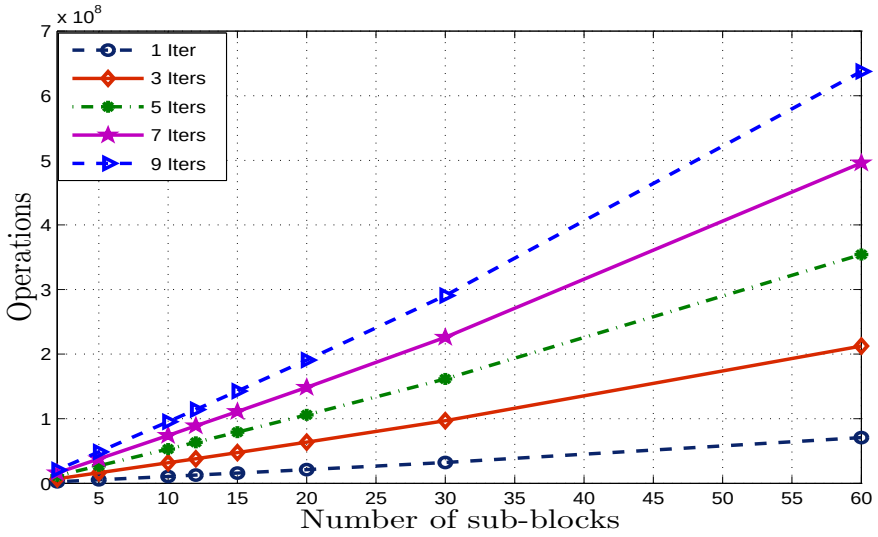

Figure 5.5: Complexity of the SQP algorithm versus the number of sub-blocks for different number of iterations.

Another thing to note is, the complexity evaluation using the number of operations does not give realistic impression for iterative algorithms. Specially, since the algorithm is running in real time application, the time cost would be more descriptive than the number of operations and that is, although the number of operations for PSO algorithm is almost comparable with SQP, the number of iterations in PSO and consequently time is much higher for acceptable performance. In order to compare the complexity for reasonable parameters the results in Figure 5.2.2 is calculated,

Algorithm	Operations
Exhaustive search	$2K^M \times MN$
PSO	$2kn \times (M+1)N$
SQP	$k \times \Big((0.65M^3 + 2.7M^2 + 6NM + 2N) \times l + (22NM + 9M + N)\Big)$

Figure 5.6: The complexity of different algorithms to search optimum phase set in PTS

To be more explicit, the notation used in the table explained below,

- k is the number of iterations.
- l is the number of QP iterations in SQP.
- N is the number of subcarriers in one OFDM symbol.
- K is the number of allowed phase for conventional PTS exhaustive search.
- M is the number of sub-blocks.
- n is the number of initial particles for PSO.

To have a better perception of the PTS complexity, assume the allowed phase set is ± 1, so $K = 2$ and no phase rotation required. Also, the number of sub-blocks is $M = 60$ and the same setting preserved as the SQP; then approximately, 10^{23} additions and 10^{21} comparisons have to be performed to find the optimum phase. In contrast, the SQP requires 10^8 flops which is roughly equivalent to the PTS exhaustive search with only 12 sub-blocks and 2 phase options. Also PSO requires initial particle number of $n = 100$ and $k = 50$

PSO iteration to have almost the same performance as SQP which gives the complexity of order 10^9 with effectively more iterations, when this iterations can not be performed in parallel and this is not acceptable a for real time application.

Chapter 6

Conclusion Remarks

In this thesis, a novel solution for the PAPR reduction in WiMAX systems has been introduced based on restricting the whole signal processing procedure in the base station, and this means; the job has to be done in the transmitter side. By this criteria, possible options of PAPR reduction techniques became more narrow. Taking advantage of the embedded potentials in WiMAX standard, a new method has been developed without sending any information to the receiver to process. That is changing the phase of clusters in the WiMAX data structure in a way that the total PAPR of the OFDM symbol is reduced ,on the other side, the receiver inverses the phase change by aids of channel equalization techniques which is already performed in the receiver to compensate for channel imperfections.

By exploration of the possible choices to implement this idea, PTS was found to be the most promising method by dividing the whole OFDM subcarriers into disjoint sub-blocks and choosing the best phase weights to multiply by sub-blocks. Since the phase set applied in the new technique is not quantized unlike the conventional PTS, and also the number of sub-blocks are quite large, the exhaustive search is not applicable here, so the problem is formulated as the optimization of phase vector.
Searching for existed minimization techniques, the particle swarm approach is a feasible point to start. The algorithm was implemented in Matlab and the results has been illustrated in simulation chapter. High computational complexity of PSO was a serious issue and it gets worse as the number of sub-blocks increases.

A new method proposed to search the optimum phase coefficients for the PTS sub-blocks by means of non-linear optimization technique. The objective function for the PAPR reduction purpose has been defined and the gradient of this function is calculated analytically to take more intelligent steps in the phase optimization process, rather than using the evolutionary algorithms without having knowledge about the gradient of the cost function.
The SQP algorithm is chosen to solve the minimization problem and details are adjusted for this particular application. Complexity calculation of the algorithm is done and an approximation has been given in Chapter 5 to see the implementation cost.

Using the built-in functions in Matlab to program the algorithm, makes it very fast in terms of time but the actual cost of this technique, when it is implemented in real DSP kits, should be evaluated which is not in the scope of this thesis.
Note, SQP algorithm is not the only way to solve this problem and other approaches can be used to achieve less expensive algorithm that works on the gradient base concept, but we can say the SQP is almost the best in terms of performance and it finds the optimum phase coefficient. In other words, the best result with changing the phase of clusters and without increasing the average transmitted power can be achieved by the SQP algorithm.

Finally, The complexity issues regarding IFFT box which has a main role in OFDM modulation, and it

becomes more important when it comes to the PTS technique, discussed. And an smart resolution introduced to effectively reduce the computations in the IFFT part based on the Cooley-Tukey FFT algorithm. Moreover, a table of complexity evaluation is presented which gives an explicit information about different phase search algorithms. The SQP algorithm needs less computation to perform, comparing other methods while it also gives better results.
Although the SQP algorithm works effectively with great PAPR reduction gain, other gradient based optimization techniques can be developed with lower complexity; LSE is presented in the simulation results, which can be run with much less computations. Also, different modifications of SQP can be derived to simplify the algorithm, and probably can perform the minimization in parallel, which is more suitable for hardware implementation.

Appendix A

Calculation of the search direction $\acute{\mathbf{d}}_l$

The procedure of deriving search direction of the QP is explained in [26], which is brought here for convenience. Once $\mathbf{Z}_l$ is derived, a new search direction $\acute{\mathbf{d}}_l$ is updated that minimizes the QP objective function $q(\mathbf{d})$, which is a linear combination of the columns of $\mathbf{Z}_l$ and is located in the null space of the active constraints. Thus, the quadratic objective function can be reformulated as a function of some vector $\mathbf{b}$ by substituting for $\acute{\mathbf{d}}_l = \mathbf{Z}_l\,\mathbf{b}$, in general QP problem.

$$q(\mathbf{b}) = \frac{1}{2}\,\mathbf{b}^T\,\mathbf{Z}_l^T\,\mathbf{H}\,\mathbf{Z}_l\,\mathbf{b} + \mathbf{c}^T\,\mathbf{Z}_l\,\mathbf{b}\,, \tag{A.1}$$

Differentiating with respect to $\mathbf{b}$ yields, $\nabla q(\mathbf{b}) = \mathbf{Z}_l^T\,\mathbf{H}\,\mathbf{Z}_l\,\mathbf{b}^T + \mathbf{Z}_l^T\,\mathbf{c}$ where $\nabla q(\mathbf{b})$ is referred to as the projected gradient of the quadratic function, because it is the gradient projected to the subspace defined by $\mathbf{Z}_l$. The minimum of the function $q(\mathbf{b})$ in the subspace defined by $\mathbf{Z}_l$ occurs when $\nabla q(\mathbf{b}) = 0$, which is the solution of the system of linear equations.

$$\mathbf{Z}_l^T\,\mathbf{H}\,\mathbf{Z}_l\,\mathbf{b} = -\,\mathbf{Z}_l^T\,\mathbf{c}\,. \tag{A.2}$$

Solving (A.2) for $\mathbf{b}$ at each QP iteration gives the $\acute{\mathbf{d}}_l$,then the step is taken as $\mathbf{d}_{l+1} = \mathbf{d}_l + \alpha\acute{\mathbf{d}}_l$. Since the objective is a quadratic function, there are only two choices of step length α; it is either 1 along search direction $\acute{\mathbf{d}}_l$ or less than 1. If the step length 1 can be taken without violation of the constraints, then this is the exact step to the minimum of the quadratic function. Otherwise, the distance to the nearest constraint should be found and the solution is moved along it as in (3.14).

Bibliography

[1] Andrea Goldsmith, "Wireless Communications," Cambridge University Press, 2006 .

[2] Math H. J. Bollen, Irene Yu-Hua Gu, "Signal processing of power quality disturbances," IEEE power series of power engineering, p. 207, 2006.

[3] Ali Behravan, "Evaluation and Compensation of Nonlinear Distortion in Multicarrier Communication Systems," PhD thesis in Communication System Group, Department of Signals and Systems, Chalmers University of Technology, Gothenburg, Sweden 2006.

[4] Jeffrey G, Andrews, Arunabha Ghosh, Rias Muhamed "Fundamentals of WiMAX: Understanding Broadband Wireless Networking," Prentice Hall Communications Engineering and Emerging Technologies Series, 2007.

[5] R. van Nee and A.de Wild, "Reducing the peak-to-average-power ratio of OFDM," in Proceedings of the 48th IEEE Vehicular. Technology Conference, vol. 3, pp. 2072-2076,1998.

[6] H. Ochiai and H. Imai, "On the Distribution of the PeaktoAverage Power Ratio in OFDM Signals," IEEE Trans. Commun., vol. 49, no. 2, pp. 282-89, Feb. 2001.

[7] Seung H. Han, Jae H. Lee, "An overview of peak-to-average power ratio reduction techniques for multicarrier transmission," IEEE Wireless Communications, vol.12, no. 2, pp. 56-65, 2005.

[8] R. O'Neill and L. B. Lopes, "Envelope Variations and Spectral Splatter in Clipped Multicarrier Signals, Proc. IEEE PIMRC '95, pp. 71-75, Toronto, Canada, Sept. 1995.

[9] X. Li and L. J. Cimini, Jr., "Effect of Clipping and Filtering on the Performance of OFDM, IEEE Commun.Lett., vol. 2, no. 5, pp. 131-33, May 1998.

[10] J. Armstrong, "Peak-to-Average Power Reduction for OFDM by Repeated Clipping and Frequency Domain Filtering, Elect. Lett., vol. 38, no. 8, pp. 246-47, Feb. 2002.

[11] A. E. Jones, T. A. Wilkinson, and S. K. Barton, "Block Coding Scheme for Reduction of Peak to Mean Envelope Power Ratio of Multicarrier Transmission Scheme," Elect. Lett., vol. 30, no. 22, pp. 2098-99, Dec. 1994.

[12] A. E. Jones and T. A. Wilkinson, "Combined Coding for Error Control and Increased Robustness to System Nonlinearities in OFDM ," Proc. IEEE VTC '96, pp. 90408, Atlanta, GA, May 1996.

[13] S. H. Mller and J. B. Huber, "A Comparison of Peak Power Reduction Schemes for OFDM," Proc. IEEE GLOBECOM '97, pp. 1-5, Phoenix, AZ, Nov. 1997.

[14] R. W. Buml, R. F. H. Fisher, and J. B. Huber, "Reducing the Peak-to-Average Power Ratio of Multicarrier Modulation by Selected Mapping," Elect. Lett., vol. 32, no. 22, pp. 205657, Oct. 1996.

[15] J.Tellado,"Peak to Average Power reduction in Multicarrier Modulation," PhD disertation, Stanford University, 2000.

[16] B. S. Krongold and D. L. Jones, "PAR Reduction in OFDM via Active Constellation Extension," IEEE Transactions on Broadcasting, vol.3, pp. 258-268, September 2003.

[17] Cristina Ciochina, Fabien Buda and Hikmet Sari, "An Analysis of OFDM Peak Power Reduction Techniques for WiMAX Systems," Sequans Communications, 19 Le Parvis, 92073 Paris La Dfense, France, Proceedings of IEEE International Conference on Communications, June 2006 .

[18] H. Breiling, S. H. Mller-Weinfurtner, and J. B. Huber, "SLM Peak-Power Reduction without Explicit Side Information," IEEE Commun. Lett., vol. 5, no. 6, pp. 239-41, June 2001.

[19] L. J. Cimini, Jr. and N. R. Sollenberger, "Peak-to-Aver-age Power Ratio Reduction of an OFDM Signal Using Partial Transmit Sequences," IEEE Communications Letter, vol.4, no. 3, pp. 86-88, March 2000.

[20] S. H. Mller and J. B. Huber, "A Novel Peak Power Reduction Scheme for OFDM," in Proceedings IEEE PIMRC '97, Helsinki, Finland,vol.3, pp. 1090-94, September 1997.

[21] S. Wei, D. L. Goeckel, and P. E. Kelly, A Modern Extreme Value Theory Approach to Calculating the Distribution of the PAPR in OFDM Systems, Proc. IEEE ICC, pp. 168690, New York, May 2002.

[22] M. Sharif, M. Gharavi-Alkhansari, and B. H. Khalaj, "On the Peak-to-Average Power of OFDM Signals Based on Oversampling," IEEE Trans. Commun., vol. 51, no. 1, pp. 7278, Jan. 2003.

[23] G. Wunder and H. Boche, "Upper Bounds on the Statistical Distribution of the Crest-Factor in OFDM Transmission," IEEE Trans. Info. Theory, vol. 49, no. 2, pp. 48894, Feb.2003.

[24] R. O'Neill and L. B. Lopes, "Envelope Variations and Spectral Splatter in Clipped Multicarrier Signals, Proc. IEEE PIMRC '95, pp. 71-75, Toronto, Canada, Sept. 1995.

[25] Kennedy, J. and Eberhart, R. ," Particle Swarm Optimization [C]," Proc. IEEE International Conference on Neural Networks, Perth, pp. 1942-1945, Australia 1995.

[26] Pedersen, M.E.H.,"Tuning and Simplifying Heuristical Optimization," PhD thesis, University of Southampton, School of Engineering Sciences, Computational Engineering and Design Group, 2010.

[27] Pedersen, M.E.H., Chipperfield A.J. ,"Simplifying particle swarm optimization,"Applied Soft Computing Conf., pp. 618-628, 2010.

[28] Gill, P.E., W. Murray, M.A. Saunders, and M.H. Wright, "Procedures for Optimization Problems with a Mixture of Bounds and General Linear Constraints," ACM Trans. Math. Software, Vol. 10, pp 282-298, 1984.

[29] Gill, P.E., W. Murray and M.H. Wright, "Numerical Linear Algebra and Optimization,"Addison-Wesley Publishing Company, vol.1, 1991.

[30] Jorge Nocedal, Stephen J. Wright,"Numerical Optimization," Berlin, New York: Springer-Verlag, 2nd Edition, p. 449, 2006 .

[31] Murty, K. G.,"Linear complementarity, linear and nonlinear programming," Sigma Series in Applied Mathematics, Heldermann Verlag, Berlin, Germany, vol.3, 1988.

[32] R. Fletcher, "Practical Methods of Optimization," John Wiley and Sons, 2nd Edition,ISBN: 978-0-471-49463-8, 1987.

[33] Gill, P.E., W. Murray and M.H. Wright,"Practical Optimization," Academic Press, London, 1981.

[34] Powell, M.J.D., "Variable Metric Methods for Constrained Optimization",in Mathematical Programming: The State of the Art, (A. Bachem, M. Grotschel and B. Korte, eds.) Springer Verlag, pp. 288-311, 1983.

[35] H. W. Kuhn,A. W. Tucker,"Nonlinear programming," Proceedings of 2nd Berkeley Symposium, Berkeley: University of California Press, pp. 481-492, 1951.

[36] Zhijun Yi,"Ab-initio Study of Semi-conductor and Metallic Systems: from Density Functional Theory to Many Body Perturbation Theory,"Doctoral thesis Johns Hopkins Studies in Mathematical Sciencesin Department of Physics University of Osnabruck, p.27, October 2009.

[37] Nima Amjady, Farshid Keynia,"Application of a new hybrid neuro-evolutionary system for day-ahead price forecasting of electricity markets," Applied Soft Computing, vol.10, Issue 3, pp. 784-792, June 2010.

[38] Jorge Nocedal, Stephen J. Wright,"Numerical Optimization," Berlin, New York: Springer-Verlag, 2nd Edition, p. 449, 2006 .

[39] Ya-Jun Qu, Bao-Gang Hu, "RBF networks for nonlinear modelsJohns Hopkins Studies in Mathematical Sciences subject to linear constraints," IEEE International Conference on Granular Computing, GRC '09, pp. 482-487, September 2009.

[40] K. Schittkowski, "NLQPL: A FORTRAN-Subroutine Solving Constrained Nonlinear Programming Problems," Annals of Operations Research, Vol. 5, pp 485-500, 1985.

[41] M.J.D. Powell, "A Fast Algorithm for Nonlinearly Constrained Optimization Calculations,"in Numerical Analysis, G.A.Watson ed., Lecture Notes in Mathematics, Springer Verlag, Vol.630, 1978.

[42] S.G. Kang, J.G. Kim, and E.K. Joo, "A novel sub-block partition scheme for partial transmit sequence OFDM," IEEE Trans. Commun., vol. 45, no.9, pp.333-338.

[43] Duhamel, P., and M. Vetterli, "Fast Fourier transforms: a tutorial review and a state of the art," Signal Processing Journal, vol. 19, Issue 4, April 1990.

[44] Bernard Mulgrew, Peter M. Grant and John S. Thompson, "Digital Signal Processing: Concepts and Applications," 2nd Edition, Edinburgh, August 2002.

[45] Matlab optimization toolbox user guide, Constrained Optimization , The MathWorks, Inc.,Chapter 6, pp. 227-235, 1984-2010.