Block-Shanno Minimum Bit Error Rate Pre-FFT Beamforming for OFDM Communication Systems

Waleed Abdallah
Technology and Applied Sciences Collage
Al-Quds Open University, Jerusalem, Palestine

ABSTRACT

In broadband wireless communication, orthogonal frequency division multiplexing (OFDM) is used as a multi-carrier technique to combat the inter-symbol interference (ISI). Adaptive array antenna (AAA) can be combined with OFDM to reduce the effect of directional interferences. The optimum beamformer weight set is based on minimum bit error rate (MBER) criteria in pilot-assisted OFDM systems. The development of a block-data adaptive implementation of the MBER beamforming solution is based on to the Parzen window estimates of probability density function. The Gradient Newton algorithm has been proposed to enhance the performance and increase the convergence rate speed, but at the expense of complexity. In this paper a block processing objective function for the MBER is formatted in three beamfroming algorithm, Least MBER (LMBER), Newton Least MBER (NLMBER), and Block-Shanno MBER (BSMBER) are proposed on Pre-FFT OFDM system. Simulation results showed that the BSMBER algorithm structure had the lowest computational complexity, the best BER performance and the fastest convergence rate over the other algorithms.

Keywords

MBER beamforming, OFDM systems, Pre-FFT, MMSE beamforming, probability density function (pdf), Smart antenna.

1. INTRODUCTION

It is well known that the orthogonal frequency division multiplexing (OFDM) can be considered as an efficient technique for high speed digital transmission over sever multipath fading channels where the delay spread is larger than the symbol duration [1]. As inserting the guard time longer than the delay spread of the channel makes the system robust against inter-symbol interference (ISI). In addition to that, channel estimation and compensation can be achieved by inserting known pilot symbols between data symbols [1]-[2]. Over the last few years beamforming antenna have gained much attention due to its ability to increase the performance of wireless communication systems, in terms of spectrum efficiency, network scalability, and operation reliability. Adaptive beamforming can separate signals that transmitted in the same carrier frequency, provided that they are separated in the spatial domain. Where the beamformer combines the signal received by the different element of an antenna array to form a single output [3].

The main motivation behind Pre-FFT scheme is reduceing the cost due to FFT processing [1]-[6]. Where the weight obtained for each pilot sub carrier can be identically applied on all data subcarriers in the same cluster thereby reducing the number of frequency domain narrow-band beamformers. The MBER technique is characterized by its good performance and amenability to adapt on implementation. The main obstacle that hamper stochastic MBER algorithm is the constant step size, which must be very carefully chosen in order to guarantee fast convergence to the minimum BER [3].

Stochastic Gradient Newton algorithm requires fewer training samples, hence, it speeds up convergence rate, is proposed in. It can optimize the step size by calculating the Hessian function [7]. Therefore the step size has been set carefully, depending on the amount of interference and the channel fading coefficients in order to ensure optimum convergence speed. Practically, wireless channels are changeable by time. Hence, a variable step size is recommended to compensate for the changes that happen in channel characteristic. The Shanno algorithm [7] provides an excellent choice, as it uses inexact search to converge to the optimum step size. A modified version of the Shanno algorithm has been deployed to optimize some cost function, e.g. the constant modulus objective function [7]-[8]. As it offers a good convergence speed with variable step size at low computational load. The main contribution of this paper is the deployment the modified versions of the Least MBER (LMBER), the Newton-Least MBER (NLMBER) and Block-Shanno MBER (BSMBER) algorithms to minimize the BER cost function in Pre-FFT OFDM beamforming system. The comparative analysis for these MBER is conducted based on algorithm in the terms of convergence rate, BER performance, array factor pattern and computational complexity.

This paper is organized as follows: Section 2 describing the Pre-FFT OFDM system model. In Section 3 MBER beamforming algorithms. Section 5 it provides simulation results and comparative analysis. Finally in section 6 conclusions and possible directions can be reached to achieve future works.

2. PRE-FFT OFDM SYSTEM MODEL

A *M*-user Pre-FFT OFDM communication system employing an *P*-element uniform linear array (ULA) with half wavelength spacing ($d = \lambda / 2$) is considered at the base station [1]. As it uses *K* subcarriers for parallel transmission.

The sample modulated by the k^{th} sub-carrier of the m^{th} user is given by

$$x_m(k) = b_m(k) \qquad 1 \le m \le M \qquad 1 \le k \le K \qquad (1)$$

where $b_m(k) \in \{\pm 1\}$ for BPSK modulation. We assume that user 1 is the desired user and the other sources are interfering users. This data can be interpreted to be a frequency-domain data in an OFDM system and subsequently converted to a time-domain signal by an IFFT operation. The output of the IFFT is transmitted to the channel after the adding of cyclic prefix (CP). This process can be considered as:

$$\bar{y}_m = \frac{1}{K} F^H \bar{x}_m \qquad 1 \le m \le M \qquad (2)$$

where

$$\bar{y}_m = [y_m(1), y_m(2), \ldots, y_m(K)]^T \qquad (3)$$

$$F = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ 1 & e^{\frac{-j2\pi(1)(1)}{K}} & \cdots & e^{-\frac{j2\pi(1)(K-1)}{K}} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & e^{\frac{-j2\pi(K-1)(1)}{K}} & \cdots & e^{-\frac{j2\pi(K-1)(K-1)}{K}} \end{bmatrix} \qquad (4)$$

$$\bar{x}_m = [x_m(1), x_m(2), \ldots, x_m(K)]^T \qquad (5)$$

F represents FFT operation matrix and H denotes the Hermitian transpose of a matrix. In order to add the CP, $\bar{y}_m$ is cyclically extended generating $\tilde{y}_m$ by inserting the last v element of y_m at its beginning,

$$\tilde{y}_m = \begin{bmatrix} J_v \\ I_K \end{bmatrix} \bar{y}_m \qquad (6)$$

where J_v contains the last v rows of a size K identity matrix I_K.

Then the OFDM time signals are transformed to the analog form by D/A converter then transmitted in the wireless channel. We assume that a multipath channel including maximum of L paths exists between the m^{th} source (desired or interference) and the array in the form of

$$h_m(k) = \sum_{l=0}^{L-1} \alpha_{m,l}\,\delta(k-l) \qquad m = 1, \cdots, M \qquad (7)$$

where $\alpha_{m,l}$ denotes a complex random number representing the l^{th} channel coefficient for the m^{th} source.

Fig.1 illustrates the architecture of Pre-FFT beamforming at the receiver of OFDM system. Assuming that the CP is longer than the channel length ($v > L$), as the received signal on the p^{th} antenna of a ULA for one OFDM block will be given as follow:

$$r_p(k) = \sum_{m=1}^{M}\sum_{l=0}^{L-1} \alpha_{m,l}\tilde{y}_m(k+v-l) e^{-j\frac{2\pi}{\lambda}(p-1)d\cos(\theta_{m,l})} + \eta_p(k)$$
$$1 \le p \le P \quad , \quad 1 \le k \le K \qquad (8)$$

where $\eta_p(k)$ represents the channel noise which enter the p^{th} antenna. $\theta_{m,l}$ denotes the direction of arrival (DOA) of the l^{th} path and m^{th} source. Without losing generalization as we have assumed here that the channels of all sources have the same length L. At the receiver, the A/D converted signal with a spatial domain for each array element is multiplied by the weight (w) of adaptive beamformer, then transform it back into frequency-domain (data and pilot) symbols by the FFT. Where this process can be written as follow:

$$Z(k) = W^H \cdot \bar{r}(k) \qquad (9)$$

$$W = [w_1\ w_2 \quad \cdots w_P]^T \qquad (10)$$

$$\bar{R} = [\bar{r}(1)\bar{r}(2)\cdots\bar{r}(k)] \qquad (11)$$

$$\bar{r}(k) = [r_1(k)\, r_2(k) \quad \cdots r_P(k)]^T \qquad (12)$$

$$\bar{Z} = [z(1)\, z(2) \quad \cdots z(K)] \qquad (13)$$

$$\hat{Z} = \bar{Z} \cdot F \qquad (14)$$

where $\hat{Z}$ is the frequency-domain data, and is given by

$$\hat{Z} = [\hat{z}(1), \hat{z}(2), \ldots, \hat{z}(K)] \qquad (15)$$

and $\hat{z}(k)$ denotes the corresponding received sample at the k^{th} subcarrier .

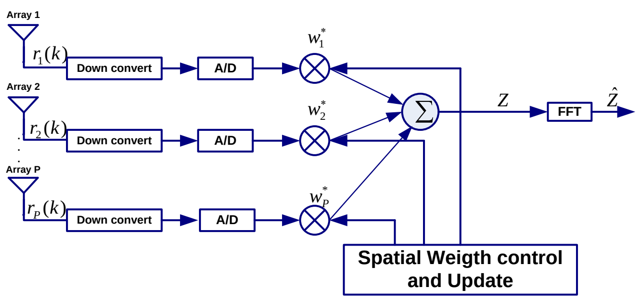

Fig.1. Block diagram of the Pre-FFT OFDM adaptive receiver

The estimate of the transmitted bit $b_1(k)$ is given by

$$\hat{b}_1(k) = \begin{cases} +1, & \mathrm{Re}(\hat{z}(k)) > 0 \\ -1, & \mathrm{Re}(\hat{z}(k)) \le 0 \end{cases} \qquad (16)$$

where $\mathrm{Re}(\hat{z}(k))$ denotes the real part of $\hat{z}(k)$.

3. MBER BASED BEAMFORMING ALGORITHMS

The theoretical MBER solution for the Pre-FFT OFDM beamformer is obtained in [2]. The error probability (BER cost function) of the frequency domain signal of beamformer is given by

$$P_E(W) = \mathrm{Prob}\{\mathrm{sgn}(b_1(k)\mathrm{Re}(\hat{z}(k)) < 0\} \qquad (17)$$

where $\mathrm{sgn}(\cdot)$ is the sign function. The weight vector that minimize the BER is then defined as

$$W = \arg\min_W P_E(W) \qquad (18)$$

From equation (17), define the signed decision variable

$$\begin{aligned} \hat{z}_s(k) &= \mathrm{sgn}(b_1(k))\mathrm{Re}(\hat{z}(k)) \\ &= \mathrm{sgn}(b_1(k))\mathrm{Re}(\hat{z}'(k)) + \eta'(k) \end{aligned} \qquad (19)$$

where

$$\hat{z}'(k) = W^H[\bar{r}(k) - \eta(k)]F(k) \qquad (20)$$

and

$$\eta'(k) = \mathrm{sgn}(b_1(k))\mathrm{Re}(W^H \eta(k)F(k)) \qquad (21)$$

$\hat{z}_s(k)$ is a best error indicator for the binary decision, i.e., if it is positive, then the decision is correct, else if it is negative, then an error is occurred. and $F(k)$ is the k^{th} column of F. Notice that F is unitary matrix, so $\eta'(k)$ is still Gaussian with zero mean and variance $\sigma_n^2 \cdot W^H W$.

The conditional probability density function (pdf) given the channel coefficients $\alpha_{m,l}$ of the error indicator, $\hat{z}_s(k)$, is a mixed sum of Gaussian distributions [3], i.e.,

$$p_z(\hat{z}_s) = \frac{1}{K\sqrt{2\pi}\sigma_n\sqrt{W^H W}} \cdot \sum_{k=1}^{K} \exp\left(-\frac{(\hat{z}_s - \mathrm{sgn}(\hat{b}_1(k))\mathrm{Real}(\hat{z}_s'(k)))^2}{2\sigma_n^2 W^H \cdot W}\right) \qquad (22)$$

and it is the best indicator of a beamformer's BER performance.

Hence, the conditional error probability given the channel coefficients $\alpha_{m,l}$ of the beamformer, $P_E(W)$, is given by [4] Deriving a closed form for the average error probability is not easy. Therefore, we use the gradient conditional error probability to update the weight vector.

$$\begin{aligned} P_E(W) &= \frac{1}{K\sqrt{2\pi}\sigma_n\sqrt{W^H W}} \cdot \sum_{k=1}^{K} \int_{q_k(W)}^{\infty} \exp\left(-\frac{u^2}{2}\right)du \\ &= \frac{1}{K}\sum_{k=1}^{K} Q(q_k(W)) \end{aligned} \qquad (23)$$

where

$$u = \frac{(\hat{z}_s - \mathrm{sgn}(b_1(k))\mathrm{Re}(\hat{z}_s'(k))}{\sigma_n\sqrt{(W^H W)}} \qquad (24)$$

where $Q(\cdot)$ is the Gaussain error function and is given by

$$Q(u) = \frac{1}{\sqrt{2\pi}}\int_u^{\infty} \exp\left(-\frac{y^2}{2}\right)dy \qquad (25)$$

$$q_k(W) = \frac{\mathrm{sgn}(b_1(k))\mathrm{Re}(\hat{z}'(k))}{\sigma_n\sqrt{W^H W}} \qquad (26)$$

In OFDM system as described in section 2, it is assumed that in every symbol there are a pilot signals in order to perform channel estimation. The pilot signals are also used in the adaptive update of the beamformer weight vector. So, the transmitted pilot signal vector of the desired user, $\bar{x}_{1p}$, and the received pilot signal vector, $\bar{\hat{z}}_p$, in frequency domain can be written as follows:

$$\bar{x}_{1p} = [x_1(1),0..,x_1(\Delta p+1),0,..,x_1((K_p-1)\Delta p+1)),0,..] \qquad (27)$$

$$\begin{aligned} \bar{\hat{z}}_p &= [\hat{z}(1),0..,\hat{z}(\Delta p+1),0,..,\hat{z}((K_p-1)\Delta p+1),0,..] \\ &= W^H \bar{R} F_p \end{aligned} \qquad (28)$$

where

$$F_p = \begin{bmatrix} 1 & 0 & \cdots & 1 & \cdots & 1 & 0\cdots \\ 1 & 0 & \cdots & e^{-j2\pi(1)(\Delta p)/K} & \cdots & e^{-j2\pi(1)(K_p-1)\Delta p/K} & 0\cdots \\ \vdots & 0 & & \vdots & \ddots & \vdots & 0\cdots \\ 1 & 0 & \cdots & e^{-j2\pi(K-1)(\Delta p)/K} & \cdots & e^{-j2\pi(K-1)(K_p-1)\Delta p/K} & 0\cdots \end{bmatrix} \qquad (29)$$

This represents FFT operation matrix at the pilot locations and Δp represents the frequency spacing between consecutive pilot symbols. The first pilot symbol is assumed to be positioned at the first subchannel. For each OFDM block,

K_p pilot signals in the frequency domain are required to obtain the optimum weight vector.

The method of approximating a pdf known as a kernel density or Parzen window-based estimate, [3], to estimate the error probability is used on OFDM systems. Given a block of K_p training samples $\{\bar{r}(k), b_1(k)\}$, a kernel density estimate of the conditional pdf given the channel coefficients $\alpha_{m,l}$ at pilot locations which was defined in (22), is given by

$$\widehat{p}_z(\hat{z}_s) = \frac{1}{K_p\sqrt{2\pi}\rho_n\sqrt{W^HW}} \times \sum_{k=0}^{K_p-1} \exp(-\frac{(\hat{z}_s - \mathrm{sgn}(b_1(k\times\Delta p+1))\mathrm{Re}(\hat{z}(k\times\Delta p+1)))^2}{2\rho_n^2 W^H W}) \tag{30}$$

where the kernel width ρ_n is related to the standard deviation σ_n of the channel noise. Therefore, the block cost function could be derived form the kernel density estimate of conditional pdf given the channel coefficients $\alpha_{m,l}$ as follows [3],[7],[12]:

$$\hat{P}_E(W) = \frac{1}{K_p}\sum_{k=0}^{K_p-1} Q(\hat{q}_k(W)) \tag{31}$$

where $\hat{q}_k(W) = \dfrac{\mathrm{sgn}(b_1(k\times\Delta p+1)\mathrm{Re}(W^H\bar{R}F_p(k\times\Delta p+1))}{\rho_n\sqrt{W^HW}}$ (32)

and $F_p(k\times\Delta p+1)$ is the $(k\times\Delta p+1)^{th}$ column of F_p. From this estimated conditional pdf given the channel coefficients $\alpha_{m,l}$, the gradient of the estimated BER is given by [4]

$$\nabla\hat{P}_E(W) = -\frac{1}{\sqrt{2\pi}\rho_n\sqrt{W^HW}}\sum_{k=0}^{K_p-1}\exp(-\frac{(\mathrm{Re}(\hat{z}(k\times\Delta p+1)))^2}{2\sigma_n^2W^HW} \times \mathrm{sgn}(b_1(k\times\Delta p+1))\bar{R}F_p(k\times\Delta p+1) \tag{33}$$

Now a block-data adaptive MBER algorithm is obtained by the gradient of $\widehat{P}_E(W)$. To reduce the computational complexity, we set $W = \dfrac{W}{\sqrt{W^HW}}$. At time instant *n*, we can find the optimum weight vector W by the steepest-descent gradient algorithm [3]

$$\nabla\widehat{P}_E(W) = -\frac{1}{\sqrt{2\pi}\rho_n}\exp(-\frac{(\mathrm{Re}(\hat{z}(k\times\Delta p+1)))^2}{2\rho_n^2}) \times \mathrm{sgn}(b_1(k\times\Delta p+1))\bar{R}F_p(k\times\Delta p+1) \qquad 1\le k\le K_p \tag{34}$$

That is to say, W weight vector can be updated K_p times in one OFDM symbol. Thus complexity is reduced.

3.1 STOCHASTIC GRADIENT (LMBER) ADAPTIVE ALGORITHM

The LMBER algorithm, which can be considered as one of the most popular stochastic gradient algorithms that minimize the cost function of the BER, was developed by Chen [3]. The LMBER beamforming seeks to minimize of the cost function in (31). Consequently, its update equation will be given as follow:

$$\begin{aligned} W(k+1) &= W(k) - \mu\nabla\widehat{P}_E(W) \\ &= W(k) + \frac{\mu}{\sqrt{2\pi}\rho_n}\cdot\exp(-\frac{(\mathrm{Re}(\hat{z}(k\times\Delta p+1)))^2}{2\rho_n^2}) \\ &\times \mathrm{sgn}(b_1(k\times\Delta p+1))\bar{R}F_p(k\times\Delta p+1) \qquad 1\le k\le K_p \end{aligned} \tag{35}$$

where μ is a step size.

3.2 STOCHASTIC GRADIENT NEWTON-LMBER

Gradient-Newton algorithm incorporates second order statistics of input signals, increasing their convergence rate. It usually has a faster convergence rates than other gradient techniques but in the cost of computational complexity [7]. Practically, estimates of the covariance and gradient matrices are required to converge to the desired solution. The weight update of Newton's method is given by

$$W(k) = W(k-1) + \mu R_{xx}^{-1}\nabla P_E \tag{36}$$

where

$$\hat{R}_{xx} = \frac{1}{K}\sum_{k=1}^{K} r(k)r^H(k) \tag{37}$$

is the autocorrelation matrix of the received signal. Since matrix inversion requires a lot of computation, the inverse can be computed according to the following rank 1 update, which reduces the computational complexity to $O(2P^2+5/2P)$ [7]:

$$R_{rr}^{-1}(k) = R_{rr}^{-1}(k-1) - \frac{R_{rr}^{-1}(k-1)\bar{r}(k)\bar{r}(k)^H(k)R_{rr}^{-1}(k-1)}{1+\bar{r}(k)^H(k)R_{rr}^{-1}(k-1)\bar{r}(k)} \tag{38}$$

with $R_{rr}^{-1}(0) = \dfrac{1}{\varepsilon}I, \qquad \varepsilon > 0$ (39)

The step size μ is introduced in order to protect the algorithm from divergence due to the usage of the noisy estimates of the covariance matrix and the gradient vector.

3.2 Block Shanno MBER

Shanno algorithm is a memoryless modified version of Newton algorithm. Like the gradient algorithm, Shanno algorithm involves an implicit computation of the Hessian [7]. However, Shanno algorithm does the conjugate gradient type search without fully optimizing the step size. The constant step size depended on the interference as well as the channel coefficients. As a result, the need to implement a new algorithm in order to take the advantage from the Newton during maintaining linearity in complexity, that's all, has motivated this work. The step size was chosen to be within a specified range so that the convergence can be guaranteed.

The higher and lower bounds of the step size must satisfy the following inequalities [8],[10]-[11]:

$$P_E(W(k)) < P_E(W(k-1)) + \alpha\mu(k)\nabla P_E(W(k-1))^T D(W(k-1)) \qquad (2P \text{ ops.}) \quad (40)$$

In addition, the lower bound of the step size must satisfy the following inequality [8],[9]-[11]:

$$\nabla P_E(W(k))^T D(W(k)) > \beta\nabla P_E(W(k-1))^T D(W(k-1)) \quad (41) \qquad (4P \text{ ops.})$$

where α and β are constant, and $D(W(k-1))$ is the search direction vector. This feature saves computation as it only includes an implicit computation of the Hessian matrix. There is no need to compute, update, and store the inverse of the Hessian, as it is the most costly part of the Newton algorithm. Shanno algorithm is also known by its rapid convergence rate, and it is designed to achieve quickly and efficiently minimizing rate of nonlinear objective functions. The weight update equation for the Shanno algorithm is given by [8]:

$$W(k) = W(k-1) + \mu(k)D(k-1) \qquad (2P \text{ ops.}) \quad (42)$$

The search direction vector is a linear combination of the negative gradient, which differ between the current gradient and the previous gradient, and the previous search direction, as it is defined as [10]

Table 1. BSMBER algorithm summary

Initialization
$i=1, \mu=.015, \varepsilon=.015, \alpha=0.25, \beta=0.5,$ Block $K=64,\ W(0)=0.01*\text{ones}(P,1)$
Outer loop (1: floor (all bits/Block))
• Form a block of data from the received signals. • Convert the complex data into real as in (50) to (51). (2*P* ops.) • Initialize $\nabla P_E = \text{ones}(2\cdot P,1)$, $P_E = 1$
Inner loop (while $k<5$ *or* $\lVert\nabla P_E\rVert < \varepsilon$
• Calculate the cost function BER and the gradient matrix over the block from equations (30) and (33). (2P ops. for each block) • If *k=1* then $D=-\nabla P_E$ else calculate D from equation (41). • Check the direction matrix D , if $\dfrac{\lvert D(k)\nabla P_E(k)\rvert}{\lVert D(k)\rVert\cdot\lVert\nabla P_E(k)\rVert} < \varepsilon$ then $D=-\nabla P_E$. (2*P* ops.) • Check the step size $\mu(k)$ as in equations (40) and (41). If it falls outside the boundaries, then increase or decrease the step size as $\mu(k)=\mu(k)\pm\Delta\mu$ • Update the weight matrix as $W_c(k)=W_c(k-1)+\mu(k)D(k)$ from equation (42). • Transfer $W_c(k)$ to complex form again.
• *end of inner loop* • Determine the detected signals in order to be used for calculating the BER. • Increment the block number $i=i+1$
• *end of outer loop*

$$D(k) = -\nabla P_E(W(k)) + a(k)u(k) + [b(k) - c(k)a(k)]D(k-1)$$
(4*P* ops.) (43)

where $u(k)$ is the gradient difference between the current gradient and the previous gradient is defined as [8],[10]-[12]

$$u(k) = \nabla P_E(W(k)) - \nabla P_E(W(k-1)) \quad (44)$$

$$a(k) = \frac{D^T(k-1)\nabla P_E(k))}{D^T(k-1)u(k)} \quad (4P \text{ ops.}) \quad (45)$$

$$b(k) = \frac{u^T(k)\nabla P_E(W\ (k))}{D^T(k-1)u(k)} \quad (2P \text{ ops.})$$

(46)

$$c(k) = \mu(k-1) + \frac{|u(k)|^2}{D^T(k-1)u(k)} \quad (2P \text{ ops.}) \quad (47)$$

The main advantage of the Shanno algorithm is its quick and efficient ability to minimize nonlinear objective function. Hence, the Shanno algorithm can be considered as the best kind of algorithm to optimize the BER cost function. The proposed BSMBER algorithm processes the data on a block-by-block basis, i.e., it takes in a block of data and iterates until matching the convergence tolerance criterion. As in order to use the Block Shanno algorithm we must convert the complex data into real. We define

$$W(k) = W_{\mathrm{Re}}(k) + jW_{\mathrm{Im}}(k) \quad (48)$$

$$R(k) = R_{\mathrm{Re}}(k) + jR_{\mathrm{Im}}(k) \quad (49)$$

where $j = \sqrt{-1}$ and Im is imaginary-part .Then, we define the following new vectors [10]:

$$W_c(k) = \begin{bmatrix} W_{\mathrm{Re}}(k) \\ W_{\mathrm{Im}}(k) \end{bmatrix} \quad (50)$$

$$R_c(k) = \begin{bmatrix} R_{\mathrm{Re}}(k) \\ R_{\mathrm{Im}}(k) \end{bmatrix} \quad (51)$$

Now, the detect signal $\hat{z}_c$ is given by

$$\hat{z}_c = W_c^T R_c \quad (2P \text{ ops. for each block}) \quad (52)$$

The proposed BSMBER algorithm is summarized in Table 1

The proposed BSMBER algorithm is composed of two main loops. The outer loop is for each block of data and the inner loop is repeated over the same block of data until certain number of iterations reached.

In the main loop, we formulate a block of data (64 bits) from the output of the antenna array and convert it to real format, as in (50)-(51). In the inner loop, the BER cost function and the gradient vector are determine from (30) and (33) at pilot locations (N_p =16). Then, we compute the search direction vector form (41), and unless we start the beginning of the inner loop, we set it to be equal to the negative of the gradient. After that we check the search direction vector, if it was in the wrong direction; we reset it back to the negative of the gradient. Then, we check the step size and adjust it according to its boundary. Then, we compute the weight update vector from (42) and convert it back to the complex form. After the end of the inner loop we determine the detected signal by multiplying the computed optimized weight vector with the received signal in order to use it in calculating the BER the last update at the end of each OFDM block $(W(K))$ which used as the initial value of the next block. Then, we back to the main loop and form another block of data and so on.

These processes iterate until we finish all the incoming data. According to observation, algorithm keeps tracking of the received signal with different channel and interference parameters rather than having a constant step-size that may force the algorithm to diverge in case of changing the channel and interference parameters. So the proposed BSMBER beamfoming using the stochastic gradient algorithm can obtain steady BER only with shorter training OFDM symbols.

4. Simulation Results

In this section the simulation is performed to illustrate and compare the performances of Pre-FFT beamformer using different MBER based algorithms. 64 subcarriers (16 + 48) are used The OFDM system is perfectly synchronized, with a CP length larger than the channel length (*v=16*). BPSK modulation is used in the system with six antenna elements and half-wavelength spacing. The example used in our computer simulation study considers one desired user with DOA at 90° and two interferers with SIR = -3dB and DOA 50° and 130° .We further assumed normalized channels with different lengths and with real coefficients 0.864, 0.435, 0.253 and 0 for all sources, and with angle spread of $\pm 15^o$ (for all sources) [2].

Figures 2 compare the BER performance against SNR for the LMBER, NLMBER, and BSMBER beamformer for the number of element is 6. Fig. 3 the number of elements is 8. It's observed that BER performance of the BSMBER beamformer is better than MBER and NMBER as it enhanced when the number of antenna elements are increased. The average BER which based on the Q function is determined after reaching the steady state.

Fig.4 illustrates the convergence rate of the LMBER, NLMBER and BSMBER beamformers for Pre-FFT OFDM adaptive antenna array. It can be seen in Fig.4 that the

LMBER and NMBER algorithms converges after 200 bits to reach the steady state, whereas the proposed BSMBER converges after 200 bits to reach steady state at the an SNR of 16 dB.

Fig.5 illustrates the beam pattern of the LMBER, NLMBER and BSMBER beamformers for Pre-FFT OFDM adaptive antenna array when the number of antenna elements is 6. It shows that the BSMBER pre-FFT beamformer has lowest sidelobe levels and deepest nulls.

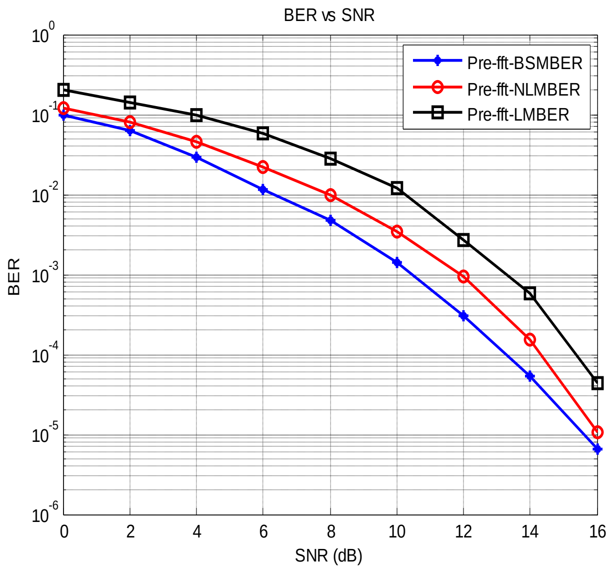

Fig.2 Comparison of the bit error rate performance.

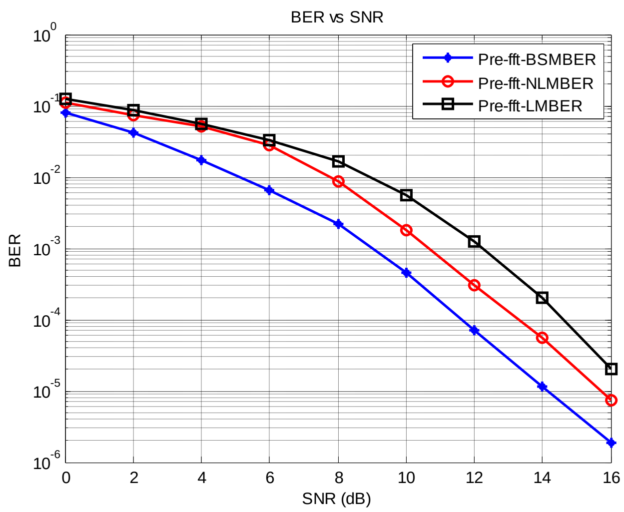

Fig.3. Comparison of the bit error rate performance.

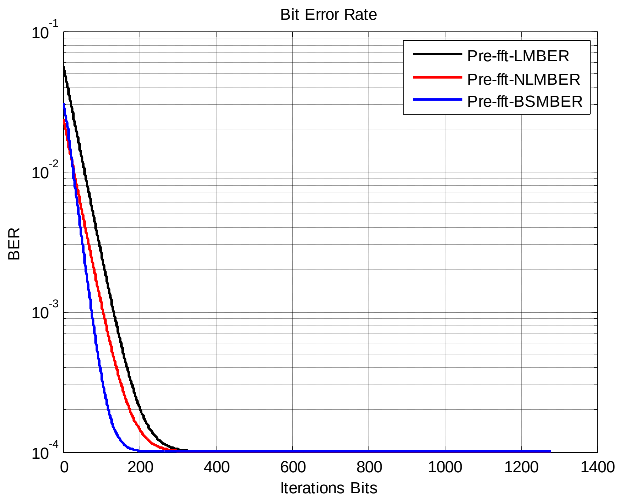

Fig.4. BER versus iterations SNR=16 dB and SIR=-3dB

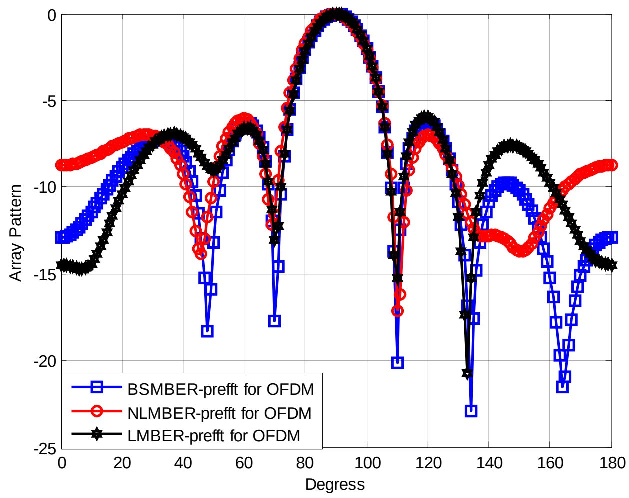

Fig. 5. Beam pattern of the BSMBER and NLMBER and LMBER.

5. CONCLUSION

In this paper, a block –Shanno algorithm for minimizing the BER cost function has been proposed. A comparison between this algorithm LMBER, Newton-LMBER and BSMBER is conducted in terms of BER performance and computational complexity. The BER performance is compared against the iteration index during adaptive implementation, SNR, beam pattern. The proposed BSMBER yields the best convergence speed while maintaining linear complexity.

6. REFERENCES

[1] Lingyan Fan, Haibin Zhang and Chen He ,"Minimum bit error beamforming for Pre-FFT OFDM adaptive antenna array", IEEE Intenational Conference, 28-25,Sept. 2005.

[2] Saeid Seydnejad and Sadegh Akhzari "A combined time-frequency domain beamforming method for OFDM systems", 2010 International ITG Workshop on Smart Antennas, IEEE, 23-24 Feb. 2010.

[3] S. Chen, N. N. Ahmad, and L. Hanzo, "Adaptive minimum bit-Error rate beamforming", IEEE Trans. on Wireless Comm., Vol. 4, No. 2 March 2005, pp.341-348.

[4] Z. Lei and P. S. Chin, "Post and Pre-FFT beamforming in an OFDM system" Proc. of IEEE Vehicular Tech. Conf., pp. Vol. 1, 17-19 May 2004, pp.39-43.

[5] Sannella, M. J. 1994 Constraint Satisfaction and J. Via´ , V. Elvira, I. Santamari´a, and R. Eickhoff, "Minimum BER beamforming in the RF domain for OFDM transmissions and linear receivers," in IEEE Int. Conf. on Acoustics, speech and signal Proc. (ICASSP 2009), Taipei, Taiwan, Apr. 2009.

[6] Shinsuke Hara, Montree Budsabathon, and Yoshitaka Hara, "A Pre-FFT OFDM adaptive antenna array with eigenvector combining", 2004 IEEE International Conference on comm. Vol. 4, June 2004, pp. 2412 - 2416.

[7] Tamer A. Samir, Said Elnoubi, and Ayman Elnashar, "Block Shanno MBER Beamformimg", International Symposium on Signal Processing and its Applications, ISSPA 2007, Sharjah, UAE, 12-15 Feb. 2007.

[8] D. F. Shanno, "Conjugate Gradient Method with Inexact Searches", Mathematics of operations research, vol. 3, pp.244-256, August 1978.

[9] S. Chen, Ahmad K.Samingan, and Bernard Mulgrew, "Adaptive minimum-BER linear multiuser detection for DS-CDMA signals in multipath channels," IEEE Trans. on signal process. vol.49, No.6, Jun.2001, pp.1240-1247.

[10] Z. Wang and E. M. Dowling, "Block-Shanno constant modulus algorithm for wireless equalizations," Proc. ICASSP, Atlanta, GA, USA, April 1996, pp. 2678-2681.

[11] U. G. Jani, E. M. Dowling, R. M. Golden, Z. Wang, "Multiuser interference suppression using block Shanno constant modulus algorithm," IEEE Trans. On Sig. Proc. Vol., 48, No. 5, May 2000.

[12] Tamer A. Samir, Said Elnoubi, Ayman Elnashar, "Block Shanno Minimum Bit Error Rate Beamforming " IEEE Trans. On Vehicular Technology, Vol., 57, No.5, September 2008.