**UNIVERSITE DE LIMOGES**

ECOLE DOCTORALE SCIENCE – TECHNOLOGIE –SANTE

FACULTE DES SCIENCES ET TECHNIQUES

XLIM – DEPARTEMENT $C^2S^2$ (UMR CNRS 6172)

Année : 2007 N°29-2007

# THESE

Pour obtenir le grade de

**DOCTEUR DE L'UNIVERSITE DE LIMOGES**

Discipline : Télécommunications des Hautes Fréquences et Optique

Présentée et soutenue par

Monsieur **Amir SAEMI**

Le 20 Septembre 2007

# SYNCHRONISATION DES SYSTEMES DE TRANSMISSION MIMO-OFDM

Directeurs de Thèse : Jean-Pierre CANCES et Vahid MEGHDADI

## JURY

| | |
|---|---|
| **Président** | M. Jean-Michel DUMAS, Professeur, Université de Limoges |
| **Rapporteurs** | M. Samir Saoudi, Professeur, ENST Bretagne |
| | M. Philippe Ciblat, Maître de conférences, HDR, ENST Paris |
| **Examinateur** | Mme. Yi Yuan, France TELECOM R&D |
| **Co-directeurs de thése** | M. Jean-Pierre CANCES, Professeur, Université de Limoges |
| | M. Vahid MEGHDADI, HDR, Université de Limoges |

## *Acknowledgement*

This manuscript is the result of my efforts in three years spent at the department C2S2 in the XLIM laboratory within the project "*Etudes des systèmes de Télécommunications de l'ENSIL*" (ESTE). Funding of this thesis was provided by *France Telecom R&D* under grant 46 129562.

During this time, I was fortunate enough to have excellent supervisors: Vahid Meghdadi and Jean-Pierre Cances and I am so thankful to them for their continuing guidance, support and insights. Also, I have to express my thanks to Jean-Michel Dumas, the head of the project at the time, who supported all of us very kindly and generously.

Moreover, I would like to thanks the R&D section of *France Telecom* for their continuing guidance and their scientific and financial supports. A special mention for Mr. Patrick Tortelier, Mrs. Maryline Hélard.

I would also like to thank my colleagues, nay, friends, at ENSIL. I specially think to Reza Zahabi who was a great help and support for me during these years. I would also like to mention Ahmed Kora and Hamid Meghadadi.

Needless to say, I am very grateful to my parents. Although we were separated by thousands of miles, they provided me unlimited love and support throughout my many years of education. Without them, I would not achieve what I have accomplished.

Last but not least, thanks to my wife, Maryam Alaeddini, whose love, understanding, encouragement and support are endless. I would like to express my profound gratitude and love to her.

.

# CONTENTS

# FIGURES CAPTION

# NOTATIONS

The following notations are usually used in this manuscript unless another notation is explicitly mentioned.

**1)** $(.)^T, (.)^H$ denote transpose and conjugate transpose of $(.)$, respectively.

**2)** $\left[(.)\right]_{m,n}, \left[(.)\right]_{m,:}$ denote the $(m,n)^{\text{th}}$ element and $m^{\text{th}}$ row of $(.)$, respectively.

**3)** $E(.)$ denotes the expectation (.).

**4)** $\boldsymbol{A}_{K\times P}$ denotes a matrix with $K$ rows and $P$ columns.

**5)** $\hat{a}$ denotes the estimate of $a$.

**6)** $\boldsymbol{A}\otimes\boldsymbol{B}$ denotes the Kronecker tensor product of matrices $\boldsymbol{A}$ and $\boldsymbol{B}$.

$$\boldsymbol{A}\otimes\boldsymbol{B}=\begin{bmatrix} a_{11}.\boldsymbol{B} & a_{12}.\boldsymbol{B} & \cdots & a_{1n}.\boldsymbol{B} \\ a_{21}.\boldsymbol{B} & \cdots & \cdots & \vdots \\ \vdots & \ddots & \cdots & \vdots \\ a_{n1}.\boldsymbol{B} & a_{n2}.\boldsymbol{B} & \cdots & a_{nn}.\boldsymbol{B} \end{bmatrix}$$

**7)** Scalars are printed in italic and vectors and matrices are printed in bold.

**8)** We generally assume that we have a MIMO system with $N_t$ transmit and $N_r$ receive antennas.

**9)** $n$ and $m$ are usually used as the indexes for transmit and receive antenna. (In chapter 7 $p$ and $q$ are used as the indexes for transmit and receive antenna.

**10)** In an OFDM system we assume that we have $N_c$ subcarriers and $N_{CP}$ is the length of cyclic prefix.

**11)** $N_p$ is used to represent the length of training sequence or pilots.

**12)** $k$ is usually used as an index for time. In chapter 7, $k$ is used to refer to different users.

**13)** $\Delta f$ represents the frequency error. $\varepsilon$ which is normally called CFO is the multiplication of frequency error by sampling time: i.e. CFO (or ε) $\equiv \Delta f \times T_s$ ($T_s$ is sampling time)

**14)** $\eta_0$ represents the time offset less than sampling time.

**15)** $h$ and $H$ represent the channel. $h_{nm}(l)$ is the $l^{\text{th}}$ taps of equivalent channel between $n^{\text{th}}$ transmit antenna and $m^{\text{th}}$ receive antenna. We normally assume a quasi-static channel with length $L$. Moreover the following definitions are normally used through this manuscript:

$$\mathbf{h}_{nm}=\left[h_{nm}(0) \quad h_{nm}(1) \quad \ldots \quad h_{nm}(L-1)\right]^T_{L\times 1}$$

$$\mathbf{h}_{:m} = \begin{bmatrix} \mathbf{h}_{1m}^T & \mathbf{h}_{2m}^T & \dots & \mathbf{h}_{N_t m}^T \end{bmatrix}^T_{LN_t \times 1}$$

$$\mathbf{H}(l) = \begin{bmatrix} h_{11}(l) & h_{12}(l) & \cdots & h_{1N_r}(l) \\ h_{21}(l) & h_{22}(l) & \cdots & h_{2N_r}(l) \\ \vdots & \vdots & & \vdots \\ h_{N_t 1}(l) & h_{N_t 2}(l) & \cdots & h_{N_t N_r}(l) \end{bmatrix}$$

The above mentioned notations are used for time domain channels. On the other hand, $H_{mn}[p]$ represents the frequency domain channel corresponding $p^{\text{th}}$ sub carrier and between $n^{\text{th}}$ and $m^{\text{th}}$ transmit and receive antenna.

**16)** **r** represents the received vector and $r_{m,k}$ represents the $k^{\text{th}}$ received sample at the $m^{\text{th}}$ receive antenna.

Moreover, $\mathbf{r}_{m,k}$ denotes a column vector of received signal at the $m^{\text{th}}$ receive antenna starting from the $k^{\text{th}}$ instant i.e. $\begin{bmatrix} r_{m,k} & r_{m,k+1} & \dots & r_{m,k+N-1} \end{bmatrix}^T$. and $\mathbf{r}_k = \begin{bmatrix} \mathbf{r}_{1,k}^T & \mathbf{r}_{2,k}^T & \dots & \mathbf{r}_{N_r,k}^T \end{bmatrix}^T_{NN_r \times 1}$.

**r**($k$) denotes all the received sample at the $k^{\text{th}}$ instant from all receive antenna i.e. $\begin{bmatrix} r_{1,k} & r_{m,k} & \dots & r_{N_r,k} \end{bmatrix}^T$

In frequency domain, $r_m[p]$ presents the signal of $m^{\text{th}}$ antenna corresponding $p^{\text{th}}$ subcarrier.

The same indexation rules are applied to all the other signals represented by *s,u,* etc.

**17)** **w** represents the noise vector and $w_{m,k}$ represents the $k^{\text{th}}$ noise sample at the $m^{\text{th}}$ receive antenna. In this manuscript we suppose that we have zero mean complex Gaussian noise with covariance matrix $\sigma_w^2\mathbf{I}$.

*Chapter*


# INTRODUCTION

## 1.1 MIMO-OFDM

Orthogonal frequency division multiplexing (OFDM) has become a popular technique for transmission of signals over wireless channels. OFDM has been adopted in several wireless standards such as digital audio broadcasting (DAB), digital video broadcasting (DVB-T), the IEEE 802.11a / Wi-Fi [15] local area network (LAN) standard and the IEEE 802.16a / WiMax [16] metropolitan area network (MAN) standard. OFDM is also being pursued for dedicated short-range communications (DSRC) for road side to vehicle communications and as a potential candidate for fourth-generation (4G) mobile wireless systems.

OFDM converts a frequency-selective channel into a parallel collection of frequency flat subchannels. The subcarriers have the minimum frequency separation required to maintain orthogonality of their corresponding time domain waveforms, yet the signal spectra corresponding to the different subcarriers overlap in frequency. Hence, the available bandwidth is used very efficiently. If knowledge of the channel is available at the transmitter, then the OFDM transmitter can adapt its signalling strategy to match the channel. Due to the fact that OFDM uses

a large collection of narrowly spaced subchannels, these adaptive strategies can approach the ideal water pouring capacity of a frequency-selective channel. In practice, this is achieved by using adaptive bit loading techniques, where different sized signal constellations are transmitted on the subcarriers. OFDM is a block modulation scheme where a block of information symbols is transmitted in parallel on subcarriers. An OFDM demodulator can be implemented as a discrete Fourier transform (IDFT) on a block of information symbols followed by an analogue-to-digital converter (ADC). To mitigate the effects of intersymbol interference (ISI) caused by channel time spread, each block of IDFT coefficients is typically preceded by a cyclic prefix (CP) or a guard interval consisting of samples, such that the length of the CP is at least equal to the channel length. Under this condition, a linear convolution of the transmitted sequence and the channel is converted to a circular convolution. As a result, the effects of the ISI are easily and completely eliminated. Moreover, the approach enables the receiver to use fast signal processing transforms such as a fast Fourier transform (FFT) for OFDM implementation [17].

Multiple antennas can be used at the transmitter and receiver, an arrangement called a multiple-input multiple-output (MIMO) system. A MIMO system takes advantage of the spatial diversity that is obtained by spatially separated antennas in a dense multipath scattering environment. MIMO systems may be implemented in a number of different ways to obtain either a diversity gain to combat signal fading or to obtain a capacity gain. Generally, there are categories of MIMO techniques. The first aims to improve the power efficiency by maximizing spatial diversity. Such techniques include delay diversity, space–time block codes (STBC) [18], [19] and space–time trellis codes (STTC) [20]. One can think also in layered approach to increase capacity. One popular example of such a system is V-BLAST suggested by Foschini *et al.* [21] where full spatial diversity is usually not achieved.

OFDM has been adopted in the IEEE802.11a LAN (Wi-Fi) and IEEE802.16a LAN/MAN standards (WiMAX). OFDM is also being considered in IEEE802.20a, a standard for maintaining high-bandwidth connections to users moving at speeds up to 96 km/h. The IEEE802.11a LAN standard operates at raw data rates up to 54 Mb/s (channel conditions permitting) with a 20-MHz channel spacing, thus yielding a bandwidth efficiency of 2.7 b/s/Hz. The actual throughput is highly dependent on the medium access control (MAC) protocol.

Likewise, IEEE802.16a operates in many modes depending on channel conditions with a data rate ranging from 4.20 to 22.91 Mb/s in a typical bandwidth of 6 MHz, translating into a bandwidth efficiency of 0.7 to 3.82 bits/s/Hz. Recent developments in MIMO techniques promise a significant boost in performance for OFDM systems. Broadband MIMO-OFDM systems with bandwidth efficiencies on the order of 10 b/s/Hz are feasible for LAN/MAN environments. The physical (PHY) layer techniques described in this manuscript are intended to approach 10 b/s/Hz bandwidth efficiency.

A multicarrier system can be efficiently implemented in discrete time using an inverse FFT (IFFT) to act as a modulator and an FFT to act as a demodulator. The transmitted data are the "frequency" domain coefficients and the samples at the output of the IFFT stage are "time" domain samples of the transmitted waveform. Figure 1 shows a typical MIMO-OFDM implementation.

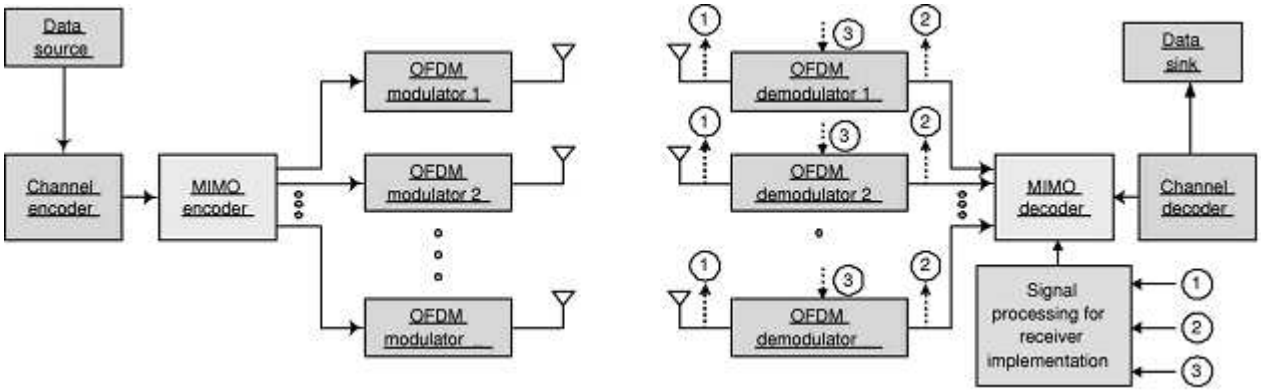


Figure 1: a typical MIMO-OFDM realization.

## 1.2 Synchronization of MIMO OFDM

Apparently, the synchronization of MIMO systems might seem closely related to the problem of synchronization in single-input-single-output (SISO) systems. In MIMO systems, the synchronization problem takes a special form because signals from different antennas are superimposed together, and some algorithms commonly proposed for SISO systems do not work well in MIMO systems and the other need some modifications.

The problem of synchronization in MIMO-OFDM systems as well as in all OFDM systems consists in three parts: channel estimation, frequency and time synchronization. The main focus of this thesis is on the problem of time and frequency synchronization for MIMO-OFDM systems. However, in chapter 2 a literature survey on the channel estimation will be presented. Channel state information is essential for channel equalization. Moreover, most MIMO algorithms suppose that channel state information is available. To acquire the channel state information normally some known data is used. The known data can be either a training sequence is used in the first of the data packet or some known pilots are inserted in the data packet. However, there are some other algorithms which use the information of data packet in order to increase the performance of information. Moreover, blind algorithms are discussed in the literature.

One of the most important tasks in a MIMO-OFDM receiver is frequency synchronization. A frequency offset can be introduced due a difference between the transmitter and receiver clock or due to the Doppler effect. OFDM systems work in frequency domain by using frequency orthogonal sub-carriers; Fourier transformation is used in an OFDM receiver as the modulator. Because of these facts even a very small frequency offset in a MIMO-OFDM system can destroy the orthogonality between the sub-carriers and thus the performance of the system significantly degrades. Chapter 3 is about frequency synchronization. We will investigate there how much carrier frequency error can affect in the performance of receiver. In this chapter several frequency synchronization algorithms will be discussed.

Although OFDM is well known for its ability to combat inter-symbol interference (ISI) introduced by multipath channel, incorrect positioning of the FFT window within an OFDM symbol reintroduces ISI during data demodulation, causing serious performance degradation.

Time synchronization is therefore one of the important tasks performed in the receiver. The problem of time synchronization in MIMO systems is divided into two parts:

- Frame synchronization: The task of the frame synchronization is to identify the preamble in order to detect a packet arrival
- Symbol-timing: The task of symbol timing is to identify the beginning of the symbol.

Thanks to use of FFT as demodulator in OFDM receiver, the time delay smaller than a symbol time will be converted to a phase shift in frequency domain and this phase shift can be estimated in the channel estimation stage. Hence, symbol timing in an OFDM system is to find the exact place of FFT window or, in other words, the exact beginning of the OFDM symbol. Moreover, concerning OFDM synchronization, one can find another problem referred as sampling clock frequency offset. This problem is not investigated in this manuscript due to preliminary assumption on the precision of clock frequency and small and moderate OFDM frame length.

## 1.3 Outline and Contribution

### 1.3.1 Outline

The chapter 2 deals with channel estimation. Data aided methods are discussed first. In the data aided methods we distinguish the case where a training sequence is used and the case where the pilots are used. Then, a brief introduction of blind and semi-blind channel estimation methods will be presented and at the end of this chapter we will see the iterative channel estimation methods which use that information of data packet to improve the estimation.

Chapter 3 is about frequency synchronization. In this chapter several methods for frequency synchronization will be presented and compared.

The issue of time synchronization will be discussed in chapter 4. To understand MIMO-OFDM time synchronization methods, we begin this chapter by speaking about non MIMO-OFDM time synchronization methods. These non MIMO-OFDM time synchronization methods can be either MIMO non-OFDM methods or non-MIMO OFDM methods. After a rather long introductory

section, chapter 4 will address MIMO OFDM time synchronization issue. MIMO-OFDM time synchronization problem, as it is previously said, is divided into frame synchronization and symbol timing and each issue will be presented in a separate section.

In Chapter 5, we will present a joint ML synchronization method. Using this method time and frequency synchronization as well as channel estimation will be performed. To describe this algorithm first a signal model will be appeared. Then, the algorithm will be presented and finally the result of this algorithm will be compared to the results of the algorithms presented in the previous chapter.

Chapter 6 deals with iterative synchronization method by using EM (Expectation-Maximization) algorithm. This algorithm by using the information of data packet will improve the spectral efficiency. Also this algorithm can be served in the data-transmission mode for estimating of parameters. After a signal model, a brief description of EM algorithm will be appeared and then the EM synchronization algorithm will be explained and simulation results will show the performance of algorithm.

A multi user version of MIMO-OFDM systems is an interesting issue to consider. Therefore, in chapter 7, as a perspective, we investigate the problem of synchronization in MIMO OFDMA systems. In this chapter we will explain how the EM algorithm can be served to tackle this problem in MIMO-OFDMA systems.

### 1.3.2 Contribution

1) In Chapter 4 section 4.3.2 we propose some modifications in autocorrelation algorithm to be served as a symbol timing algorithm. This new algorithm was presented in [7]. This algorithm was originally designed for flat fading channels but we showed experimentally in [8] that this algorithm has a rather good performance in frequency selective channels.
2) In chapter 4 section 4.4.2 we propose a new symbol timing algorithm that has been presented in [5]
3) Chapter 5 is the results of our efforts to develop a ML joint time-frequency synchronization algorithm. Our initial idea that was to develop a new fine frame synchronization algorithm has been presented in [6]. Since the algorithm, in its first

version, did not consider CFO, the algorithm has been elaborated in [4] to estimate CFO. Finally, our results has been submitted in [2] as an invited paper.

4) In chapter 6 we propose a new EM MIMO-OFDM synchronization method. The results of this algorithm will be published in [9] and submitted as a journal paper in [3]

5) Chapter 7 proposes EM algorithm to tackle the problem of synchronization in OFDMA systems. The result of this algorithm will be presented in [10]. Moreover, it will be published in IEEE *transaction on wireless communication* [1].

*Chapter*


# CHANNEL ESTIMATION

## 2.1 Introduction

For channel state information (CSI) acquisition, three classes of methods are available in literature:

- Data aided methods which rely on training symbols that are known to the receiver.
- Differential ones that bypass CSI estimation by means of differential encoding
- EM and blind methods, which estimate CSI merely from the received symbols

In comparison with training based schemes, differential approaches, by design, incur performance loss [59], while blind methods typically require longer data records to converge and entail higher complexity ([75],[76],[77]). On the other hand, the insertion of known training symbols can be suboptimal and bandwidth consuming but it remains attractive especially when it decouples symbol detection from channel estimation and thus simplifies the receiver implementation, and relaxes the required identification conditions ([78]). Training symbols can be placed either at the beginning of each burst (as a preamble) or regularly through the burst.

Besides, there are some iterative channel estimation algorithms where a minimal amount of training is used to provide preliminary channel estimates. These estimates are used as side information for decoding the unknown data signals. Then, data estimates obtained via decoding are used as an uncertain training sequence in order to improve channel estimation. This process is continued in an iterative fashion until some stopping criteria is met.

In this section we investigate data aided methods both in preamble and pilot forms and also in both OFDM and non OFDM system. Then, we take a look at blind and semi blind algorithms, and finally we consider iterative channel estimation algorithm and EM approach.

Figure 2 shows the outline of the channel estimation methods.

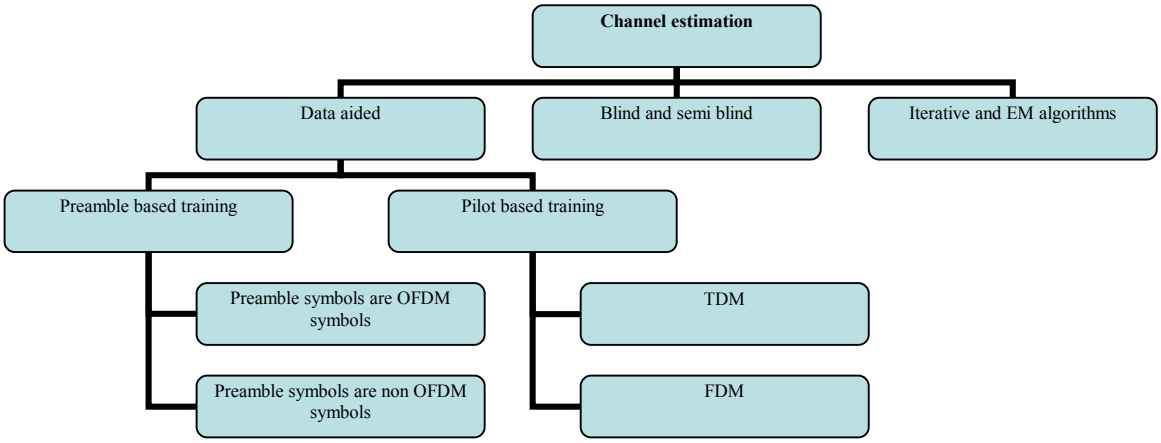


Figure 2: Outline of the channel estimation

# 2.2 Data aided methods

## 2.2.1 Preamble based training

### 2.2.1.1 Frequency domain preambles

In the packet based MIMO OFDM system in wireless local area networks (LAN), due to the low mobility in this network, a quasi static channel can be assumed for each packet. Training signals are thus needed only at the beginning of the packet as in IEEE 802.11 a [79] and Hiperlan/2. Different articles discuss about the constraints over this training sequence, such as low peak-to-average power-ratio (PAPR) [80], easy time and frequency synchronization properties, DC offset, the problem of using subcarriers at high frequencies [61] and so on. There are several papers which discuss about OFDM training sequence design [60] [61] [62]. Anyway, a very important question concerning OFDM training sequence is that if we use the preambles in the frequency domain (i.e. before applying IFFT) or in the time domain (i.e. after applying IFFT), in other words, if our preamble are itself a OFDM symbol or not. Here, we take a look at [62] as a frequently cited reference that uses an OFDM symbol as a preamble:

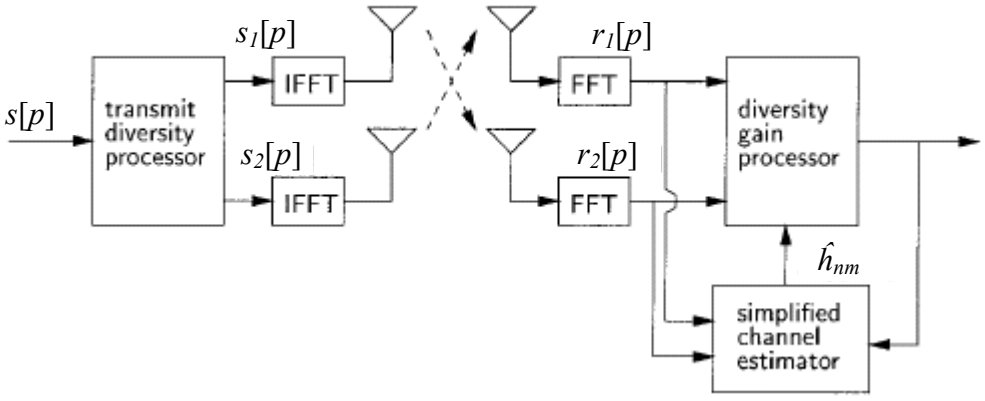


Figure 3: System diagram for a channel estimator

Considering an OFDM system with two transmit and two receive antennas. For every OFDM symbols, a data block of length $2N_c$ represented by $\{s[p]: p=0,1,\ldots,2N_c-1\}$ is transformed into two data blocks of size $N_c$ using a convenient space-time coding : $\{s_n[p]: p=0,\ldots,N_c-1$ and

$m$=1,2.}. These two blocks are then modulated using OFDM signaling scheme to generate two OFDM symbols which propagate simultaneously through lowpass equivalent channel (see Figure 3). In these representations, $N_c$, $p$ and $m$ are the number of subchannels of the OFDM systems, subchannel (or tone) index and antenna index, respectively (for simplicity the index for number of OFDM symbol has been omitted). The frequency domain signal received at the $m^{\text{th}}$ receive antenna after can be expressed as:

$$r_m[p] = \sum_{i=1}^{2} H_{nm}[p] s_n[p] + w_m[p] \tag{1}$$

where $w_m[p]$ denotes the additive zero mean complex Gaussian noise with variance $\sigma^2_w$ for $p^{th}$ tone and at the $m^{\text{th}}$ receive antenna. $H_{nm}[p]$ is the channel complex coefficient in frequency domain for $p^{\text{th}}$ tone between $n^{\text{th}}$ and $m^{\text{th}}$ transmit and receive antenna. These coefficients can be calculated from the time domain channel taps as:

$$H_{nm}[p] = \sum_{l=0}^{L-1} h_{nm}(l) W_{N_c}^{lp} \tag{2}$$

where $L$ is the number of nonzero taps of the channel impulse response. $W_{N_c}^{l}$ is equal to $e^{-\frac{2\pi j}{N_c}}$ Hence, to obtain $H_{nm}[p]$, we only need to estimate $h_{nm}(l)$. Supposing $\hat{h}_{nm}(l)$ the temporal estimation of $h_{nm}(l)$, it is shown [82] that it can be found by

$$\begin{pmatrix} \hat{\mathbf{h}}_{1m} \\ \hat{\mathbf{h}}_{2m} \end{pmatrix} = \begin{pmatrix} \mathbf{Q}_{11} & \mathbf{Q}_{21} \\ \mathbf{Q}_{12} & \mathbf{Q}_{22} \end{pmatrix}^{-1} \begin{pmatrix} \mathbf{u}_{1m} \\ \mathbf{u}_{2m} \end{pmatrix} \tag{3}$$

where $\hat{\mathbf{h}}_{nm}$ is defined as $\hat{\mathbf{h}}_{nm} = [\hat{h}_{nm}(0), \ldots, \hat{h}_{nm}(L-1)]^T$ and, $\mathbf{Q}_{nn'}$ and $\mathbf{u}_{mn}$ are defined as:

$$\begin{aligned} q_{nn'}(l) &= \sum_{p=0}^{N_c-1} s_n[p] s_{n'}^*[p] W_{N_c}^{-pl} \\ \mathbf{Q}_{nn'} &= \left[ q_{nn'}(l_1 - l_2) \right]_{l_1, l_2 = 0}^{L-1} \\ u_{nm}(l) &= \sum_{p=0}^{N_c-1} r_m[p] s_n^*[p] W_{N_c}^{-pl} \\ \mathbf{u}_{nm} &= \left[ u_{nm}(0), \ldots, u_{nm}(L-1) \right]^T \end{aligned} \tag{4}$$

As it can be seen in (3), the problem of channel estimation is reduced to a $2L \times 2L$ matrix multiplication. Since the preamble is known and the fact that the **Q** matrix depends only on the preamble, the inversion operation can be done once and the inverse matrix is stored in receiver.

The vectors $\mathbf{u}_{nm}$ are the functions of the received signals and the training symbols. It is seen that here is no constraint over the preamble. However, with a clever design of preamble, the matrix multiplication can de done more simply. In [81], the authors propose significant-tap-catching method (STC) in order to reduce the computational complexity. In this method, only the more significant coefficients are identified and calculated and it is shown that the degradation is not significant.

There is another method to reduce the complexity proposed in [62]. It proposes that the preamble for the first transmit antenna be chosen arbitrary. The preambles of other transmit antennas for a system with $N_t$ transmit antennas are a simple phase shifted of the first one and can be calculated as:

$$s_n[p] = s_1[p] W_{N_c}^{-\bar{L}(n-1)p} \tag{5}$$

where $\bar{L} = \lfloor N_c / N_t \rfloor \geq L$ and $\lfloor x \rfloor$ denotes the largest integer no larger than x. With this constraint on the preambles, we obtain the following interesting result:

- $\mathbf{Q}_{ij} = 0$ for all $i \neq j$
- $\mathbf{Q}_{ii} = N_c \mathbf{I}$

The condition required is that $N_t$ should be less than or equal to $N_c/L$. Using equation (3):

$$\hat{h}_{nm}(l) = \frac{1}{N_c} u_{nm}(l) \tag{6}$$

It is easy to show that $\hat{h}_{nm}(l)$ can be written as a function of $u_{1\mathrm{m}}$ alone:

$$\hat{h}_{nm}(l) = \frac{1}{N_c} u_{1m}(l - (n-1)\bar{L}) \tag{7}$$

Figure 4 depicts calculated sequences $u_{n1}(l)$ for 4 transmit antennas. As it can be seen, by carefully selecting the relative phase between the training sequence for different transmit antennas, the effect of the channels, corresponding to different transmit antennas on $u_{n1}(l)$ is shifted to different region. So channel estimation complexity is reduced to the calculation of $\mathbf{u}$.

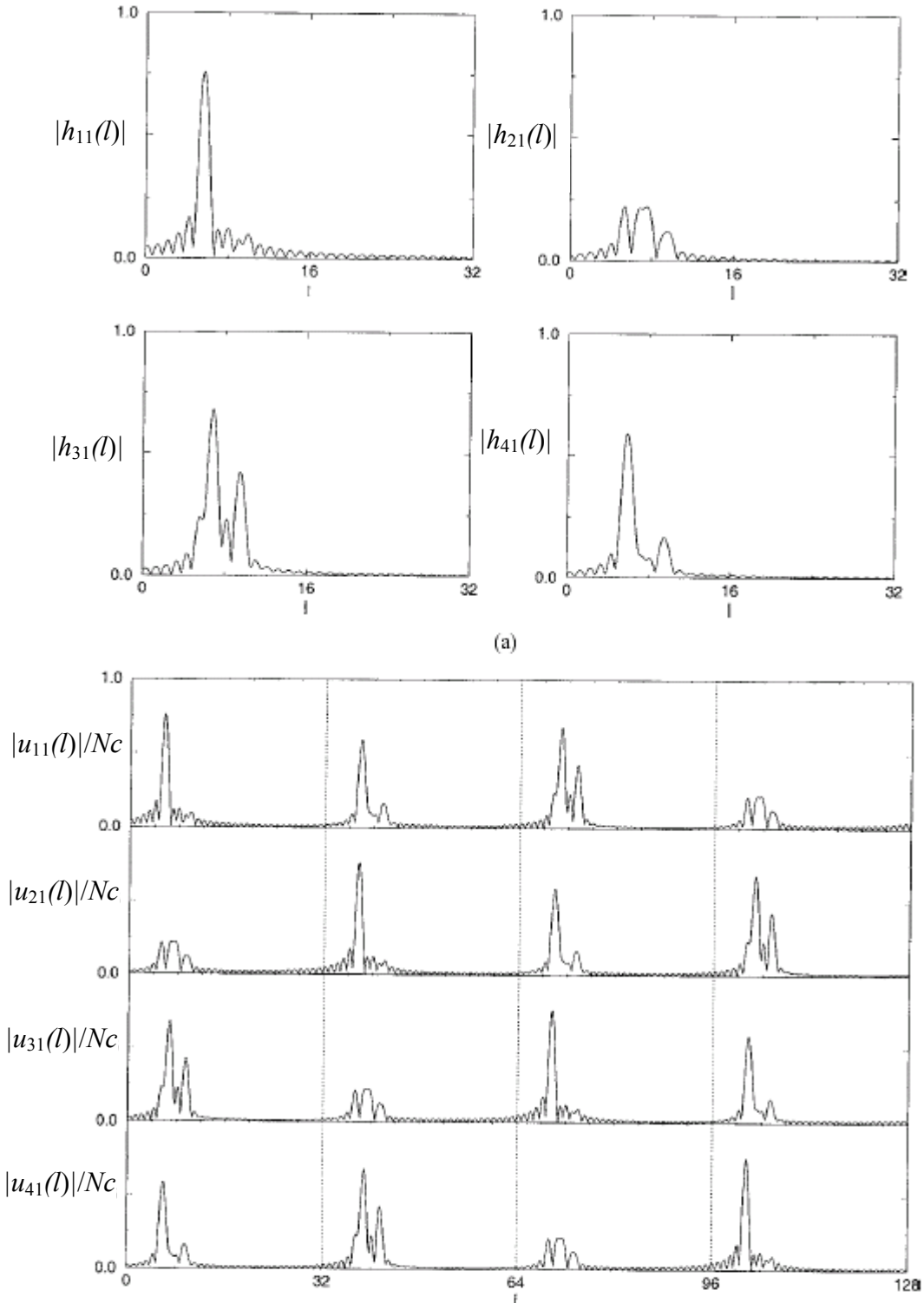


Figure 4 : $u_{n1}$ ($l$) for OFDM with optimum training sequence [62]

Moreover, Geoffrey Li [62] presented some modification in his algorithm to be able to track the variation of channel. In fact he used a simplified algorithm to track channel variation without using pilot symbols (Figure 3):

By using the same notation as before, for systems with constant modulus modulation, $\mathbf{Q}_{ii} = N_c\mathbf{I}$ we have:

$$\hat{\mathbf{h}}_{1m} = \frac{1}{N_c}(\mathbf{p}_{1m} - \mathbf{Q}_{21}\hat{\mathbf{h}}_{2m})$$
$$\hat{\mathbf{h}}_{2m} = \frac{1}{N_c}(\mathbf{p}_{2m} - \mathbf{Q}_{12}\hat{\mathbf{h}}_{2m}) \quad (8)$$

From the above equation, if $\hat{\mathbf{h}}_{2m}$ is known, then $\hat{\mathbf{h}}_{1m}$ can be estimated without any matrix inversion. However, neither $\hat{\mathbf{h}}_{2m}$ nor $\hat{\mathbf{h}}_{1m}$ is known but we can use the channel estimation result at previous time i.e. $\hat{\mathbf{h}}_{2m}^{(k-1)}, \hat{\mathbf{h}}_{1m}^{(k-1)}$ to substitute $\hat{\mathbf{h}}_{2m}, \hat{\mathbf{h}}_{1m}$ in the right sides of (8). This substitution reduces the computational complexity of the channel estimation [62]. It is claimed in this paper that the performance degradation of the simplified estimation is negligible when the square of normalized Doppler frequency is much less than the noise power. Furthermore, the simplified estimator does not require a large matrix inversion; therefore, it has lower complexity and also is numerically stable, so we can use it in transmission mode.

## 2.2.1.2 Time domain preambles

Using time domain preambles, the same methods are used for both OFDM and non-OFDM systems, that is we use either MMSE estimation or Least Square estimation as follows:

$$\hat{\mathbf{h}}_{MMSE} = (\mathbf{s}^H\mathbf{s} + \sigma_w{}^2\mathbf{I})^{-1}\mathbf{s}^H\mathbf{r} \quad (9)$$

$$\hat{\mathbf{h}}_{LS} = (\mathbf{s}^H\mathbf{s})^{-1}\mathbf{s}^H\mathbf{r} \quad (10)$$

In these equations

- **h** is a matrix of size $LN_t\ N_r \times 1$ where $N_t$ and $N_r$ are the number of TX and RX antennas respectively and $L$ is the length of multi path channel.
- **s** is an $N_c \times LN_t$ matrix generated from all the samples in one cyclic extended OFDM symbol sent by all the TX antennas and all the $N_c$ sub carriers
- **r** is an $N_r N_c \times 1$ matrix containing the received signals of all the RX antennas for one OFDM symbol.
- $\sigma_w$ is the variance of Gaussian noise

Space–time codes exploit the spatial diversity of multiple antennas to increase the capacity of wireless links and thus satisfy current demands for higher data rates. The receivers observe the

superposition of training sequence transmitted through different channels. The preamble based training sequences that achieve minimum mean square error have an impulse like autocorrelation sequence and additionally zero cross-correlation. This last property makes the channel estimation problem different for multiple antenna systems from single antenna system. Training based estimation for a single-input single-output frequency selective channel has been widely investigated in the literature (see for example [82] and the reference therein). For frequency selective channels, a straightforward method to achieve zero cross-correlation is to transmit training symbols only from one antenna at time, while not transmitting from the other antennas, as Figure 5 demonstrates for a two transmit antenna system. This approach results in high pick to average power ratio and hence is undesirable in practice.

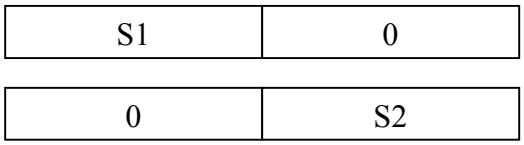


Figure 5: A single training sequence transmission scheme for two antennas that achieves zero cross-correlation

The construction of training sequence with constant envelope can be classified in two main categories according to the training sequence alphabet size $N$. the first approach [83] constructs optimal sequence from an $N^{th}$ root of unity alphabet $A_N=\{\exp(i2\pi k/N)\}$, $k=0,1,\ldots,N-1$, without constraining the alphabet size $N$. Such sequences are the perfect roots-of-unity sequences or polyphase sequences that have been proposed in the literature for different applications (see [84] and the references therein). For any training sequence length $N_p$, there exist optimal training sequences that belong to an $N^{th}$ root-of-unity alphabet. The training sequence length $N_p$ determines the smallest possible alphabet size. Chu [85] shows that for any length $N_p$, there is a perfect roots-of-unity sequence with alphabet size $N=2N_p$, and Mow [86] shows that for some $N_p$ smaller alphabet sizes are possible.

The second approach in the literature constrains training sequence symbols to belong to a specific constellation, typically BPSK or QPSK, to have a simpler transmitter/receiver implementation

[87]. In this case optimal sequence does not exist for all training length $N_p$. Instead, exhaustive searches can identify suboptimal sequence according to some performance criteria.

The training sequence best suited to particular applications depends on the training sequence length $N_p$, (which for standardized systems is predetermined), the number of channel taps to estimate, and the signal constellation used. Perfect roots-of-unity sequences of a predetermined length may not belong to a standard constellation, while exhaustive searches are, in many cases, computationally prohibitive. For a system with $N_t$ transmit antennas over frequency selective channels with L taps each, an exhaustive search must identify $N_t$ training sequence. Fargouli in [74] presented some way to reduce the amount of exhaustive search. Her main idea is to reduce problem of designing multiple training sequence with impulse-like autocorrelation and zero cross correlation to designing a single truing sequence with impulse-like autocorrelation. In fact the restriction from multiple to a single training sequence makes exhaustive searches more practical and thus facilitates the identification of good sequences. In some case no search is necessary since optimal sequences are available from the literature.

### 2.2.2 Pilot based training

In rapidly fading preamble based training may not work well and this may motivate embedding training sequence in each transmitted block, instead of concentrating them in the preambles. It is obvious that by using more training symbols, we achieve less MSE for channel estimation but on the other hand, given a constant burst length, the use of training sequence reduces the information symbol count. So there is a trend to link training sequence discussion with channel capacity or it's bound. For *flat fading channel*, Hassibi in [63] showed how training affects the capacity of a fading channel. With too little training the channel will be estimated improperly. On the other hand, with too much training there is no time left for data transmission before the channel changes. The question then is the relation between capacity which can be obtained from a MIMO channel and the number of symbols attributed to training sequence. We will discuss about optimum amount of training sequence based on [63] and [64]. We will do it in this subsection as a preface to pilot design discussion. Then, we will concentrate on the methods to design training and to arrange them in the burst. This will be done for OFDM and non OFDM system separately in two following subsections.

It is shown that the minimum and sufficient (optimum) number of pilot symbols (training sequence length) is equal to the number of transmitter antennas [63]. This is correct if optimization over the training and data powers is allowed, i.e., one can use different transmit power for data and pilots. In the contrary, if the training and data powers are required to be equal, then the optimal number of training symbol should be larger than the number of antennas. Figure 6 displays the lower bound of capacity obtained as a function of block length $T$ for $N_t$=$N_r$=10. $N_t$ is the number of transmit antennas, $N_r$ is the number of receive antennas, $\rho_t$ and $\rho_d$ are the SNR during the training and data transmission respectively. The larger the block is, the ratio of data over pilot is more important, so we achieve to more capacity. As it can be seen by comparing the curves, a power control over pilot symbols can increase 10-15% the capacity achieved. Figure 7 displays the optimal number of pilots ($N_p$) in respect of the length of the block when we can not adjust the power of pilots and we use the same power for pilots and the data. We see the trend that as the SNR decreases the amount of training increases. It can be shown that in SNR equal to zero the training amounts increases until it reaches to half of the packet ($T$/2).

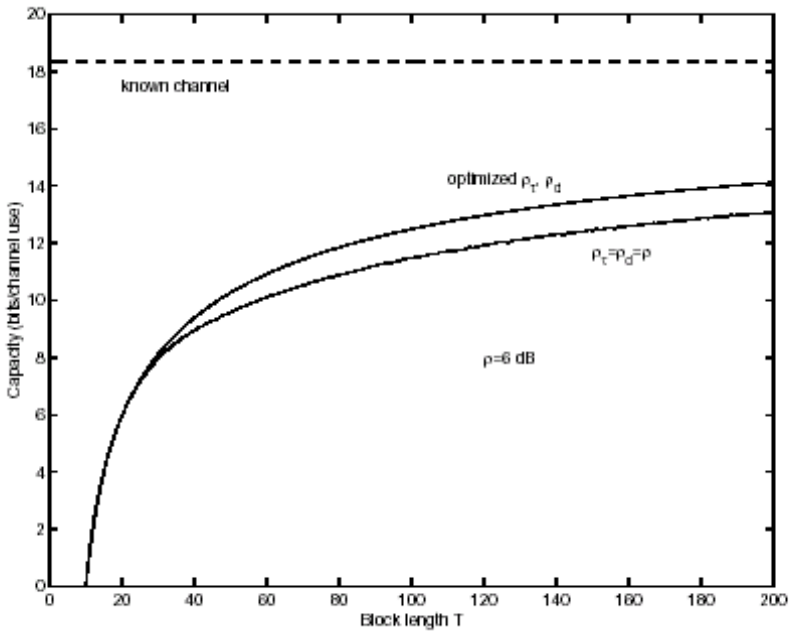


Figure 6: The lower bound in capacity as a function of $T$ ($N_t$=$N_r$=10)

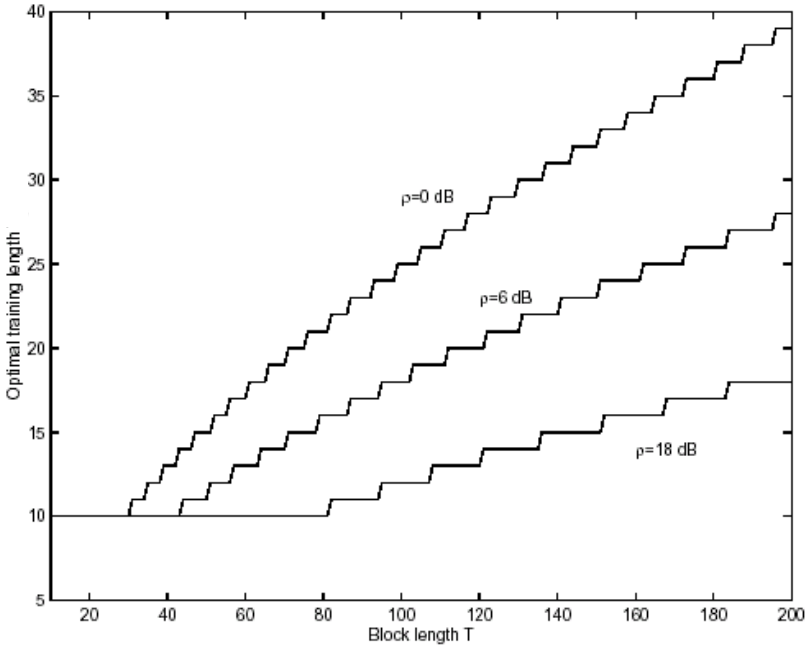


Figure 7: The optimal value of $N_p$ ($N_t$=$N_r$=10)

X. Ma in [64] showed that for a system with $T$=69, $N_r$=3, $L$=6 ($L$ is the channel length) in a frequency fading channel, when $3 \geq N_r \geq 1$, the average capacity increase with the number of transmit antennas (Figure 8). However when $N_r$>3, the average capacity starts decreasing. This is because training symbols take over larger and larger portion of the whole transmitted block, leaving less and less to be used for transmitting information symbols.

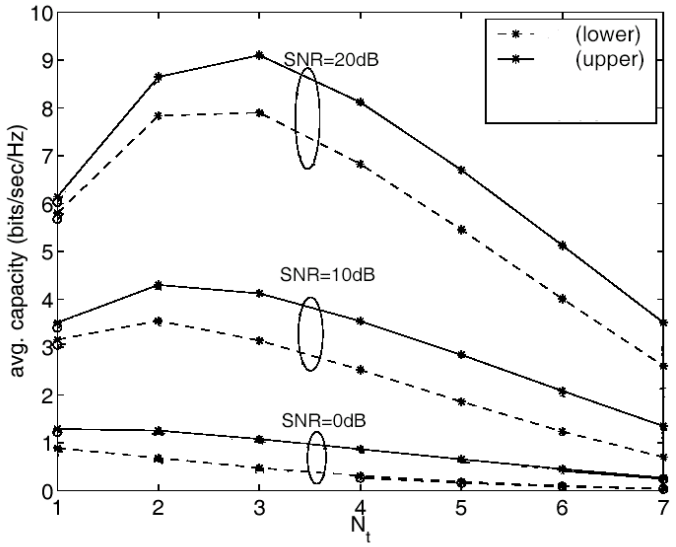


Figure 8: Average capacity bound with increasing number of transmit antennas

Therefore, we found that choosing optimum number of pilots have to be done carefully and also we saw that if we want use the pilots inside the packets in opposition of common idea that increasing transmitter antenna will end to more performance, there is an optimum amount for the number of transmitters.

In the following section we will take a look at the pilot design in OFDM and non-OFDM systems. Note that in two following subsection we suppose that $h_{nm}(l)$, $l \in [0,L]$ is the discrete-time baseband equivalent channel that includes transmit-received filters as well as the frequency-selective propagation effects between the $n^{th}$ transmit antenna and $m^{th}$ receive antenna. With $\tau_{nm}$ denoting the delay spread between the $n^{th}$ transmit antenna and $m^{th}$ receive antenna, we define the maximum delay spread as $\tau_{max}=\max\{\tau_{nm}\}$, the maximum channel order is then given by $L:=[\tau_{max}/T_s]$ where $T_s$ is symbol time.

## 2.2.2.1 Pilot base training for MIMO non-OFDM systems

Suppose a general multi-antenna system that employs $N_t$ transmit and $N_r$ receive antennas (Figure 9). At the transmitter, the information bearing symbols having a symbol rate $1/T_s$ are parsed into blocks of size $N_{data}\times 1$. Each block $\mathbf{s}$ is first encoded and/or multiplexed in space and time. The resulting $N_t$ blocks $\mathbf{c}_{n,qN_{data}} = \left[ c_{n,qN_{data}} \;\; c_{n,qN_{data}+1} \;\; \ldots \;\; c_{n,qN_{data}+N_{data}-1} \right]^T_{n=1,\ldots,N_t}$ (where $c_{n,k}$ is transmitted at the $k^{th}$ instant and from $n^{th}$ transmit antenna) have length $N_{data}$ and each is directed to one transmit antenna).

If we want to consider the training symbols in the most general form we can suppose that at each $n^{th}$ transmit antenna $\mathbf{c}_{n,qN_{data}}$ is proceed by a matrix $\mathbf{A}_n$, so the vector of $\mathbf{c}_{n,qN_{data}}$ with length $N_{data}$ is extended to $\mathbf{A}_n\mathbf{c}_{n,qN_{data}}$ with length $N_c$ by matrix $\mathbf{A}_n$ ( $N_c\times N_{data}$ ). Training blocks $\bar{\mathbf{b}}_{n,qNc}$ of length $N_c$, which are known to both transmitter and receiver, are then added to $\mathbf{A}_n\mathbf{c}_{n,qN_{data}}$. Now (11) is a general term for embedding pilots imposed to the data:

$$\bar{\mathbf{u}}_{n,qNc} = \mathbf{A}_n\mathbf{c}_{n,qN_{data}} + \bar{\mathbf{b}}_{n,qNc} \qquad (11)$$

Where $\bar{\mathbf{u}}_{n,qN_c} = \left[ \bar{u}_{n,qN_c} \;\; \bar{u}_{n,qN_c+1} \;\; \ldots \;\; \bar{u}_{n,qN_c+N_c-1} \right]^T$ and $\bar{\mathbf{b}}_{n,qN_c} = \left[ \bar{b}_{n,qN_c} \;\; \bar{b}_{n,qN_c+1} \;\; \ldots \;\; \bar{b}_{n,qN_c+N_c-1} \right]^T$

This affine model is fairly general: it encompasses linear precoding via $\mathbf{A}_n$ as well as inserted training symbols by having non zero entries of $\overline{\mathbf{b}}_{n,qNc}$ where $\mathbf{A}_n\mathbf{c}_{n,qN_{data}}$ has zero entries.

Because of channel length we have Inter Block Interference IBI between two adjacent block $\overline{\mathbf{u}}_{n,qN_c}$ and $\overline{\mathbf{u}}_{n,(q+1)N_c}$ so we have to add something like a guard interval to eliminate IBI. One way to separate each block $\overline{\mathbf{u}}_{n,qN_c}$ from the adjacent one is to insert sufficiently zeros ($L$) at the end of each block.

For this purpose we can consider a matrix **T** which adds redundant (guard) symbol to $\overline{\mathbf{u}}_{n,qN_c}$.

In fact we can multiply **T** by $\overline{\overline{\mathbf{u}}}_{n,qN_c}$ to model guard interval adding and of course in the receiver we need to remove this added guard interval. We can model it by multiplying matrix **R** in the received vectors (Figure 9). To achieve this, we can select the matrices **T** and **R** as follows:

$$\mathbf{T} = \left[\mathbf{I}_{N_c}\ \mathbf{0}_{N_c\times L}\right]^T, \mathbf{R} = \mathbf{I}_{N_c+L} \tag{12}$$

To estimate **h**, from **r** (the received vector after removing guard interval) we can use linear MMSE estimator as follows:

$$\hat{\mathbf{h}} = (\sigma^2\mathbf{R}_h^{-1} + \mathbf{I}_{N_r} \otimes (\overline{\mathbf{B}}^H\overline{\mathbf{B}}))^{-1}(\mathbf{I}_{N_r} \otimes \overline{\mathbf{B}}^H)\mathbf{r} \tag{13}$$

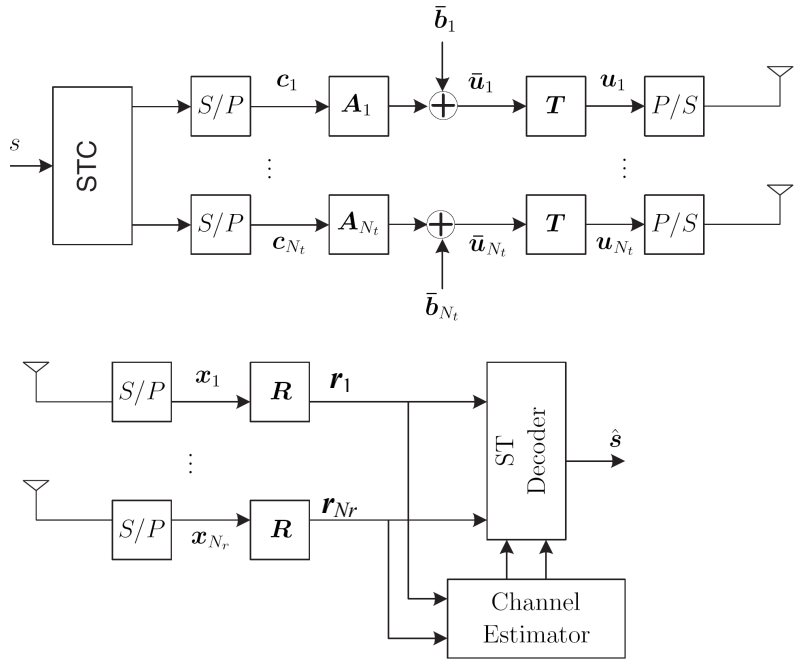


Figure 9: Discrete time baseband equivalent Tx-Rx MIMO model

where $\mathbf{R}_h$=E[$\mathbf{hh}^H$] (E(.) means the expected value of (.) )and $\bar{\mathbf{B}} = [\bar{\mathbf{B}}_1,...,\bar{\mathbf{B}}_{N_t}]$ which $\bar{\mathbf{B}}_n$ is an $N_c$+$L$ × ($L$+1) Toeplitz matrix having first column $[\bar{b}_{n,qN_c},...,\bar{b}_{n,qN_c+N_c-1},0,...,0]^T$ and $\hat{\mathbf{h}}$ is an estimation of $\mathbf{h} := [\mathbf{h}_{:1}^T ... \mathbf{h}_{:N_r}^T]^T$ with $\mathbf{h}_{:m} = [\{h_{0m}(0)...h_{1m}(L)\}^T ... \{h_{N_t m}(0)...h_{N_t m}(L)\}^T]^T$. Besides, $\mathbf{r}_{m,qN_c} = \left[ r_{m,qN_c} \;\; r_{m,qN_c+1} \;\; ... \;\; r_{m,qN_c+N_c-1} \right]^T$ and $\mathbf{r} = \left[ \mathbf{r}_{1,qN_c}{}^T \;\; \mathbf{r}_{2,qN_c}{}^T \;\; ... \;\; \mathbf{r}_{N_r,qN_c}{}^T \right]^T$

X. Ma in [64] has discussed that it is better to decouple channel estimation from symbol decoding. She argues that without being decoupled we encounter the following problems:

- The nonlinear search required is not only computationally complex but also its convergence to a globally optimum solution is not guaranteed.
- Symbol identifiability can not be ensured in general.
- The designed training sequence will be tailored for only a specific encoder.

Since in the r there are unknown symbols in addition to the training sequence we can not use (13) to estimate channel. In [64] it has been proved that in order to decouple channel estimation from symbol decoding, the superimposed model $\mathbf{u}_{n,qNc} = \mathbf{T}\left(\mathbf{A}_n \mathbf{c}_{n,qN_{data}} + \bar{\mathbf{b}}_{n,qNc}\right)$ results to time division multiplexing (TDM) of the ST blocks $\mathbf{c}_n$ with the training symbol blocks $\mathbf{b}_n$.

$$\mathbf{A}_n = \begin{bmatrix} \boldsymbol{\Theta} \\ \mathbf{0}_{N_c - N_{data} \times N_{data}} \end{bmatrix} \qquad \bar{\mathbf{b}}_{n,qN_c} = \begin{bmatrix} \mathbf{0}_{N_{data}+L} \\ \mathbf{b}_n \end{bmatrix} \tag{14}$$

where $\mathbf{b}_n$ contains all the non zero entries of $\bar{\mathbf{b}}_{n,qNc}$ and $\boldsymbol{\Theta}$ is an $N_{data} \times N_{data}$ matrix that optionally precodes (if $\boldsymbol{\Theta} \neq \mathbf{I}_{Ndata}$) the information bearing block $\mathbf{c}_{n,qN_{data}}$ linearly. Certainly TDM ensures decoupling, but the inverse is not obvious and it has been proved in [64].

Besides, in order to minimize channel mean square error it can be proved that the training sequence across transmit antennas have to be orthogonal. The optimal parameters of training sequence such as training length , power allocation has been derived in [64] by maximizing channel capacity when the source data is Gaussian and remains Gaussian after ST-mapper also (in ST-mappers as BLAST type multiplexer, block codes and even other codes when the block size is large). The optimal number of training sequence is $N_t$($L$+1) and the optimal power allocation factor is : $\frac{\sqrt{N_{data}}}{\sqrt{N_{data}} + \sqrt{N_t(L+1)}}$. Figure 10 shows the optimal training sequence which purposed by [64].

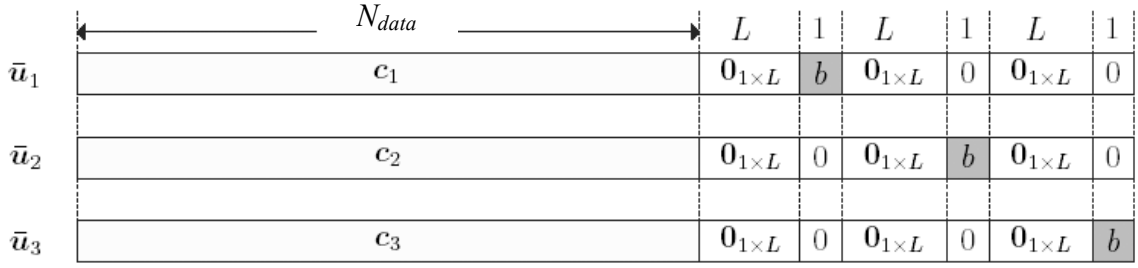


Figure 10: Training scheme example ($N_t$=3)

## 2.2.2.2 Pilot base training for OFDM systems

We use the same model as previous section (Figure 9). There is only one difference for the OFDM systems for separating different block in multi path channel. Here instead of having zeros at the end of each block, cyclic prefix is used. In fact instead of adding $L$ zeros at the end of each block $\mathbf{u}_n$, in previous model an alternative way to eliminate IBI is by adding a cyclic prefix (CP) of length $L$ at the transmitter, and removing it at the receiver, so instead of (12) we have:

$$\mathbf{T} = \begin{bmatrix} \mathbf{I}_{CP} & \mathbf{I}_{N_c} \end{bmatrix}^T, \quad \mathbf{R} = \begin{bmatrix} \mathbf{0}_{N_c \times L} & \mathbf{I}_{N_c} \end{bmatrix}^T \tag{15}$$

where $\mathbf{I}_{CP}$ contains the last $L$ columns of $\mathbf{I}_{Nc}$.

In fact in an OFDM system we have the same consideration as previous section as to decouple channel estimation from symbol detection. We can estimate channel linear MMSE estimator as (13). Here in OFDM systems Ma has proved that for decoupling channel estimation from symbol detection instead of equation (14) we have to respect this criterion:

$$\mathbf{A}_n = \mathbf{F}_{N_c}^H \mathbf{P}_A \boldsymbol{\Theta}, \bar{\mathbf{b}}_n = \mathbf{F}_{N_c}^H \mathbf{P}_b \mathbf{b}_n \tag{16}$$

Where $\mathbf{F}$ is Fourier matrix, $\boldsymbol{\Theta}$ is a $N_{data}\times N_{data}$ matrix that optionally precodes (if $\boldsymbol{\Theta} \neq \mathbf{I}_{td}$) the information bearing block $\mathbf{c}_{n,qN_{data}}$ linearly. $\mathbf{b}_n$ consists of the $L$ possibly non zero training symbol from the $i^{\text{th}}$ transmit antenna, and permutation matrices $\mathbf{P}_A$, $\mathbf{P}_b$ satisfy $\mathbf{P}_A^T \mathbf{P}_b = \mathbf{0}_{N_{data}\times(N_c - N_{data})}$ in fact the permutation matrices play the role of assigning carriers to each (possibly precoded) information symbol and training symbol (pilot tones). As a result, in order to decouple the channel estimation and symbol detection the superimposed model boils down to frequency division multiplexing (FDM) the ST codewords $\mathbf{c}_{n,qN_{data}}$ with the training blocks $\mathbf{b}_n$. Also according to [64] in order to minimizing MSE one can result that one pilot tone can not be shared

by more than one non zero training symbols on different transmit antennas in other words, the non zero pilot tones across the $N_t$ transmit antenna are mutually orthogonal. In this case also like non OFDM system as the number of non-zero pilot tones increases, the capacity lower bound decreases. Therefore, the number of non-zero pilot tones should be chosen to the minimum possible i.e. $L$+1 per transmit antenna. The optimal power allocation factor is $\frac{\sqrt{N_{data}}}{\sqrt{N_{data}}+\sqrt{N_t(L+1)}}\frac{N_c}{N_{data}}$ where the length of cyclic prefix is $N_c$-$N_{data}$. In [65] by optimizing a lower bound on channel capacity, the optimal amount of resource allocated to the pilot symbols have been derived in varying fading Rayleigh channel. Figure 11 shows an optimal training sequence which purposed by [64].

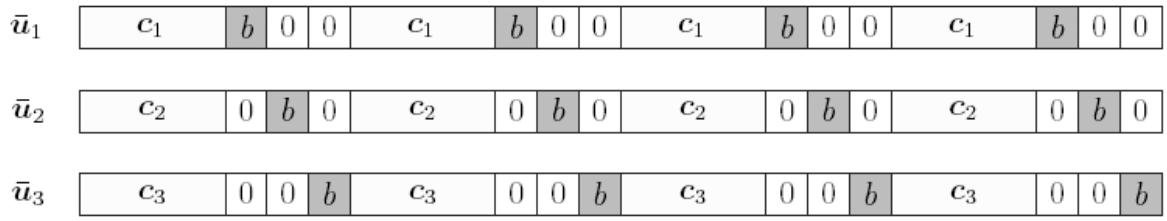


Figure 11: Training scheme example ($N_t$=3, $L$=3)

## 2.2.3 Comparison between preamble based training and pilot based training

Here by using [64] we are going to compare pilot base training (section 2.2.2.2) and preamble base training 2.2.1.1). Figure 12 shows that with the same power per training block, in quasi static channels, both of them achieve similar channel MSE.

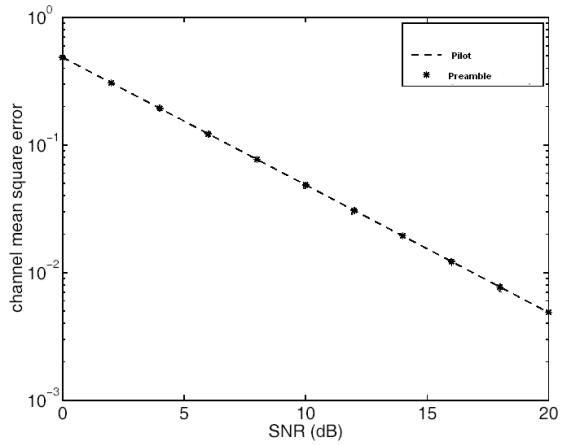


Figure 12: Channel MSE comparison ($N_t$=2,$N_r$=1 and $L$=6)

Figure 13 plot the channel MSE for the three cases when the channel is slowly time-varying. Each channel tap is generated by Jakes' model with a terminal speed of 3, 5, 10 m/s, and a carrier frequency of 5.2 GHz. The variances of channel taps satisfy the exponential power profile. To maintain the same information rate, every training block is followed by four information blocks for the preamble scheme of 2.2.1.1 .Figure 35 shows the channel MSE versus the number of transmitted blocks for the two schemes at SNR= 10dB. Although the channel is slowly time-varying, the MSE of the scheme of section 2.2.2.2 remains invariant from block to block, while that of 2.2.1.1 increases very fast. Note that the scheme of 2.2.1.1 yields smaller MSEs at the beginning of each re-training burst, because the total transmitted power is used for training.

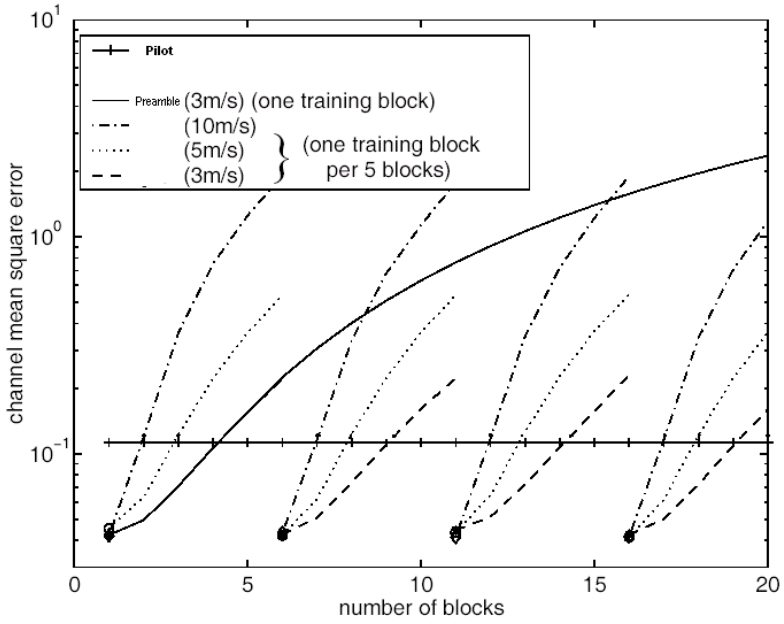


Figure 13: Comparison of the channel MSE ($N_t$=2,$N_r$=1 and $L$=6)

## 2.3 Blind and Semi blind methods

Existing blind channel estimation methods for OFDM system can be classified as pre DFT (time domain) methods post DFT (frequency domain) methods. Pre DFT methods explore the cyclostationarity that the cyclic prefix induces to the transmitted signal. They recover the channel using cyclic statistics of the received signal [73] or subspace decomposition of the correlation matrix of the pre DFT received blocks [93].

The post DFT methods process the post DFT received blocks, and exploit the finite alphabet property of the information bearing symbol. Besides several papers which apply a transformation to blocks of symbols before they enter the OFDM system, and they perform channel blind estimation at the output of the OFDM system via cross correlation operations ([66] and its references).

Also there is some articles which use the redundancy of space-time coded signals to apply subspace method in [94] in order to channel identification [67] [95].

In semi blind channel estimation we encounter two approaches. In the first approach a combination between data aided methods and blind methods is used. In fact pilot symbols as known symbols used to estimate the channel coefficients but since there are a number of channel coefficients, a large number of pilot symbols may be required. It would result in a decrease of data throughput. To avoid it, the semi blind channel estimation methods with fewer pilot symbols can be used. For example Choi in [67] by using a $L \times N_t$ pilot symbol (the minimum number of know data to avoid ambiguity) present a semi blind method that improve channel estimation MSE in comparison with using equal pilots in LS (Least square) method. Although there are other approaches for the semi blind method as in [96].

In the second approach in semi blind methods in MIMO channel estimation, there is a trend to use superimposition of pilot and data symbols[68][69]. In fact, these methods by superimposing pilot and data symbols in the same time economize the bandwidth.

Since the blind channel estimation is time-consuming and complicated, it seems that for practical purpose it can not be suitable.

## 2.4 Iterative channel estimation

The channel estimates can be improved if the estimation process is included in the iterative receiver chain, and the channel is re-estimated in several iterations, using not only the training, but also information on the data symbols obtained from decoder in the feed back loop. Such iterative estimation of the frequency flat fading channel in SISO system is considered in [97][98]. A solution to the corresponding problem in MIMO system is proposed in [71], where the channel estimate is updated in each iteration using hard decisions on the data symbols obtained from the decoder. For getting the initial estimation of channel one can use the TDM pilot symbols; however there are some other solutions like using pilot embedding as in [73]. In fact [73] propose transmission of pilot symbols that are transmitted along with data. These pilots, which are at a level much lower than signal level, are used to obtain the initial coarse estimates of the channel. The idea of pilot is transmitting of pilots symbol The idea of pilot embedding was first proposed in [99] in the context of single input single output single carrier systems and has been extended to multicarrier systems [100]. The initial work in [100] uses pilot symbols to obtain an initial estimate of the channel so that subsequently the receiver can work in a decision directed mode for tracking. In [100] pilot symbols are used to initialize an iterative joint channel estimation and data detection loop in an OFDM system. [73] extends the idea of pilot embedding to ST turbo coded systems.

Using Expectation-Maximization (EM) algorithm is a good way to profit the information of data symbols and perform channel estimation iteratively. In [124]-[128] one can find the general explanation and discussion about EM-algorithm and its application to the estimation problem. Xie in [130] propose two EM type channel estimation algorithm in OFDM system to convert a multiple-input channel estimation problem into a number of single-input channel estimation problems, a much more palatable problem. EM algorithm we calculate channel in two steps: At the E-step it estimates the corresponding component in the received signal for each of the OFDM links. At the M-step, as in the conventional OFDM scheme, it divides the corresponding component by the reference symbol (either known from training, or previously decode symbols) in the frequency domain and performs an inverse fast Fourier transform (IFFT) to obtain an updated estimate of channel impulse response. For high speed wireless data packet applications, if a data packet is short compared to the channel coherence time, channel fading can be assumed to be the same for the whole packet, for each transmitter, one or two pilot symbol can be at the beginning of each packet. In [130] two different EM-algorithm, classical EM algorithm and SAGE algorithm, has been compared with each other in terms of convergence and it has been

shown that the convergence rate of both algorithm are unrelated to the channel delay profile and the. In [118] an iterative MIMO-OFDM receiver has been presented that use ML channel estimation based on EM algorithm.

Finally in [120] there is an interesting approach to perform joint channel estimation and frame synchronization in MIMO systems in flat fading channel based on EM algorithms. We will discuss more about the EM algorithm in chapters 6 and 7.

*Chapter*


# CARRIER FREQUENCY OFFSET ESTIMATION

## 3.1 Introduction

One of the most important tasks in a MIMO-OFDM receiver is frequency synchronization. As a matter of fact, even a very small frequency offset in a MIMO-OFDM system can destroy the orthogonality between the sub-carriers and as a result performance of the system degrades significantly. In this chapter we are to summarize the best existing methods for MIMO frequency synchronizations in unknown channels. Four recent algorithms for MIMO frequency synchronization is collected in this chapter. For each method, first we will state its specification, and then we will describe the algorithm. In the next section we perform some simulations to compare these algorithms and in conclusion we compare these algorithms with each other.

In this chapter we suppose that we have an $N_t \times N_r$ MIMO-OFDM system. The channel is frequency selective with length $L$. Each OFDM symbol is composed of $N_c$ subcarrier and $N_{cp}$ cyclic prefix.

Before simulation of our methods, we might like to have a sense about maximally tolerated error of frequency synchronization in a MIMO-OFDM system, that is, how much carrier frequency error can be left as a residue without degrading performance or with acceptable performance degradation. Concerning this subject, we investigate the relation between carrier frequency offset and bit error rate (BER) of a $2 \times 2$ STBC-OFDM system. For the simulation setup we considered an OFDM symbol with length 64 and a cyclic prefix for each OFDM symbol with length 8. Also, an exponentially decaying channel power delay profile of length 8 is used.

Figure 14 and Figure 15 represent such a relation. Figure 14 depicts the BER of a MIMO-OFDM system in terms of $E_b/N_0$ for various residual carrier frequency errors. Figure 15 represents BER of a MIMO-OFDM system in terms of carrier frequency offset for various EbN0s. In these figures, CFO or $\varepsilon$ is the multiplication of frequency error by sampling time: i.e. CFO (or $\varepsilon$) $\equiv \Delta f \times T_s$ ($T_s$ is sampling time). Based on Figure 15, one can claim that for a system with this setup the residual carrier frequency error must be less than $10^{-4}$ to have no sensible degradation over BER.

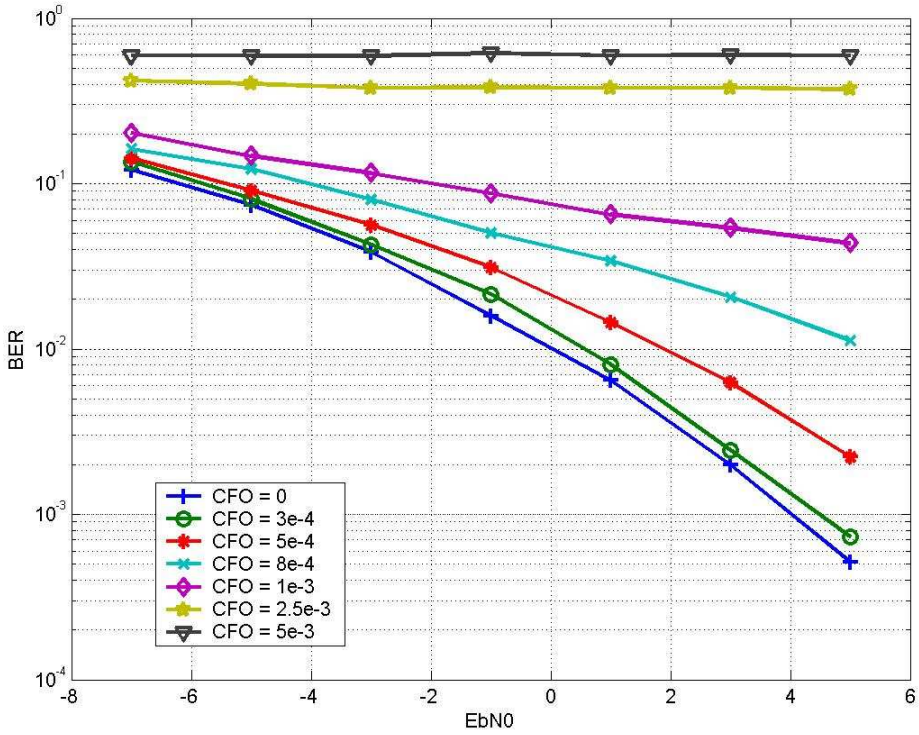


Figure 14: BER vs $E_b/N_0$ for various existing CFOs in a 2×2 STBC-OFDM system

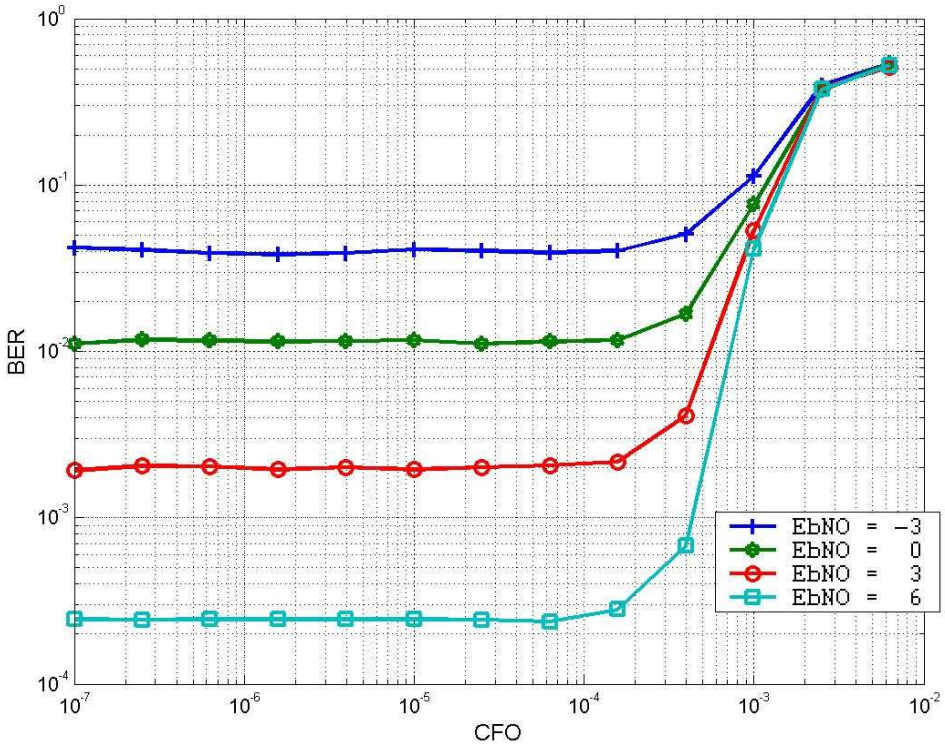


Figure 15: BER vs CFO for various $E_b/N_0$ in a 2×2 STBC-OFDM system

## 3.2 Carrier frequency offset estimation

In this section we present and compare some good carrier frequency offset estimation methods which are presented in literature or developed by us. For each method its algorithm will be described and then its specification will be outlined.

### 3.2.1 FFT method by using frequency domain training sequence

In this method appeared in [118], using FFT, carrier frequency offset can be estimated from the preamble.

#### 3.2.1.1 Description of method

Let the training sequence at the $p^{th}$ subcarrier of the $n^{th}$ transmitter be $B_n[p]$ and the received vector in frequency domain of $m^{th}$ receiver be $R_m[p]$. In [118] only a $2\times N_r$ MIMO system was considered but it seems that this method is extendable over MIMO systems for $N_t>2$. We assume that at each transmitter, K out of the total of $N_c$ OFDM subcarriers are used to transmit symbols. This is different from inserting guard interval in OFDM systems, because nothing is transmitted from the remaining ($N_c - K$) subcarriers. This truncated observation window of subcarriers is need for good estimation of the carrier frequency offset (CFO). The MIMO OFDM system model subject to CFO can be written as:

$$\begin{aligned} \mathbf{R}_m &= \mathbf{W}_N \mathbf{E} \mathbf{W}_K^I \mathbf{B} \mathbf{W}_{KL} \mathbf{h}_{:m} + \mathbf{Z}_m \\ m &= 1,\ldots,N_r \end{aligned} \tag{17}$$

with

$$\mathbf{R}_m = \left[R_m[0], R_m[1], \ldots, R_m[N_c-1]\right]^T_{N_c\times 1} \tag{18}$$

$$\mathbf{E} = diag(1, e^{j(2\pi\varepsilon/N_c)}, \ldots, e^{j(2\pi\varepsilon(N_c-1)/N_c)}) \tag{19}$$

$$\mathbf{B} = (\mathbf{B}_0, \mathbf{B}_1)_{K\times 2K} \text{ and} \tag{20}$$

$$\mathbf{B}_0 = diag(B_0[0], B_0[1], \ldots, B_0[K-1])_{K\times K} \tag{21}$$

$$\mathbf{B}_1 = diag(B_1[0], B_1[1], ..., B_1[K-1])_{K\times K} \tag{22}$$

$$\left[\mathbf{W}_N\right]_{r,s} = \frac{1}{\sqrt{N_c}} e^{-j(2\pi/N_c)rs}, \quad r = 0,1,...,N_c-1 \; ; \; s = 0,1,...,N_c-1 \tag{23}$$

$$\left[\mathbf{W}_K^I\right]_{r,s} = \frac{1}{\sqrt{N_c}} e^{j(2\pi/N_c)rs}, \quad r = 0,1,...,N_c-1 \; ; \; s = 0,1,...,K-1 \tag{24}$$

$$\mathbf{W}_{KL} = diag(\mathbf{w}_{KL}, \mathbf{w}_{KL})_{2K\times 2L} \tag{25}$$

$$\mathbf{W}_{KL} = diag(\mathbf{w}_{KL}, \mathbf{w}_{KL})_{2K\times 2L} \tag{26}$$

$$\left[\mathbf{w}_{KL}\right]_{r,s} = e^{-j(2\pi/N)rs}, \quad r = 0,1,...,K-1 \; ; \; s = 0,1,...,L-1 \tag{27}$$

$$\mathbf{h}_{:m} = \left[\mathbf{h}_{0m}^T \quad \mathbf{h}_{1m}^T\right]_{2L\times 1}^T$$
$$\mathbf{h}_{nm} = \left[h_{nm}(0), h_{nm}(1), ..., h_{nm}(L-1)\right]^T \tag{28}$$

where $h_{nm}(l), l = 0,1,...,L-1$ is the fading coefficient of the $l^{\text{th}}$ tap associated with the $n^{\text{th}}$ transmit antenna and $m^{\text{th}}$ receive antenna. $\mathbf{Z}_m$ denotes the additive with Gaussian noise at each receive antenna. The carrier frequency offset (CFO) $\varepsilon$ is introduced through the matrix **E**, since $\mathbf{E}\neq\mathbf{I}_{Nc}$ for $\varepsilon\neq 0$, orthogonality among all $N_c$ subcarriers is destroyed. **W** represents the *DFT* matrix.

In the receiver, first of all, we remove the cyclic prefix. Let the received signal after removing cyclic prefix at $m^{\text{th}}$ receive antenna in the $k^{\text{th}}$ instant be $r_m(k)$. For ML CFO estimation one have to write the ML criterion as follows:

$$\varepsilon = \arg\max_{\varepsilon}\left(\sum_{m=1}^{N_r} \log\left(p(\mathbf{R}_m \mid \mathbf{h}_m, \varepsilon)\right)\right) \tag{29}$$

Since the channel coefficients are unknown, we have to eliminate them. To this purpose, as EM algorithm, we calculate the expectation of the probability function over **h** i.e.

$$\varepsilon = \arg\min_{\varepsilon}\left(\sum_{m=1}^{N_r} E_{\boldsymbol{h}_m|\mathbf{R}_m,\varepsilon}\left\{\left\|\mathbf{R}_m - \mathbf{W}_N\mathbf{E}\mathbf{W}_K^I\mathbf{B}\mathbf{W}_{KL}\mathbf{h}_m\right\|^2\right\}\right) \tag{30}$$

Using this approach, after some simplifications and approximations, CFO, $\varepsilon$, can be estimated as [118]:

$$\hat{\varepsilon} = \arg\min_{\varepsilon} \Psi(\varepsilon) \tag{31}$$

$$\Psi(\varepsilon) = \Re\left(F(\varepsilon)\right) \tag{32}$$

$$F(\varepsilon) = \left\{ \sum_{s=1}^{N_c-1} r'_s e^{-j2\pi\varepsilon s} \right\} \tag{33}$$

$$r'_s = \sum_{m=1}^{N_r} \sum_{p=0}^{N_c-1-s} [\mathbf{T}]_{p,p+s} r^*_{m,p+k} r_{m,p+s+k} \tag{34}$$

$$\mathbf{T} = \mathbf{S}^H \mathbf{S} \tag{35}$$

$$\mathbf{S} = \left(\mathbf{I} - (2/K)\mathbf{W}_K^I \mathbf{B} \mathbf{W}_{KL} \mathbf{W}_{KL}^H \mathbf{B}^H \mathbf{W}_K^{IH}\right) \tag{36}$$

Therefore, for frequency synchronization first we calculate matrices **S** and **T** (eq. (36) and (35) ) (which are constants and can be calculated only once) and at the receive antennas after packet arrival, we remove cyclic prefix and then by using **T**, the coefficients $r'_s$, $s = 1,...,N_c - 1$ can be calculated (eq. (34) ). Eq. (33) is similar to Discrete Fourier Transform and so **E** can be easily calculated by Fast Fourier Transform (FFT) algorithm. Better frequency resolution can be achieved by zero-padding to effectively increase the length of symbol sequence. The minimum of real part of Fourier transform indicates the estimated carrier frequency offset, $\hat{\varepsilon}$.

## 3.2.1.2 Specifications

i. This method is a frequency-time domain method i.e. at the transmitter, training sequence is inserted in the frequency domain (before IFFT operation). At the receiver frequency estimation is performed based on time domain received signal. (cyclic prefix is removed before applying this method )
ii. Time synchronization (frame synchronization + symbol timing) has to be done before this method.
iii. One OFDM word (with its cyclic prefix) is used as training sequence in the beginning of each packet, the structure of training sequence and information blocks can be seen in Figure 16.

iv. For frequency synchronization orthogonality between preambles is not supposed but for channel estimation the orthogonal preamble was used according to [62]
v. The criterion is derived by supposing SNR>0 dB [118].
vi. For frequency synchronization, at each transmitter, $K$ out of the total of $N_c$ OFDM subcarriers have to be used to transmit symbols (this is different from inserting a cyclic prefix for each OFDM word) e.g. if we have $N_c$=128 we can choose $K$=120.
vii. The fading multi-path channel is considered to be quasi static
viii. The only constraint over the length of the channel is that it must be less than cyclic prefix length.
ix. After frequency offset compensation channel can be estimated using [62].
x. This method is designed for an EM based iterative receiver.

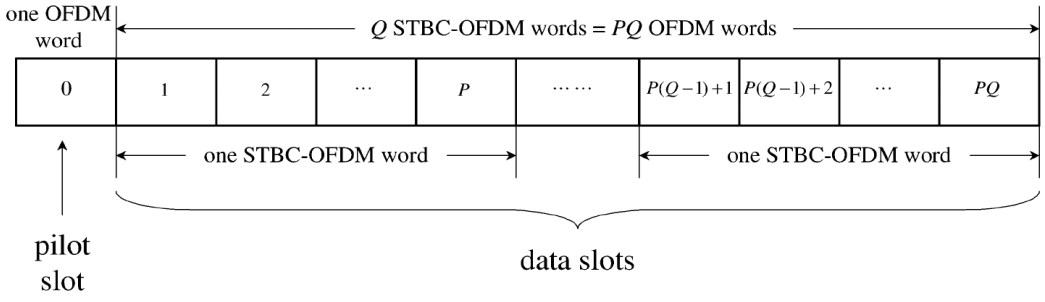


Figure 16: Data burst structure for FFT method.

## 3.2.2 Hopping pilots method

In this method appeared in [114], by inserting hopping pilots into information block, CFO can be estimated in frequency domain.

### 3.2.2.1 Description of method

Let $\mathbf{x}_{n,qN_{data}} = \left[ x_{n,qN_{data}} \quad x_{n,qN_{data}+1} \ \ldots \ x_{n,qN_{data}+N_{data}-1} \right]^T$ be the $q^{th}$ information block at $n^{th}$ transmit antennas where $x_{n,k}$ is the $k^{\text{th}}$ sample transmitted from the n$^{\text{th}}$ transmit antenna. Also, let

$\mathbf{b}_{n,qN_{data}} = \left[ b_{n,qN_p} \quad x_{n,qN_p+1} \ \ldots \ x_{n,qN_p+N_p-1} \right]^T$ the $q^{th}$ training block at $n^{th}$ transmit antennas. $N_{data}$ and $N_p$ is the number of information symbol and training symbol per block respectively. At the transmitters, we insert $N_t$ training symbol $\{\boldsymbol{b}_n(k)\}_{n=1}^{N_t}$ in the information blocks corresponding to the $N_t$ transmit antennas $\{\mathbf{x}_{n,qN_{data}}\}_{n=1}^{N_t}$, as follows:

$$\tilde{\mathbf{u}}_{n,qK} = \boldsymbol{P}_A \mathbf{x}_{n,qN_{data}} + \boldsymbol{P}_B \mathbf{b}_{n,qN_{data}} \tag{37}$$

where $\tilde{\boldsymbol{u}}_n(k)$ contains both information-bearing symbols and training symbols. $\boldsymbol{P}_A$ and $\boldsymbol{P}_B$ are two permutation matrices with sizes $N \times N_{data}$ and $K \times N_p$ respectively ($K$ is the number of non-zeros subcarriers), and are selected to be mutually orthogonal: $\boldsymbol{P}_A^T \boldsymbol{P}_B = \mathbf{0}_{N_{data} \times N_p}$. Note that $N_{data}+N_p=K$ and $K<N_c$ ($N_c$ is the number of subcarriers). One example of permutation matrices is to form $\boldsymbol{P}_A$ with the last $N_{data}$ column of $\mathbf{I}_{N_p+N_{data}}$ and $\boldsymbol{P}_B$ with the first $N_p$ column of $\mathbf{I}_{N_p+N_{data}}$, given as:

$$\boldsymbol{P}_A = \left[ \boldsymbol{e}_{N_p} \ \ldots \ \boldsymbol{e}_{K\text{-}1} \right] \quad \text{and} \quad \boldsymbol{P}_B = \left[ \boldsymbol{e}_0 \ \ldots \ \boldsymbol{e}_{N_p\text{-}1} \right] \tag{38}$$

$\boldsymbol{e}$ is a zero vector with a one corresponding to its index. Afterwards, we insert $N_c - K$ zeros per block to obtain $\mathbf{u}_{n,qN_c}$. This insertion can be implemented by left-multiplying $\tilde{\boldsymbol{u}}_n(k)$, with the null-subcarrier insertion matrix:

$$\begin{aligned} &\tilde{\mathbf{u}}_{n,qN_c} = \boldsymbol{T}_{sc}(q) \tilde{\mathbf{u}}_{n,qK} \\ &\boldsymbol{T}_{sc}(q) = [\boldsymbol{e}_{v_q (\mathrm{mod}\, N_c)}, \ \ldots \ , \boldsymbol{e}_{v_q + K\text{-}1 (\mathrm{mod}\, N_c)}] \end{aligned} \tag{39}$$

Where $v_q=q[N_c/(L+1)]$ (L is the channel length). Dependence of the null-subcarrier insertion matrix $\boldsymbol{T}_{sc}$ on the block index $q$ implies that the position of the inserted zero is changing from block to block. In other words, (39) implements a null-subcarrier hopping operation from block to block.

Subsequently we perform the standard OFDM operations of IFFT and CP insertion per transmit antenna. We perform CFO estimation at receivers as follows:

Let the $q^{th}$ received vector with length $N_c$ at $m^{th}$ receive antenna after removing cyclic prefix be $\mathbf{r}_{m,qNc}$. We collect $P$ vector namely $\{\boldsymbol{r}_{m,qN_c}\}_{q=1}^{P}$. We multiply each vector by dehopping matrix $\boldsymbol{D}_{N_c}^H(q)$:

$$\overline{\boldsymbol{r}}_{m,qN_c} = D_{N_c}^H(q) \boldsymbol{r}_{m,qN_c} \tag{40}$$

where:

$$D_{N_c}^{H}(q) = diag\left[1, e^{-j\frac{2\pi v_q}{N_c}}, ..., e^{-j\frac{2\pi v_q (N_c-1)}{N_c}}\right] \tag{41}$$

CFO destroys the orthogonality among subcarriers, and the training information is mingled with the unknown symbols and channel. The dehopping matrix makes the null subcarriers in different blocks done at the same location.

Then, we form the covariance matrix of $\overline{\boldsymbol{r}}_{m,qN_c}$ averaged over $M$ blocks (M ≥ K).

$$\hat{\boldsymbol{R}}_{\overline{r}_m} = \frac{1}{M}\sum_{q=0}^{M-1} \overline{\boldsymbol{r}}_{m,qN_c} \overline{\boldsymbol{r}}_{m,qN_c}^{H} \tag{42}$$

It has been shown that the column space of $\hat{\boldsymbol{R}}_{\overline{r}_m}$ consists of two parts, the signal subspace and the null subspace. In the absence of CFO, the null space of $\hat{\boldsymbol{R}}_{\overline{r}_m}$ is spanned by the missing columns (the location of the null subcarriers) of the FFT matrix. The presence of CFO introduces a shift in the null space. A cost function can be built to measure this CFO-induced shift for MIMO-OFDM setup. CFO can be estimated by minimizing the cost function by the following equations:

$$\hat{\varepsilon} = \arg\min_{\varepsilon} J(\varepsilon) \tag{43}$$

$$J(\varepsilon) = \sum_{s=K}^{N_c-1} \boldsymbol{f}_{N_c}^{H}\left(\frac{2\pi s}{N_c}\right) \boldsymbol{D}_{N_c}^{-1}(\varepsilon) \left\{\sum_{m=1}^{N_r} \hat{\boldsymbol{R}}_{\overline{r}_m}\right\} \boldsymbol{D}_{N_c}(\varepsilon) \boldsymbol{f}_{N_c}\left(\frac{2\pi s}{N_c}\right) \tag{44}$$

where:

$$\begin{aligned} \boldsymbol{f}_{N_c}(\omega) &= \left[1, \exp(j\omega), ..., \exp(j(N_c-1)\omega)\right]^T \\ \boldsymbol{D}_{N_c}(\varepsilon) &= diag\left[1, \exp(2\pi j\varepsilon), ..., \exp(2\pi j\varepsilon(N_c-1))\right] \end{aligned} \tag{45}$$

Complexity of the estimator in (43) depends on the number of points searched over the interval $[-\pi, \pi)$ and so as usual, there are tradeoffs emerging among complexity acquisition range and variance of the CFO estimator. The finer the search, the higher the complexity one incurs.

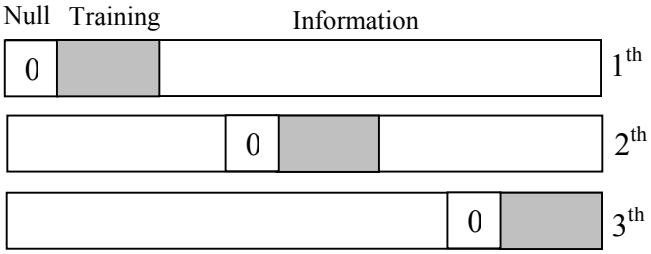


Figure 17: One example of training structure

### 3.2.2.2 Specifications

i. This method is a frequency-time domain method. That is, at the transmitter training sequence is inserted in frequency domain before IFFT operation but at the receiver frequency synchronization is performed based on time domain received signal. Cyclic prefix is removed before applying the method.

ii. Time synchronization (frame synchronization + symbol timing) has to be performed before applying this method

iii. In each OFDM word we have to insert at least $N_n$ hopping null subcarrier and to perform frequency synchronization we have to use $M \geq N_c - N_n$ words, the structure of OFDM word can be seen in Figure 17.

iv. The channels remain time-invariant during $M$ blocks.

v. This method is highly dependent on training symbols design.

vi. This method is flexible to adjust the training sequence, depending on the channel's coherence time and the pertinent burst duration; e.g., if the burst is long, we can insert fewer symbols per block.

vii. A phase estimation method is presented in [114] asserting that it leads to BER performance enhancement.

viii. Using the structure of training sequence presented in [78], the channel can be estimated as well.

## 3.2.3 Iterative method

In this method, using time domain training sequence, CFO can be estimated in two stages: in the first stage a rough estimation is calculated by using FFT and then in the second stage in an iterative manner fine frequency estimation is provided.

This algorithm is developed by us and presented for the first time in [4] and then will be published in [2]. We will describe this method in detail in Chapter 5. But here to cover all of the good CFO estimation methods we will outline the algorithm without mathematical proof.

### 3.2.3.1 Description of method

Supposing that time synchronization is already performed. Let the training sequence of $n^{\text{th}}$ transmit antenna be $[c_{n,0}\ c_{n,0}\ \ldots\ c_{n,N_P-1}]$ ($N_P$ is the length of training sequence) and the received signal at $m^{\text{th}}$ receive antenna and $k^{\text{th}}$ instant be $r_m(k)$. If the length of the cyclic prefix for the OFDM symbol is $N_{CP}$, we have $[c_{n,-N_{CP}}\ \ldots\ c_{n,-1}] \equiv [c_{n,N_P-N_{CP}}\ \ldots c_{n,N_P-1}]$ as the cyclic prefix of training sequence.

In the first step, CFO is estimated by FFT method. To this purpose we define matrix $\mathbf{\Phi}$ as follows:

$$\mathbf{\Phi} = \mathbf{C}\ (\mathbf{C}^H\mathbf{C}\ )^{-1}\mathbf{C}^H \tag{46}$$

where:

$$\mathbf{C}\ = \begin{bmatrix} C_1 & C_2 & \ldots & C_{N_t} \end{bmatrix}_{N_p \times LN_t} \tag{47}$$

$$C_n = \begin{bmatrix} c_{n,0} & c_{n,-1} & \ldots & c_{n,-L+1} \\ c_{n,1} & c_{n,0} & \ldots & c_{n,-L+2,} \\ . & . & . & . \\ c_{n,N_p-1} & c_{n,N_P-2} & \ldots & c_{n,-L+N_P} \end{bmatrix}_{N_p \times L} \tag{48}$$

Now, CFO, $\varepsilon$, is the maximum of real part of Fast Fourier Transformation (FFT) of $\left[r'_s\right]_{s=1}^{N_p-1}$; That is:

$$\hat{\varepsilon} = \arg\max_{\varepsilon}\left(\Re e\left(\left\{\sum_{s=1}^{N_p-1} r'_s e^{-j2\pi\varepsilon s}\right\}\right)\right) = \arg\max_{\varepsilon}\left(\Re e\left(\mathrm{FFT}\left[r'_s\right]_{s=1}^{N_p-1}\right)\right) \tag{49}$$

where:

$$r'_s = \sum_{m=1}^{N_r}\sum_{p=0}^{N_p-1-s}[\phi]_{p,p+s} r^*_{m,p+k} r_{m,p+s+k} \tag{50}$$

and $\Re e(.)$ denotes the real part of (.).

Better frequency resolution can be achieved by zero-padding to effectively increase the FFT length but for initial estimation, a 512-FFT point FFT would be sufficient. After frequency compensation, channel can be estimated as follows:

$$\hat{\mathbf{h}} = [(\mathbf{I}_{N_r} \otimes \mathbf{C}\ )^H (\mathbf{I}_{N_r} \otimes \mathbf{C}\ )]^{-1} (\mathbf{I}_{N_r} \otimes \mathbf{C})^H \mathbf{r} \tag{51}$$

where $\mathbf{r} = \begin{bmatrix}\mathbf{r}_1^T & \mathbf{r}_2^T & \dots & \mathbf{r}_{N_r}^T\end{bmatrix}^T_{N_p N_r \times 1}$ and $\mathbf{r}_m = \begin{bmatrix} r_m(0) & r_m(1) & \dots & r_m(N_p-1)\end{bmatrix}^T_{N_p \times 1}$.

In the next step we are going to estimate the residual frequency in an iterative manner. To this purpose we use the channel estimated in (51) and we re-estimate $\varepsilon$ by using a cross correlation method: we define $r''(s) = \sum_{p=0}^{N-1-s} y(p) y^*(p+s)$ and $y(s) = \sum_{m=1}^{N_r} \hat{\mathbf{h}}_{:m}^H \mathbf{C}_0^H(s,:)\mathbf{r}_{m,0}(s)$ where $\mathbf{C}^H(s,:)$ is $s^{th}$ row of matrix $\mathbf{C}$ (eq. (48)) and $\mathbf{h}_{:m} = \begin{bmatrix}\mathbf{h}_{1m}^T & \mathbf{h}_{2m}^T & \dots & \mathbf{h}_{N_t m}^T\end{bmatrix}^T_{LN_t \times 1}$, $\mathbf{h}_{nm} = \begin{bmatrix} h_{nm}(0) & h_{nm}(1) & \dots & h_{nm}(L-1)\end{bmatrix}^T_{L\times 1}$. $\varepsilon$ can be calculated as follows:

$$\hat{\varepsilon} = \frac{1}{2\pi}\frac{\sum_{s=1}^{N_p-1} s\left|r''(s)\right|\arg(r''(s))}{\sum_{s=1}^{N_p-1} s^2\left|r''(s)\right|} \tag{52}$$

We have to note that because of presence of the arg(.) in (52), $\hat{\varepsilon}$ is ambiguous when $|\arg(r''(s))|$ exceed $\pi$. This limits $\hat{\varepsilon}$ to be less than $1/2N_p$.

Now algorithm can be expressed as follows: in the first stage we estimate arrival packet instant, the channel coefficients, and CFO (as describe in eq. (49)), after compensating CFO, the channel can estimated (51) and then residual CFO can be estimated by eq. (52). Estimating residual CFO,

the channel can be re-estimated by (51). Better channel estimation leads to better CFO estimation and so we re-estimate CFO by eq. (52). This procedure is performed for several iterations.

#### 3.2.3.2 Specifications

i. This method is time domain i.e. both at the transmitter and at the receiver we deal with the time domain signals.
ii. Time synchronization (frame synchronization) can be incorporated in this method.
iii. One preamble is used as training sequence in the beginning of each packet.
iv. For frequency synchronization (and incorporated time synchronization) orthogonality between preambles is not supposed.
v. The fading multi-path channel is considered to be quasi static and the length of the channel must be less than cyclic prefix length.
vi. This method incorporates an estimation of channel coefficients.
vii. This method presents a very accurate and rather complicated CFO estimator

### 3.2.4 Autocorrelation method

In this method presented in [111] CFO can be estimated by autocorrelation of two same consecutive preambles.

#### 3.2.4.1 Description of method

Let the received signal at $m^{th}$ receive antenna and $k^{th}$ instant be $r_m(k)$. To estimate CFO, we calculate the correlation function over received signal and delayed version of it. This operation should be done in the maximum of the correlation function. The correlation function $\Lambda$ is defined as:

$$\Lambda(k_0) = \sum_{m=1}^{N_r} \sum_{k=k_0}^{m+N_p-1} r_m(k) r_m^*(k-N_p) \qquad (53)$$

The estimated frequency offset $\hat{\varepsilon}$ is the phase of autocorrelation function in its maximum, it can be given by:

$$\hat{\varepsilon} = \frac{\theta_{k_{\max}}}{2\pi N_P T_s} = \frac{f_s \angle \Lambda(k_{\max})}{2\pi N_P}$$

where $\theta_k = \angle\Lambda(k)$ indicates the phase of $\Lambda(k)$ and $\Lambda(k)$ reaches to its maximum at $k_{\max}$, $f_s = 1/T_s$ is sampling frequency.

In this method the preambles on the different TX antennas have to be orthogonal and shift-orthogonal for at least the channel length. To meet these requirements reference [111] proposes the use of constant envelope orthogonal codes with good periodic correlation properties, such as Frank-Zadoff codes [112] and reference [47] proposes the time orthogonal codes as shown in Figure 18 (see 4.3.1).

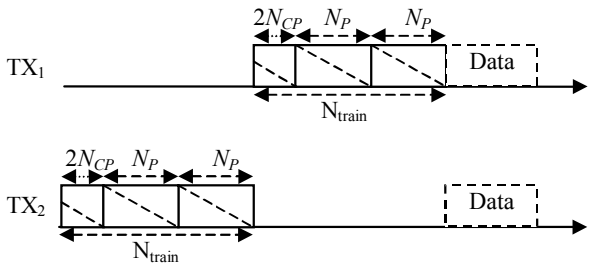


Figure 18: Two identical consecutive preambles

## 3.2.4.2 Specifications

i. This method is a time domain method i.e. both at the transmitter and at the receiver we deal with the time domain signals.
ii. Time synchronization (frame synchronization) can be incorporated in this method easily.
iii. Two same consecutive preambles are used as training sequence in the beginning of each packet. (Figure 18)
iv. For frequency synchronization (and incorporated time synchronization) different preambles have to be orthogonal and each preamble has to be shift-orthogonal and so the challenge in this method is to find appropriate preambles.
v. This method is performed before removing cyclic prefix and this methods works well even if the time synchronization has an error within the cyclic prefix.
vi. The fading multi-path channel is considered to be quasi static and the length of the channel must be less than cyclic prefix length.
vii. This method reduces the spectral efficiency.
viii. This method provides a very simple CFO estimator with a rather good accuracy.

| METHOD | | Number of complex multiplication | A typical order |
|---|---|---|---|
| FFT | | $(N^2-N)N_r+2R\log R$ | $10^4$ |
| Hopping pilots | | $N_r(N+N^2)+N^3+N^2+2N^2R$ | $10^7$ |
| Iterative: | Initial estimation | $(N^2-3N+2)N_r+2R\log R$ | $10^4$ |
| | Each iteration | $\frac{(N^2+N)}{2}+NN_rN_tL$ | $10^3$ |
| Autocorrelation | | $NN_r$ | $10^2$ |

Table 1: An approximation of required complex multiplication to implement the methods

# 3.3 Comparison and conclusion

Table 1 represents an approximation of required complex multiplication to implement the presented methods. In this table $R$ stands for selected resolution in the algorithm, e.g in FFT method with a 512 FFT length, $R$ is 512, and $N$ stands for preamble length for example a typical order is 64. $N_r$ and $N_t$ are typically equal 2.

According to Table 1 the simplest method is autocorrelation method and the most complicated is hopping pilots method. Table 2 represents a brief comparison between different methods. In this table we see a column dedicated to accuracy of these methods. The accuracy of these methods will be illustrated in chapter 5 section 5.4.3 by numeric simulation. FFT method, as stated in the table, has a good spectral efficiency with a rather simple implementation, it can be incorporated in an EM receiver (as described in [118]) but it has a rather poor accuracy. Hopping pilots method is a rather complicated method and with increasing its complexity it can be more accurate. This method would be suitable if one perform synchronization with inserting the pilots. Iterative method has a good initial estimation and by several iterations its MSE can be reach to Cramer-Rao lower bound (it will be calculated in 5.4.1). It can be also incorporated into time synchronization algorithm (see 5chapter ). However, the disadvantage of this algorithm is its rather high complexity. On the other hand, autocorrelation method is a very simple method and

its MSE is near to Cramer Rao lower bound. It is rather robust against timing error and can be incorporated in time synchronization method. The disadvantage is that we have to force the preambles to be consecutive and also orthogonal. If we use time orthogonal code spectral efficiency is poor otherwise we have to generate orthogonal and shift orthogonal preambles.

Altogether, it seems the best methods for MIMO frequency synchronization is either autocorrelation or iterative method but our final decision depends on the selected time synchronization method and performance of the complete synchronization system.

Table 2: Comparison between different methods

| METHODS | COMPLEXITY | ACCURACY* | SPECTRAL EFFICACY | DOMAIN | TIME SYNCHRONIZATION | ORTHOGONALITY |
|---|---|---|---|---|---|---|
| ***FFT method*** by using frequency domain training sequence | Intermediate | Rather Poor | Rather good | Frequency-Time | Time synchronization must already be done | Not required |
| ***Hopping pilots*** method | High | Good<br>more complexity more accuracy | Rather good<br>inserted pilots | Frequency-Time | Time synchronization must already be done | Not required |
| ***Iterative method*** | Intermediate | Excellent | Good | Time-Time | Can be joint with time synchronization | Not required |
| ***Autocorrelation*** method | Low | Very good | Rather poor | Time-Time | Can be joint with time synchronization | Required |

* The simulations are described in chapter 5 section 5.4.3

*Chapter*


# TIME SYNCHRONIZATION

## 4.1 Introduction

The problem of time synchronization in MIMO systems is divided into two parts:

- Frame synchronization: The task of the frame synchronization is to identify the preamble in order to detect a packet arrival
- Symbol-timing: The task of symbol timing is to identify the beginning of the symbol. Symbol timing in an OFDM system is to find the exact place of FFT window or, in other word, the beginning of the OFDM symbol.

The task of frame synchronization for both MIMO and MIMO-OFDM systems is same and so the same algorithms are used for both MIMO and MIMO-OFDM systems. In contrary, the

problem of symbol timing is different for MIMO and MIMO-OFDM systems. This difference is due to different definition of the symbol in MIMO and MIMO-OFDM systems. In fact, an OFDM symbol is composed of several sub-carriers and thanks to Fourier transformation the time offset smaller than sampling time can be absorbed as a phase offset in the channel coefficients whereas in a non-OFDM MIMO system the major task of a symbol timing algorithm is to find the time offset smaller than sampling time. In other words, if we suppose that the received signal is sampled at $t=kT_s+\eta_0 T_s$, where $T_s$ is the sampling rate, and $\eta_0 \in [0,1)$ is the unknown time offset induced by the combination of channel delay and the sampling phase offset, in OFDM systems $\eta_0$ after Fourier transformation will be appeared a channel phase and thus, contrary to non-OFDM systems, we do not need to estimate $\eta_0$. On the other hand, MIMO symbol-timing synchronization for non OFDM is an important issue in space-time coding systems because perfect symbol timing at the receiver is usually assumed in literature.

The goal of this chapter is to present the different approaches to MIMO-OFDM time synchronization. Since MIMO-OFDM time synchronization approaches are quite influenced by non MIMO−OFDM time synchronization methods, we will first look at some well-known non MIMO−OFDM time synchronization methods that are designed either for MIMO non-OFDM or for SISO OFDM systems. After this rather long introductory section on non MIMO−OFDM approaches, the different methods for MIMO-OFDM time synchronization will be discussed

## 4.2 non MIMO-OFDM time synchronization

### 4.2.1 MIMO non-OFDM time synchronization

The same methods that are generally used for MIMO frame synchronization are employed for MIMO-OFDM frame synchronization too. Therefore, the difference between MIMO time synchronization and MIMO-OFDM time synchronisation is due to different methods used for symbol timing. Hence, here we take a look at the problem of symbol timing in MIMO and not MIMO-OFDM systems. However, these ideas can be used also for MIMO-OFDM time synchronization.

Apparently, the synchronization of MIMO systems might seem closely related to the symbol timing estimation in single-input-single-output (SISO) systems. In MIMO systems, the synchronization problem takes a special form because signals from different antennas are superimposed together, and the symbol timing estimation algorithms commonly proposed for SISO systems do not work well in MIMO systems. Furthermore, in MIMO systems, training sequences must be used to estimate simultaneously a number of channels. This opens up two questions. How can we make use of the training sequences to perform symbol timing estimation? What kind of training sequences is beneficial to symbol timing estimation?

The problems that we have to handle in symbol timing are as follows:

How to estimate the timing delay in MIMO system for the case i) when the transmitted data is assumed to be known, ii) and when it is unknown? iii) How to design optimal training sequences in data-aided case in order to get the best estimation performances?

In this section, we divide MIMO symbol timing problem into two parts: the blind or non data aided approaches and the data aided ones

Symbol timing synchronization in MIMO uncorrelated flat fading channel was first studied by Naguib [50]; where the timing delay is estimated by selecting the sample with maximum amplitude from the oversampled approximated log-likelihood function. This algorithm was extended in [51] to increase its estimation accuracy and the authors again generalize the algorithm in [54]. On the other hand, because of the problem of identifying training sequence

epoch, non data aided methods are also interesting. For the first time a good algorithm for non data aided synchronization was presented by Oerder and Meyr [52]. The well-known squaring algorithm by Oerder and Meyr for non data aided symbol timing estimation in SISO channels was extended in MIMO channels in [53] and [54], resulting in a non data aided MIMO estimator. However the estimator proposed in these references as well as [52] suffers from the problem of self noise, which is inherited from the original squaring algorithm.

#### 4.2.1.1 Data aided Symbol timing

Both ST block coding (STBC) and ST trellis coding (STTC) systems can be described by the same basic communication model [50]. The simplified baseband equivalent model with $N_t$ transmit and $N_r$ receive antennas is shown in Figure 19.

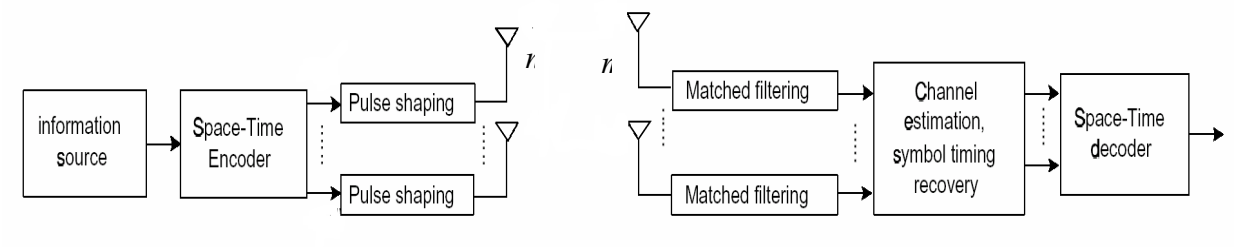


Figure 19: Simplified baseband model for space-time coding system

Let $c_{n,i}$ ($i$=0,…,$N_p$-1) (n=1,…,$N_t$) be the $i^{th}$ orthogonal training sequence of length $N_p$. Let the received signal be sampled $Q$ times faster than the symbol rate $1/T_s$. Each received sample can be indexed by the $l^{th}$ training bit and $k^{th}$ phase such as $s = lQ + k$ . The $s^{th}$ symbol of the received signal $r_{m,s}$ in a flat fading channel according to [50] can be written as:

$$\begin{aligned} r_{m,s} &= \sqrt{\frac{E_S}{N_t}} \sum_{n=1}^{N_t} h_{nm} \sum_i c_{n,i}\, p(sT_s / Q - iT_s - \eta T_s) + w_{m,s} \\ &= \sqrt{\frac{E_S}{N_t}} \sum_{n=1}^{N_t} h_{nm} \sum_i c_{n,i}\, p(kT_s / Q + (l-i)T_s - \eta' T_s) + w_{m,(lQ+k)} \\ &\textit{for} \quad l = 0,1,...,N_p - 1 \quad \textit{and} \quad k = 0,1,...,Q-1 \end{aligned} \tag{54}$$

Where $p(t)$ is the result of convolution between transmit and receive filter. $T_s$ is the symbol duration, $\eta'$ denotes the distance between the first sample of current symbol and optimal instant while $\eta$ represents the distance between center sample and optimal instant. $w_{m,s}$ denotes

complex circularly distributed Gaussian white noise at the $m^{\text{th}}$ receive antenna and the $s^{\text{th}}$ instant. Grouping the samples with the same phase, one can form the vector $\mathbf{r}_{m,k}$ as follows:

$$\mathbf{r}_{m,k} = \begin{bmatrix} r_{m,k} & r_{m,(Q+k)} & \cdots & r_{m,((N_P-1)Q+k)} \end{bmatrix}^T \tag{55}$$

And we can rewrite (54) in matrix form as follows:

$$\mathbf{r}_{m,k} = \sqrt{\frac{E_s}{N_t}} \sum_{n=1}^{N_t} h_{nm} \mathbf{C}_n \mathbf{p}(k) + \mathbf{w}_{m,k} \tag{56}$$

Where:

$$\mathbf{C}_{n,0} = \begin{bmatrix} c_{n,(\mathrm{mod}(-N_{CP},N_p))} & c_{n,(\mathrm{mod}(-N_{CP}+1,N_p))} & \cdots & c_{n,(\mathrm{mod}(N_{CP},N_p))} \\ c_{n,(\mathrm{mod}(-N_{CP}+1,N_p))} & c_{n,(\mathrm{mod}(-N_{CP}+2,N_p))} & \cdots & c_{n,(\mathrm{mod}(N_{CP}+1,N_p))} \\ \vdots & \vdots & & \vdots \\ c_{n,(\mathrm{mod}(-N_{CP}+N_p-1,N_p))} & c_{n,(\mathrm{mod}(-N_{CP}+N_p,N_p))} & \cdots & c_{n,(\mathrm{mod}(N_{CP}+N_p-1,N_p))} \end{bmatrix} \tag{57}$$

$$\mathbf{p}(k) = \begin{bmatrix} p(kt/Q - N_{CP}T - \eta' T_s) & p(kt/Q - (N_{CP}-1)T_s - \eta' T_s) & \mathrm{L} & p(kt/Q + N_{CP}T_s - \eta' T_s) \end{bmatrix} \tag{58}$$

$$\mathbf{w}_{m,k} = \begin{bmatrix} w_{m,k} & w_{m,(Q+k)} & \cdots & w_{m,((L_t-1)Q+k)} \end{bmatrix}^T \tag{59}$$

In these equations $N_{CP}$ the size of prefix and suffix (here we use a cyclic prefix and cyclic suffix, each of length $N_{CP}$ [51]).

We multiply the received signal by training sequence to form $\Psi_{nm}(k) = \mathbf{C}_n{}^H \mathbf{r}_n(k)$. Supposing the relative delay is zero and $\mathbf{C}_n$ 's of different antennas are orthogonal to each other we have:

$$\Psi_{nm}(k) = \sqrt{\frac{E_s}{N_t}} h_{nm} p(kT_s / Q - \eta' T_s) \left\| \mathbf{C}_{n,0} \right\|^2 + \sqrt{\frac{E_s}{N_t}} \sum_{n=1}^{N_t} h_{nm} \mathbf{C}_n^H \tilde{\mathbf{C}}_n \tilde{\mathbf{p}}(k) + \mathbf{C}_n^H \mathrm{w}_{m,k} \tag{60}$$

$\tilde{\mathbf{C}}_n$ is the same as $\mathbf{C}_n$ but with $(N_{CP}+1)^{\text{th}}$ column removed and $\tilde{\mathbf{p}}$ is the same as $\mathbf{p}$ but with the $(N_{CP}+1)^{\text{th}}$ entries removed. The second term in (60) represents the ISI which reduce to zero if the training sequences are orthogonal and the relative delay is zero. The last term in (60) is the noise term. From (60), it can be observed that, if the second and the third terms are very small, $\Psi_{nm}(k)$ has the same shape as $p(t)$ except that it is scaled by a complex channel gain. In order to remove the effect of the channel, consider the sequence $\Lambda_{nm}(k) = \left| \Psi_{nm}(k) \right|^2$ which has a same shape as $|p(t)|^2$. Now the optimum sampling phase $k=k_0$ is selected such that it maximizes $\Lambda_{nm}(k)$.

$$k_o = \max_{k=0,1,\ldots,Q-1} \Lambda_{ML}(k) \tag{61}$$

$$\Lambda_{ML}(k)=\sum_{m=1}^{N_r}\sum_{n=1}^{N_t}\Lambda_{nm}(k)$$

Under optimistic assumption that the sample closest to the optimum sampling position is correctly estimated (at high SNR) the estimation error, normalized with respect to the symbol duration, is a uniformly distributed random variable in the rang [-1/2$Q$, 1/2$Q$]. Therefore, the MSE normalized with respect to the symbol duration $T_s$ is (1/12$Q^2$). As a result, the performance of this timing synchronization highly depends on the oversampling ratio. In fact, relatively high oversampling ratio might be required for accurate symbol timing estimation. Wu [51] has proposed an interpolation algorithm based on Fourier series in order to estimate symbol timing. The advantage of this algorithm is that accurate timing estimates can be obtained even if the oversampling ratio is small. Both analytical and simulation results in [51] show that for modest oversampling ratio (such as $Q$=4), the MSE of the proposed estimator is significantly smaller than that of the optimum sample selection algorithm.

In order to explain Wu's algorithm let's construct a periodic sequence $\tilde{\Lambda}_{nm}(s)$ by periodically extending the approximation log-likelihood sequence $\Lambda_{nm}(k)$ in (61). Further, denote $\tilde{\Lambda}_{nm}(\bar{\eta}')$ as the continuous and periodic approximated log likelihood function with its samples given by $\tilde{\Lambda}_{nm}(s)$. According to sampling theorem, as long as the sampling frequency $Q/T_s$ is higher than twice the highest frequency of $\tilde{\Lambda}_{nm}(\bar{\eta}')$, then $\tilde{\Lambda}_{nm}(\bar{\eta}')$ can be represented by its samples $\tilde{\Lambda}_{nm}(s)$ without loss of information. The relationship between $\tilde{\Lambda}_{nm}(s)$ and $\tilde{\Lambda}_{nm}(\bar{\eta}')$ is then given by:

$$\tilde{\Lambda}_{ML}(\bar{\eta}')=\sum_{s=-\infty}^{\infty}\tilde{\Lambda}_{ML}(s)\,\mathrm{sinc}\left(\pi\frac{\bar{\eta}'T_s-sT_s/Q}{T_s/Q}\right) \qquad (62)$$

Then we can expand $\tilde{\Lambda}_{nm}(\bar{\eta}')$ into a Fourier series and calculate the coefficient of series which are necessarily factors of $\Lambda_{nm}(k)$. By determining these coefficients, the timing delay $\bar{\eta}'$ can be estimated by maximizing $\tilde{\Lambda}_{nm}(\bar{\eta}')$. For efficient implementation, $\tilde{\Lambda}_{nm}(\bar{\eta}')$ can be approximated by an $K$-point sequence, by zero padding the high frequency of Fourier coefficients and performing a $K$-point inverse Discrete Fourier Transform (IFFT) ( for large value of $K$ *i.e.* $K>>Q$).

To avoid the complexity in performing the $K$-point IFFT, we can approximate Fourier series only by first three terms:

$$\tilde{\Lambda}_{nm}(\bar{\eta}') \approx A_0 + 2\mathrm{Re}\left\{A_1 e^{j\,2\pi\bar{\eta}'}\right\} \qquad 0 \le \bar{\eta}' \le 1 \tag{63}$$

Now our problem reduces to calculate the $\bar{\eta}'$ which maximizes $\tilde{\Lambda}_{nm}(\bar{\eta}')$. It ends to the following estimation:

$$\bar{\eta}' = -\frac{1}{2\pi}\arg\left\{\sum_{k=0}^{Q-1}\Lambda_{ML}(k)e^{-j\,2\pi k/Q}\right\} \tag{64}$$

It has been found that an oversampling factor $Q$ of 4 is sufficient to yield good estimates in practical applications. Therefore, the previous terms in (64) can be computed easily without any multiplication since $e^{-j2\pi k/4} \in \{\pm 1, \pm j\}$

Wu in [51] calculates variance of his estimator analytically and he justifies the validity of the approximation that he had used in his development.

Wu in [54] presents a ML data aided estimator by considering correlation between antennas and claims that approximated ML algorithms in the estimators in [51] and [50] are just a special case of this estimator. He drives two performance bound, the first one is the conditional Cramer-Rao bound (CCRB) (see the references of [54]) which is the Cramer-Rao bound (CRB) for the symbol timing estimation conditioned that the nuisance parameters are treated as deterministic and are jointly estimated together with unknown symbol timing. Therefore the CCRB serves as a performance lower bound for the ML estimator. The second one is the modified CRB (MRCB) (see the references of [54]), which is a lower bound for any unbiased symbol timing estimator, irrespective of the underlying assumption about the nuisance parameters. Being easier to evaluate than CRB, MCRB serves as the ultimate estimation accuracy that may be achieved. Wu shows that the MSE of the derived Data Aided (DA) ML estimator is close to the CCRB and MCRB. It means that the DA ML estimator is almost the best estimator (in term of MSE performance) for the symbol timing. Also he proves that correlation between antennas has little effect on the MSEs of DA ML estimators unless the correlation between adjacent antennas is larger than 0.5, in which case small degradations occur.

About the effect of number of transmit and receive antennas he showed that increasing of receive antenna leads to considerable MSE improvements (almost inversely proportional) but in contrary different numbers of transmit antennas result in the same estimation accuracy and therefore the MSE are approximately independent of the number of transmitters. It is tempted

to argue that using more transmit antennas, should also improve the performances of symbol timing estimation since from the experience of Alamouti and Tarokh, more transmit antennas also provide diversity gain. However, notice that the diversity gain of STBC does not come automatically by just increasing the number of transmit antennas. In STBC, the observation length for demodulating a symbol has to be increased with the number of transmit antennas. For symbol timing estimation, irrespective of the number of transmit antennas, the total transmit power and the observation length are kept constant so it is not unreasonable to have MSE performance independent of the number of transmitters. For multiple receive antennas, although the observation length is kept constant, the observation from different receive antennas are independent (similar to the situation of maximum-ratio receive combining scheme). These independent observations increase the effective length and performance is improved due to longer effective observation.

Here, Wu's algorithm in [54] will be briefly presented as the most general algorithm in the literature for flat fading channels.

With the same notation which is used in equation (54) we have:

$$r_{m,s} = \sqrt{\frac{E_S}{N_t}} \sum_{n=1}^{N_t} h_{nm} \sum_i c_{n,i} g(st/Q - iT_s - \eta T_s) + w_{m,s} \tag{65}$$

*g(t)* is the transmit filter with unit energy. After passing through the anti-aliasing filter the receive signal is then sampled $Q$ times faster than the symbol rate $1/T_s$. Note that the oversampling factor $Q$ is determined by the frequency span of $g(t)$; if $g(t)$ is band-limited to $f=\pm 1/T$ (e.g. root raised cosine (RRC) pulse) then $Q=2$ is sufficient. The received vector $\mathbf{r}_m$ which consists of $NQ$ consecutive received symbol ($N$ is observation length) from the $m^{\text{th}}$ receive antenna, can be expressed as (without loss of generality, we consider that the received sequence starts at $t=0$)

$$\mathbf{r}_m = \xi \mathbf{A}_\eta \mathbf{Z} \mathbf{H}_{:m} + \mathbf{w}_m \tag{66}$$

Where $\xi = \sqrt{\frac{E_S}{N_t}}$ and

$$\mathbf{r}_m = \begin{bmatrix} r_{m,0} & r_{m,T_s} & \cdots & r_{m,(NQ-1)T_s} \end{bmatrix}^T \tag{67}$$

$$\mathbf{A}_\eta = \left[ \mathbf{a}_{-N_g}(\eta) \;\; \mathbf{a}_{-N_g+1}(\eta) \;\; \ldots \;\; \mathbf{a}_{N_g+N-1}(\eta) \right] \tag{68}$$

$$\mathbf{a}_i(\eta) = \left[ g(-iT_s - \eta T_s) g(T_s - iT_s - \eta T_s) \ldots g((NQ-1)T_s - iT_s - \eta T_s) \right]^T \tag{69}$$

$$\mathbf{Z} = \left[ \mathbf{d}_1 \mathbf{d}_2 \ldots \mathbf{d}_{N_t} \right] \tag{70}$$

$$\mathbf{d}_n = \left[ c_{n,-N_g} \;\; c_{n,-N_g+1} \ldots c_{n,N_g+N-1} \right]^T \tag{71}$$

$$\mathbf{w}_{m,k} = \left[ w_{m,0} \quad w_{m,1} \quad \cdots \quad w_{m,NQ-1} \right]^T \tag{72}$$

$$\mathbf{H} = \begin{bmatrix} h_{11} & h_{12} & \cdots & h_{1N_r} \\ h_{21} & h_{22} & \cdots & h_{2N_r} \\ \vdots & \vdots & & \vdots \\ h_{N_t 1} & h_{N_t 2} & \cdots & h_{N_t N_r} \end{bmatrix} \tag{73}$$

With $w_{m,i} = w_{m,iT/Q}$ and $N_g$ denotes the number of the symbols affected by inter symbol interference introduced by one side of $g(t)$. Stacking the received vectors from all the receive antennas gives:

$$\mathbf{r} = \xi(\mathbf{I}_{\mathbf{N_r}} \otimes \mathbf{A}_\eta) vec(\mathbf{ZH}) + \mathbf{w} \tag{74}$$

where $\mathbf{r} = \left[ \mathbf{r}_1^T \mathbf{r}_2^T \ldots \mathbf{r}_{N_r}^T \right]^T$ and $\mathbf{w} = \left[ \mathbf{w}_1^T \mathbf{w}_2^T \ldots \mathbf{w}_{N_r}^T \right]^T$.

In order to include correlation between channel coefficients, the channel transfer function is expressed as:

$$\mathbf{H} = \sqrt{\Phi_R} \mathbf{H}_{i.d.d} \sqrt{\Phi_T}^T \tag{75}$$

Where $\Phi_R$ and $\Phi_T$ are the power correlation matrices [55] of receive and transmit antenna arrays respectively (which are assumed known) $\mathbf{H}_{i.d.d}$ contains independently and identically zero mean, unit variance, circular symmetric complex Gaussian entries. This model is based on the assumption that only immediate surroundings of the antenna array impose the correlation between antenna array elements and have no impact on the correlation at the other end of the

communication link. The validity of this model for narrow-band non-line-of-sight MIMO channels is verified by measurements (see [7–10] within [54]).

When matrix **Z** contains known training sequence the only unknown is $H_{idd}$. By resorting (74) and using well-known properties of the Kronecker product we have:

$$\mathbf{r} = \ddot{\mathbf{A}}_{\eta}\mathbf{h} + \boldsymbol{\eta} \tag{76}$$

Where $\ddot{\mathbf{A}}_{\eta} = \xi(\sqrt{\Phi_R} \otimes \mathbf{A}_{\eta}\mathbf{Z}\sqrt{\Phi_T})$ and **h** is vec($\mathbf{H}_{\text{iid}}$). Now for ML estimation of $\eta$ and **h** we have to maximize:

$$p(\mathbf{r}|\eta, h) = \frac{1}{\left(\pi\sigma_w^{\,2}\right)^{NQ}} \exp\left(-\frac{\left\|\mathbf{r} - \ddot{\mathbf{A}}_{\eta'}\mathbf{h}\right\|^2}{\sigma_w^{\,2}}\right) \tag{77}$$

Where $\eta'$ is trial values for $\eta$. In [54] it is shown that to maximize the above probability we have to maximize the following metrics:

$$\Lambda_{DA}(\eta') = \sum_{m=1}^{N_r} \mathbf{r}_m{}^{H} \mathbf{A}_{\eta'}\mathbf{Z}\left(\mathbf{Z}^{\mathbf{H}}\mathbf{A}_{\eta'}{}^{\mathbf{H}}\mathbf{A}_{\eta'}\mathbf{Z}\right)^{-1}\mathbf{Z}^{\mathbf{H}}\mathbf{A}_{\eta'}{}^{\mathbf{H}}\mathbf{r}_m \tag{78}$$

And ML data aided symbol training estimator can be written as:

$$\hat{\eta} = \arg\max_{\eta'} \Lambda_{DA}(\eta') \tag{79}$$

The maximization of the likelihood function usually involves two-step approach. The first step (coarse search) computes $\Lambda_{DA}(\eta')$ over a grid of timing delay $\eta_k = k/K$ $k=0,\ldots,K-1$, and then $\eta_k$ that maximize $\Lambda_{DA}(\eta')$ is selected in the next step (fine search) finds a global maximum by using either gradient method or interpolation or dichotomous. By using parabolic interpolation we can reach to following term:

$$\hat{\eta} = \eta_{\bar{k}} + \frac{I_1 - I_3}{2K(I_1 + I_3 - 2I_2)} \tag{80}$$

Where $I_1 = \Lambda_{DA}(\eta_{\bar{k}-1})$, $I_2 = \Lambda_{DA}(\eta_{\bar{k}})$, $I_3 = \Lambda_{DA}(\eta_{\bar{k}+1})$ and $\eta_{\bar{k}}$ is the value that maximizes $\Lambda_{DA}$ in the coarse search. The likelihood function at each receive antenna can be calculated independently and then added together to obtain the overall likelihood function.

The correlation in the transmit and receive antenna arrays does not appear in the estimator that is, the $ML_{DA}$ symbol training estimator is independent of the antenna correlations.

For large observation interval *N*, if we suppose *g*(*t*) being a Root-Raised-Cosine (RRC) pulse and the training sequence from different antennas being orthogonal we can reduce (78) to [50]

and [51] algorithms. But the sufficient amount of $N$ for this reason depends on the signal to noise ratio. Wu claims that in SNR< 20 for $N$=32, two estimators have similar performance.

Anyway, it can be seen that Wu's estimator in [54] is a very good estimator however it has been designed for flat fading channels.

In the next sub-section we take a look at preamble design discussion in symbol timing based on [54] and [56].

### 4.2.1.2 Training sequence design

In [51] the problem of how to design the training sequence for symbol timing estimation has addressed. It is said that in order to minimize MSE in the symbol timing we have to use orthogonal training sequence in order to minimize the ISI and noise term in (60). So a design procedure for designing orthogonal preambles in MIMO system was presented by using [74]. In fact, in [74] one can find some methods to construct several orthogonal sequences from one shift orthogonal sequence. (See also the references within [74])

Here we mention the method of constructing perfect sequences which has been used in [51]

1) Construct a sequence $\mathbf{s}=[s(0), s(1),\ldots, s(N_p)]$ ($N_P$ is training sequence length) such that all of its out-of-phase periodic auto-correction terms are equal to zero.

2) Construct another sequence $\mathbf{s'}$ ($N_{CP}$ is cyclic prefix length) as follows:

$\mathbf{s'}=[s(0), s(1),\ldots, s(N_P-1), s(0), s(1)\ldots s(2N_tN_{CP}-1)]$

Note that $N_p>2N_tN_{CP}$ must be satisfied. That is, if the number of transmit antenna is large, we cannot use training sequences with short length.

3) The orthogonal training sequences are given by :

$$\mathbf{c}_i=[s'((2i-1)\,N_{CP}), \ldots, s'((2i-1)\,N_{CP}+N_p-1)]$$

But if approximated log-likelihood function is not used in the estimation (e.g. the true ML estimator [54], these type of training sequence may not be optimal anymore so another approach has been in [56] for training sequence design. In [56] it has been investigated that how one can derive optimal training sequences independent of the estimation methods. Toward this

end, the orthogonal training sequences have been found by minimizing the modified Cramer-Rao bound (MCRB) with respect to the training data; in this paper it has been shown that when the transmit pulse is root raised cosine and there is no correlation among antennas, the optimal orthogonal training sequence resemble the Walsh sequences. Furthermore it has been also shown that the knowledge of antenna correlation is not important for designing training sequence. Figure 20 compares the performance of $ML_{DA}$ with different kinds of training sequence in a 4 × 4 receive antenna system with $N$=32, $N_g$=4, g(t) being a RRC pulse with α=.3. Three different kinds of training sequences are considered. The first one is the optimal training sequence derived in [56]. The second one is the Walsh sequences $w_{31}$, $w_{30}$, $w_{29}$, $w_{28}$ and extended to length 40 by adding a cyclic prefix and suffix each of length equal 4. The final one is the perfect sequence as we described above. From Figure 20, it can be seen that the perfect sequences perform not as well as Walsh sequences and the optimal sequences. This is because the true ML estimator is used in simulation and perfect sequences (which were derived based on the approximated log-likelihood function) may not have any optimality. Due to the resemblance of the optimal orthogonal sequences and the Walsh sequences, the performance of the $ML_{DA}$ by using these two kinds of sequence are close to each other, with the case of optimal orthogonal sequences performing marginally better. Of course we have to mention that the perfect sequences and Walsh sequences are constant modulus sequences while the optimal orthogonal sequences are not. The derived optimal training sequence in [54] belongs to the class of orthogonal sequences. The question of whether there exists any non orthogonal training sequence with better performances and how to find them is an open question.

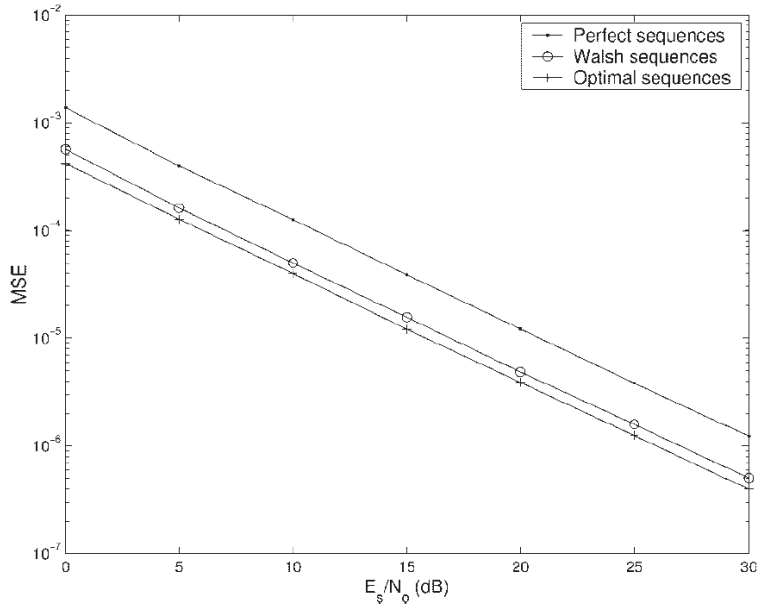


Figure 20: MSE performance of the ML estimator for the different training sequence

### 4.2.1.3 Non Data Aided Symbol Timing

The data aided algorithms are based on orthogonal training sequences and hence transmission efficiency is decreased. It also requires the epoch of training sequence to be perfectly known. Any misalignment of the orthogonal training sequence with the local copies causes serious performance degradation due to the loss of orthogonality. It, therefore, motivates us to develop a non-OFDM symbol timing estimation algorithm that does not rely on training sequences.

In fact, the well-known Oerder and Meyr [52] algorithm presents a non data aided symbol timing algorithm in SISO channels. Here we extend this algorithm in MIMO channels based on [53].

With the same notation as (54) we have the magnitude square of the sampled received signal as:

$$\left|r_{m,s}\right|^2 = \left|\sqrt{\frac{E_S}{N_t}}\sum_{n=1}^{N_t} h_{nm}\sum_i d_{n,i}\, p(sT_s/Q - iT_s - \eta T_s) + w_{m,s}\right|^2 \qquad (81)$$

where $d_n$ is the encoded information by a ST trellis or block encoder at the $n^{\text{th}}$ receive antenna. By taking expectation with respect to the joint distribution of channel coefficients, data and

noise and noting that the channel coefficients are independent and uncorrelated with noise and also the encoded data $\{d_i\}$ are uncorrelated for different values of n, we have:

$$E\left[\left|r_{m,s}\right|^2\right]=\frac{E_S}{N_t}\sum_{n=1}^{N_t}\left\{E\left[\left|h_{nm}\right|^2 E\left[\left|d_{n,i}\right|^2\right]\right]\sum_i\left|p(sT_s/Q-iT_s-\eta T_s)\right|^2\right\}+{\sigma_w}^2 \tag{82}$$

where $\sigma_w^2$ is the noise power. If the statistical properties of encoded symbols $d_{n,i}$ and the channel coefficient $h_{nm}$ are independent of $n$ and $m$ then (82) can be rewritten as:

$$E\left[\left|r_{m,s}\right|^2\right]=E_S K\sum_i\left|p(sT_s/Q-iT_s-\eta T_s)\right|^2+{\sigma_w}^2 \tag{83}$$

$K$ is a constant number. With the use of Poisson sum formula, and with the fact the $p(t)$ is bandlimited to $f$=1/$T$, (83) can be rewritten as:

$$\begin{aligned}&E\left[\left|r_{m,s}\right|^2\right]=E_S K[z_0+z_1\cos(2\pi m/Q-2\pi\eta)]+{\sigma_w}^2\\&z_q=\frac{1}{2\pi}\int_{-\infty}^{\infty}P(\omega)P^*(\omega-q\frac{2\pi}{T_s})d\omega\end{aligned} \tag{84}$$

Where $P(\omega)$ is the Fourier transform of $p(t)$. Since $p(t)$ is symmetric $P(\omega)$ is real and $z_q$ are real too. From (84), as ling as $z_1\neq0$, which is true for raised cosine pulse with excess bandwidth $\alpha>0$, the squared signal contains spectral lines at symbol rate ($f$=1/$T_s$) and its phase is closely related to $\eta$. Thus we can estimate the timing delay by computing the Fourier coefficient at symbol rate as the case of signal-transmit-single-receive system antenna [52]. It follows that

$$\hat{\eta}=-\frac{1}{2\pi}\arg\left(\sum_{s=rNQ}^{(r+1)NQ-1}\left|r_{m,s}\right|^2 e^{-j2\pi s/Q}\right) \tag{85}$$

Where $N$ is the observation length and $r$ is any non negative integer. Further more, if the same clock is used at all receivers. The squared signal from different antennas can be averaged before estimation to increase the diversity as:

$$\left|r_s\right|^2=\frac{1}{N_r}\sum_{m=1}^{N_r}\left|r_{m,s}\right|^2 \tag{86}$$

Wu in [54] develop a true ML approach for non data aided symbol timing and he shows that (86) is a special case of his algorithm. For describing his algorithm we use the same notation as equation (74) but in this case no training sequence is used and **Z** contains real data. In this case **Z** and **H** can not be separated so we define **x** as *vec*(**ZH**) here instead of (77) we have to minimize following criterion:

$$p(\mathbf{r}|\eta',h)=\frac{1}{\left(\pi\sigma_w{}^2\right)^{NQ}}\exp\left(-\frac{\left\|\mathbf{r}-\ddot{\mathbf{A}}_{\eta'}\mathbf{x}\right\|^2}{\sigma_w{}^2}\right) \tag{87}$$

With the linear model of equation (86) the ML estimation of **x** can be found and by putting it into (87) and after some calculation and dropping irrelevant terms we have:

$$\Lambda_{NDA}(\eta')=\sum_{m=1}^{N_r}\mathbf{r}_m{}^H\mathbf{A}_{\eta'}\left(\mathbf{A}_{\eta'}{}^{\mathbf{H}}\mathbf{A}_{\eta'}\right)^{-1}\mathbf{A}_{\eta'}{}^{\mathbf{H}}\mathbf{r}_m \tag{88}$$

And non data aided ML symbol training estimator can be written as:

$$\hat{\eta}=\arg\max_{\eta'}\Lambda_{DA}(\eta') \tag{89}$$

Like data aide case $ML_{NDA}$ can be implemented by the two-step approach.

Note that the implementation of the $ML_{NDA}$ estimator does not require the knowledge of correlation among antennas. Note also that the likelihood function in Equation (88) is the sum of individual likelihood functions for each receive antenna, just as the case of training-based likelihood function in Equation (78). Furthermore, applying the low complexity maximization technique [57] to the likelihood function (88) and with the approximation $\left(\mathbf{A}_{\eta'}{}^{\mathbf{H}}\mathbf{A}_{\eta'}\right)=I_{N+2N_g}$ for Nyquist zero-ISI pulse, it can be easily shown that the $ML_{NDA}$ (89) reduces to the extension of squaring algorithm proposed in Reference [53]. The effect of increasing the number of transmitter and receiver in $ML_{NDA}$ estimator is the same as data aided estimator.

## 4.2.1.4 Comparison

Here, we compare the performance of the $ML_{DA}$ and $ML_{NDA}$ estimators with their corresponding CCRBs and MCRBs for a 4 x 4 system. For simplicity, it is assumed that there is no correlation among antennas and there is no space-time coding for NDA case (since the effects of these are small as shown earlier). Figure 21 shows the results. Note that from Figure 21, the MSE performances of $ML_{DA}$ and $ML_{NDA}$ estimators are very close to their corresponding CCRBs. This means that $ML_{DA}$ and $ML_{NDA}$ are efficient estimators conditioned that the nuisance parameters are being jointly estimated together with the unknown timing delay. Also, note that the performance of $ML_{DA}$ estimator is very close to the $MCRB_{DA}$, which implies that $ML_{DA}$ is almost the best possible estimator under the problem at hand, regardless of how we deal with

the nuisance parameters. For the NDA case, unfortunately, although the performance of $ML_{NDA}$ estimator reaches the corresponding $CCRB_{NDA}$, the $CCRB_{NDA}$ is quite far away from the $MCRB_{NDA}$. It means that there is a possibility that some other NDA estimators (probably employing higher order (>2) non-linearity) would have performances closer to the MCRB.

If we want to compare data aided estimator and non data aided, as expected, $ML_{DA}$ estimator performs much better than the $ML_{NDA}$ estimator. However, there are some advantages for non data aided estimator as follows: i) The $ML_{DA}$ estimator requires training sequences, resulting in lower transmission efficiency. ii) The estimation has to be performed at specific times when the training data is available, while $ML_{NDA}$ can be performed at any time during transmission. iii) For the DA case, there is a need to synchronize the training sequences before timing estimation. This requires extra implementation complexity. In addition, degradation may occur if the positions of the training sequences are dislocated.

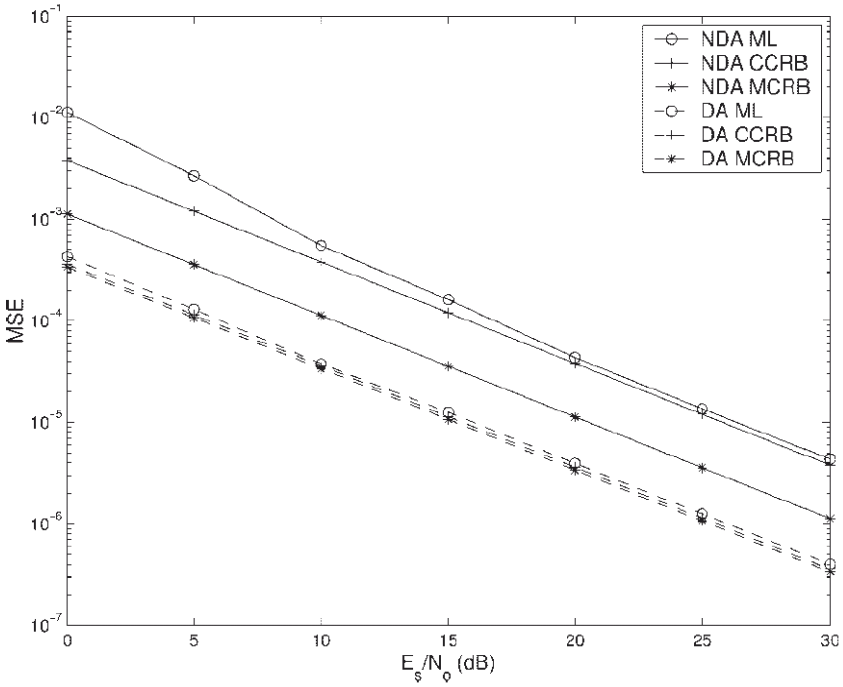


Figure 21: Comparison of MSEs of the $ML_{NDA}$ and $ML_{DA}$ and their corresponding CCRBs and MCRBs for a 4 × 4 system.

Therefore $ML_{DA}$ and $ML_{NDA}$ provide a performance, transmission efficiency and complexity trade-off for symbol timing estimation in MIMO channels.

There are some few other papers which discuss about symbol timing in MIMO systems like [58]. In this dissertation I just present the most important ones. There is a major problem in all of the papers which have been seen in the literature that all of them investigate synchronization problem for the case of flat fading channel and there is no reference in the literature (by the knowledge of the author) about symbol timing over frequency selective Rayleigh channels.

### 4.2.2 non-MIMO OFDM time synchronization

We already noted that in an OFDM system, thanks to using Fourier transformation, a time shift smaller than sampling rate time can be converted to a shift in phase in frequency domain. Therefore, in OFDM systems the delay smaller than sampling rate will be absorbed in channel coefficients phase and there is no need to estimate it in time synchronization stage.

Numerous techniques have been suggested in the literature for Single Input Single Output (SISO) OFDM time synchronization, however a lot of them can not be applied without major modification to Multi input Multi output systems (for example [22],[23]).

Within SISO OFDM time synchronization algorithms we can distinguish two approaches in literature. As a matter of fact, a number of methods for OFDM time synchronization have been proposed for OFDM system in general (for example the methods that exploit the periodic structure of cyclic prefixes and algorithms based on the use of repeated preambles). These techniques were originally developed for general OFDM systems, but they can be also applied to IEEE 802.11a WLANs. In the second approach, there are time synchronization techniques that are specifically designed for IEEE 802.11a WLAN.

#### 4.2.2.1 Time Synchronization for general SISO-OFDM systems

By exploiting the known structure of a training symbol or a cyclic prefix, several schemes have been proposed for coarse estimation of the carrier frequency offset and time synchronization [22]-[36]. Moose in [24] proposes the repetition preamble scheme for the first time for frequency synchronization. He describes a technique to estimate frequency offset by using a repeated training sequence and derives a maximum likelihood estimation procedure. Then, this idea was used for time synchronization by Schmidle [23], Speth [33] and Keller [35]. The principle of exploiting a double training sequence preceded by a cyclic prefix (CP) for frame synchronization was originally suggested for single-carrier transmission in [36].

It seems that the best way for time synchronization is to calculate correlation between a known reference and received sequence. But the presence of frequency offset reduces the peak of correlation function. In fact, this frequency offset prevents coherent addition of individual term in correlation calculation and results in a drop of the correlation peak so Shmidl [23] proposed to use the preambles like Figure 22, and calculate autocorrelation function (instead of cross correlation) as:

$$y\ (\tau) = \sum_{i=1}^{N_p} r^*(\tau - i + 1) r\ (\tau - N_p - i + 1) \tag{90}$$

where $r(\tau)$ is received signal and $N_P$ is the size of each preamble. According to (90) the coarse time is the instant where y reaches to its peak. This method is robust against frequency offset but the drawback of this method is that the correlation function exhibits a plateau at the arrival of the data packet [29],[8]. In fact during the CP period, the correlation function remains constant and there is no abrupt falling edge outside this period. This edge can not be localized precisely in low signal quality. This phenomenon, inherent to the structure of training sequence, makes it difficult to find the fine timing. Therefore, this method can only be used for frame synchronization (coarse search) and for fine timing or symbol timing one have to use another method.

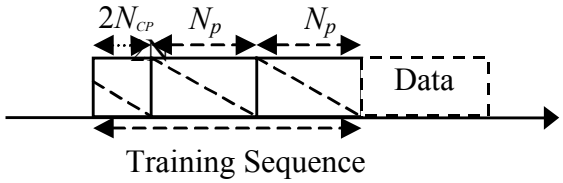


Figure 22: the repeated preamble used in the literature

Muller-Weifurtner in [25] provided a ranking for different metrics in repetition preamble method which proposed in literature [26],[23],[35],[36] via a simulation assessment. In fact he compared Minimum Mean Squared criterion [36], Maximum-Likelihood criterion [26], Maximum Correlation criterion [35] and Shmidle's criterion [23] with each other. He proved that ML metrics in [26] performs the best and for moderate noise is asymptotically identical with the slightly less complex MMSE metric in [36].

Similar to the signaling set up adopted in [23], Coulson used two repeated m-sequences as a training symbol [31],[32]. However the proposed time synchronization algorithm is likely to fail in the presence of large carrier frequency offsets, and presents high implementation complexity due to the matched filtering. Moreover, a training sequence with the same structure as the one proposed in [31] is exploited [22] to develop reliable frequency and time acquisition scheme. However, the proposed time synchronization algorithm is also sensitive to large frequency offset.

Recently Shi and Serpedin in [37] based on [29] have proposed using the preambles as follows:

$$[\pm\mathbf{B}\ \pm\mathbf{B}\ \pm\mathbf{B}\ \pm\mathbf{B}] \tag{91}$$

where **B** stands for a sequence of $N_p/4$ training samples with constant variance (amplitude) (e.g., an m-sequence) and it can be generated with good approximation by using an $N_p/4$-point IFFT of an m-sequence. They claimed that their method exhibits nearly the same performance as [31] in term of estimation accuracy (better than [23] and [29]). In addition, the proposed estimator assumes a reduced implementation complexity and is more robust against large frequency offset in comparison with [31].

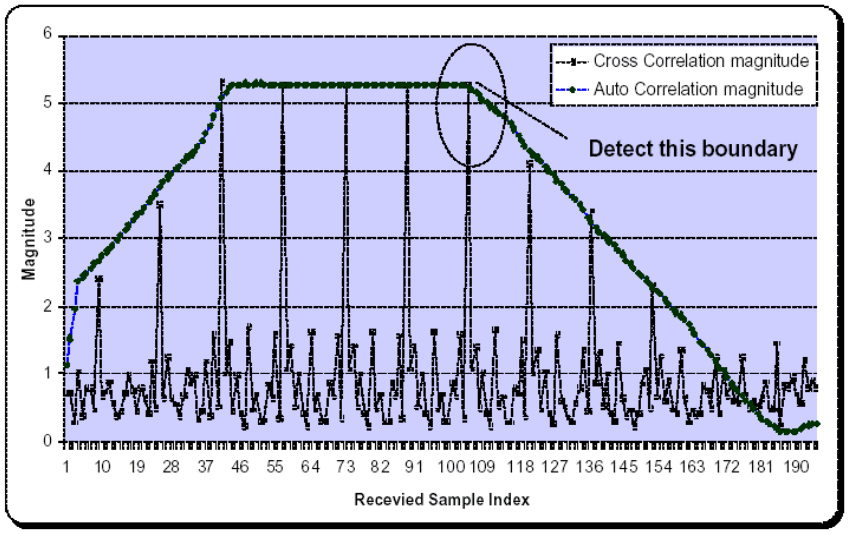


Figure 23: Using cross correlation for fine timing

**Symbol timing (fine timing):** In a SISO OFDM system, fine timing can be performed by cross-correlation the received signal and transmitted ones after frequency synchronization (Figure 23) or estimate the power of channel impulse responses as [38 pp 88-92], or by finding the magnitude of channel impulse response larger than a proper threshold [48],[49]. In [34] based on [33] Speth performs fine timing with a post-FFT algorithm using pilots and estimated channel.

## 4.2.2.2 Time synchronization for IEEE 802.11 WLAN

OFDM is employed as the transmission technique in the IEEE 802.11a WLAN standard. In the IEEE 802.11a standard, the preamble is dedicated to various synchronization tasks. However, there is no mention in the standard how to use the preamble to achieve synchronization.

IEEE 802.11a is a packet-based communication system. Each packet is preceded by a preamble as defined in IEEE 802.11a specification [79]. The preamble structure shown in Figure 24 consists of two parts. The first part comprises 10 short training symbols, each of length 800ns;

in the second part a cyclic prefix of length 1.6μs is followed by two long training symbol each of length 3.2μs.

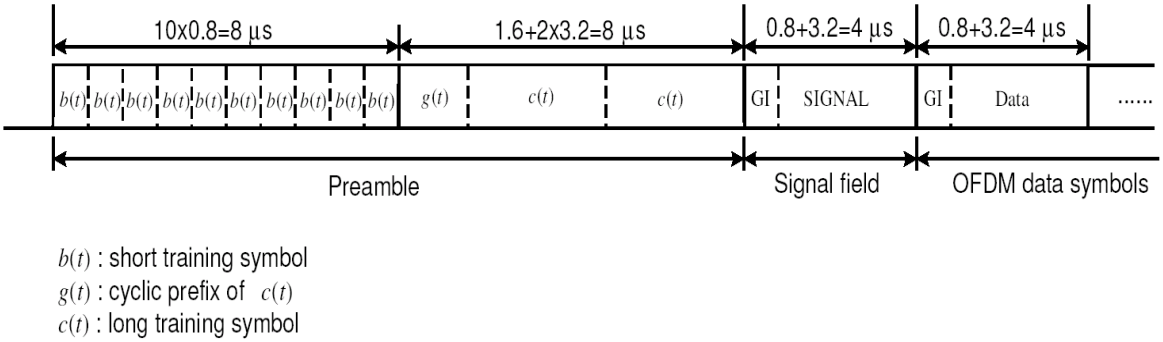


Figure 24: Packet structure for IEEE 802.11a WLANs

There are several papers which propose synchronization algorithms specifically designed for IEEE 802.11a WLAN [39]-[45].

Nandula in [39] proposes a frame timing algorithm using auto-correlation and cross-correlation of short preambles. In the absence of an analytical discussion in [39] it seems that this algorithm is not robust against frequency offset. However he claimed that his algorithm by using an average, can withstand somewhat against frequency offset. Wang in [40] has proposed a coarse time synchronization scheme based on the short training symbols. He calculated two normalized auto-correlation timing metrics. The first metrics $M_1$ is the normalized correlation between the received signal and itself with a delay of one short training which creates a plateau of the length of nine short training. The second metric $M_2$ is the normalized correlation between the received signal and itself with a delay of two short training which creates a plateau of the length of eight short symbols. By subtracting $M_2$ from $M_1$ a triangular shaped timing metric is obtained, which its pick indicates start of 9-th short symbol as Figure 25.

In [40], for fine timing synchronization, some schemes have been proposed such as conventional cross correlation method with the long training symbol or the method proposed in

[38] that selects the window which contains the maximum power of the estimated channel response.

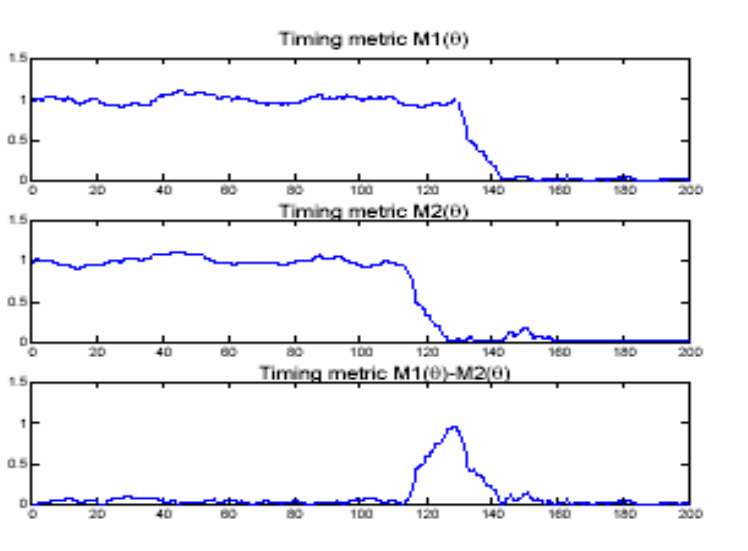


Figure 25: Timing metrics in [40]

The most recent approaches to WLAN synchronization are discussed in [28] and [42] in which a maximum likelihood symbol synchronizer is developed for IEEE 802.11a WLANs in frequency selective fading channel by using the ideas in [27]. Wu in these articles derives a ML criterion to estimate both channel length and the arrival packet time by using the short training symbols. The observation vector length in the proposed algorithm is 16 (rather than 64 in [27]) so compared with [27] he reduced complexity considerably with the same performance. In fact he used a two-stage algorithm. In the first stage, the current time offset with respect to the last short training symbol and thus the beginning of the next expected short training symbol is determined. In this stage he derived a general ML criterion by using an observation vector of length 16. In the second stage, he determines if the incoming vector belongs to a short training symbol or a cyclic prefix of a long training symbol. The authors of this article have shown by simulation that this ML algorithm has good results with rather low complexity.

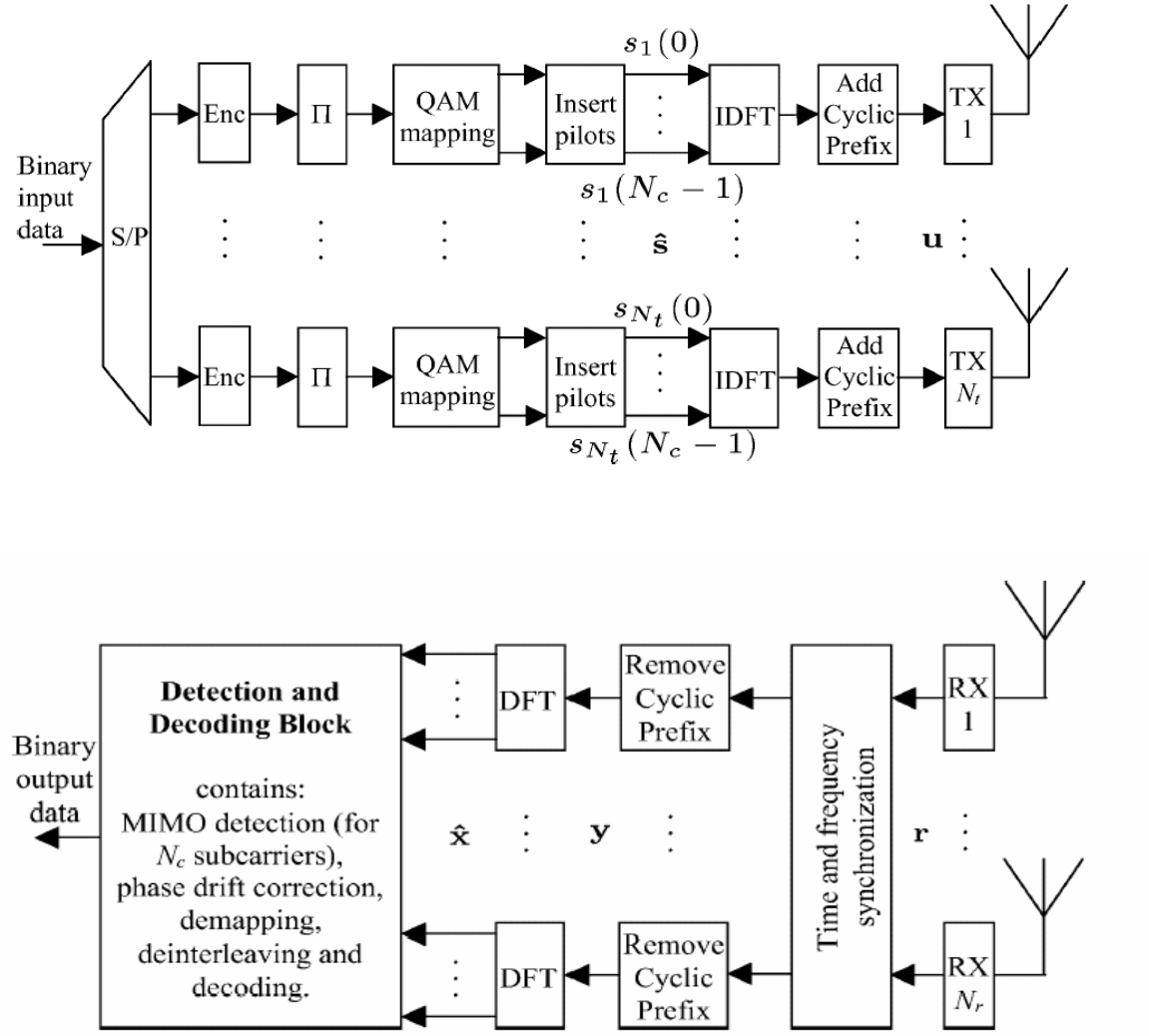


Figure 26: A typical MIMO-OFDM transmitter and receiver

# 4.3 MIMO OFDM frame synchronization

A typical MIMO-OFDM transmitter and receiver system is represented in Figure 26.

As previously said the OFDM time synchronization algorithms often include two steps. The first step is Frame Synchronization/Coarse Timing. The task of the frame synchronization is to identify the preamble in order to detect a packet arrival. This preamble detection algorithm can also be used as a coarse symbol timing since inherently provides a rough estimate of the starting

point of the packet. The second step is symbol timing which indicates where to place the start of the OFDM window within the OFDM symbol. An OFDM system is well known for its ability to combat inter-symbol interference (ISI) introduced by multipath channel, nevertheless incorrect positioning of the FFT window within an OFDM symbol reintroduces ISI during data demodulation, causing serious performance degradation [33]. This section is dealt with frame synchronization and the following section is about symbol timing.

The first paper which published about MIMO OFDM time synchronization in multipath channel, to the best of our knowledge, was [45] by Mody. He used a simple MIMO extension of Schmidl's synchronization algorithm [23] by using the modulatable orthogonal sequence proposed by Suehiro [46]. For fine time acquisition he used cross correlation techniques. Zelst in a article on the implementation of a MIMO OFDM based wireless LAN system in frequency selective channels [47] presented another extension of Schmidl's synchronization algorithm. Zelst's method is based on calculation of autocorrelation between two repeated preambles. The maximum of autocorrelation should normally show the beginning of the packet. The periodicity of preambles makes his algorithm robust against the frequency offset. In the following section we well describe the classical autocorrelation method. Then in the next sub-section, an advanced version of autocorrelation method which has been developed by us and presented in [7] will be discussed.

Moreover, we presented another new method for frame synchronization in [6]. This method first was presented in a conference and then it has been elaborated in [4] and submitted as an invited paper in [2]. This algorithm in detail will be discussed in the chapter 5.

### 4.3.1 Autocorrelation method

In the autocorrelation method which is largely accepted as one of the best methods for frame synchronization a training sequence that is orthogonal and shift orthogonal is needed. Moreover, to estimate the MIMO channel, it is important that each path from the different TX antennas to each RX antenna can be uniquely identified. There are several other methods to construct the training sequence; some of them were described in 4.2.1.2. Anyway, there are at

least two constraints to design a sequence as a preamble. The first condition is the orthogonality between the preambles of different transmit antennas and the second condition is shift-orthogonality for at least the length of the channel (to remove the effects of multipath channel). A straightforward way to generate the preamble is to turn off all other transmitters while one transmitter is transmitting the preamble. The resulted preamble generated by this method is showed in the Figure 27. This way is the simplest method of generating preambles because the orthogonality between deference antennas is assumed. This method results in simpler time synchronization algorithm but since the total length of the proposed preamble grows linearly with $N_t$, it is not highly efficient.

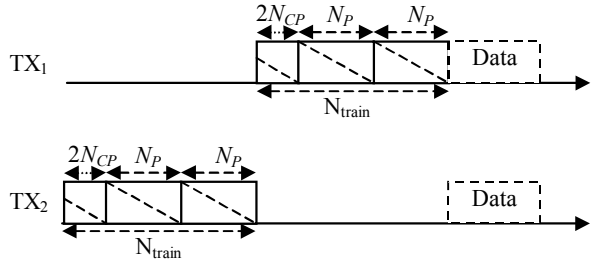


Figure 27: Time orthogonal preamble suitable for MIMO systems with two transmit antennas

To design orthogonal sequences, apart from the methods presented in 4.2.1.2, we present here a simple method based on reference [112] to construct a orthogonal code with length $N_P^2$. This kind of codes is called Frank-Zadoff codes.

In order to generate this kind of code we construct the following matrix:

$$\begin{bmatrix} 1 & 2 & \dots & N_P \\ 2 & 4 & \dots & 2N_P \\ . & . & . & . \\ N_P & 2N_P & & N_P^2 \end{bmatrix} \tag{92}$$

We form a long vector by putting side by side the rows of matrix (92) :

$$P = \begin{bmatrix} 1 & 2 & \dots & N_P & 2 & 4 & \dots & 2N_P & \dots & N_P & 2N_P & \dots & N_P^2 \end{bmatrix} \tag{93}$$

Then, the $s^{\text{th}}$ symbol of preamble will be obtained by the following formula:

$$B(s) = (e^{2\pi jP(s)/N_P}) \qquad s = 0, 2, \dots, N_P^2 - 1 \tag{94}$$

Having constructed the Frank-Zadoff codes, we use it as the training sequence of the first transmit antenna and the other transmit antennas' training sequence will be the shifted version

of the first training sequence. One has to consider that the amount of shift must be greater than the channel length.
Using the orthogonal training sequence, the correlation function $\Lambda$ can be defined as:

$$\Lambda(k) = \sum_{m=1}^{N_r} \sum_{s=k}^{k+N_p-1} r_{m,s} r^*_{m,s-N_p} \tag{95}$$

where $r_{j,k}$ is the received signal at the $m^{\text{th}}$ receive antenna and at the $k^{\text{th}}$ instant. $N_P$ is the length of a preamble and since we use two preamble plus cyclic prefix the length of the training sequence is $2N_p+2N_{CP}$ (Figure 28). $s$ is the beginning of a sliding window over received signal.

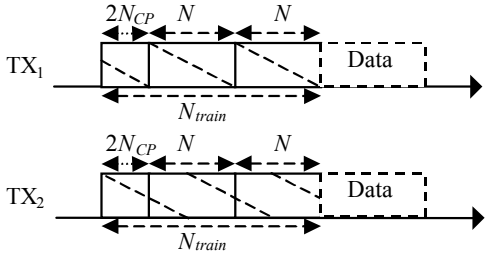


Figure 28: Orthogonal preamble for MIMO systems with two transmit antennas

The beginning of the packet will be:

$$t_{start} = \arg\max_k \Lambda(k) \tag{96}$$

However, the drawback of this method is that the correlation function does not produce a sharp peak at the arrival of the data packet. In fact, during the CP period, the correlation function remains constant and there is no abrupt falling edge outside this period. This edge can not be localized precisely in a cheap signal quality. This phenomenon, inherent to the structure of training sequence, makes it difficult to find the fine timing and we need necessarily another algorithm to perform fine timing (symbol timing). We will discuss about MIMO OFDM symbol timing algorithms in section 4.4 but in the following sub-section a new method presented by us in [7] will be discussed. In this new method which is called *advanced autocorrelation method*, contrary to the autocorrelation method, we do not have a plateau during CP period and the new correlation method produces a sharp peak.

### 4.3.2 Advanced autocorrelation method

In absence of frequency carrier offset, the best way for symbol-timing is to calculate correlation between a known reference and received sequence. However the presence of frequency offset reduces the peak of correlation function. In fact, this frequency offset prevents coherent addition of individual term in correlation calculation and results in a drop of the correlation peak. This method proposes an advanced auto-correlation method in the presence of frequency offset and produces a sharp peak in contrast with conventional autocorrelation method which is used for coarse timing (frame synchronization). Therefore, in this method the coarse time synchronization and fine time synchronization step are merged into a single step and our frame synchronizer and time synchronizer are merged into a simple time synchronizer. The main drawback of this method is that it is designed for flat fading channels. However, according to the experimental results presented in [8] it has a good performance in frequency selective channels .

#### 4.3.2.1 Signal Model

The received signal by the $m^{th}$ antenna at instant $\tau$ can be written as:

$$r_{m,\tau} = \sum_{n=1}^{N_t} \sum_{l=0}^{L-1} u_{n,\tau-l} h_{nm}(l) + w_{m,\tau} \tag{97}$$

where $w_{m,\tau}$ represents complex additive Gaussian noise at instant $\tau$ and at the $m^{\text{th}}$ receive antenna with variance $(1/2)\sigma_w^2$, stationary and independent from the other antennas. $h_{nm}(l)$ is the $(n,m)^{th}$ element of the matrix **H(*l*)** and represents the path between the $n^{th}$ transmit and $m^{th}$ receive antenna This equation can be written for all the receive antennas as follows:

$$\mathbf{r}(\tau)=\sum_{l=0}^{L-1}\mathbf{H}(l)\mathbf{u}(\tau-l)+\mathbf{w}(\tau) \tag{98}$$

where $\mathbf{r}(\tau)$ and $\mathbf{w}(\tau)$ are column vectors of dimension $N_r$ and $\mathbf{u}$ is a column vector of dimension $N_t$.In the presence of frequency error due to local oscillator deviation or Doppler effect, each received symbol undergoes a phase rotation proportional to the delay between the transmit and receive instants and the frequency error. The received complex vector can be written as follow:

$$\mathbf{r}(\tau)=e^{j\,2\pi\Delta f\,\tau}\sum_{l=0}^{L-1}\mathbf{H}(l)\mathbf{u}(\tau-l)+\mathbf{w}(\tau) \tag{99}$$

where $\Delta f$ is frequency offset.

#### 4.3.2.2 Advanced auto correlation synchronizer

Instead of conventional autocorrelation function, we propose a new function, which has a very sharp response, even in the presence of relatively large frequency offset. We define the function $y_{mn}$ for the $m^{th}$ receive antenna while the $n^{th}$ transmit antenna is considered:

$$y_{mn}(\tau)=\sum_{k=1}^{N_p-1} r^{*}_{m,\tau-k+1}c_{n,N_p-k}r_{m,\tau-k}c^{*}_{n,N_p-k-1} \tag{100}$$

where $c_{n,k}$ represents the cyclic prefix sequence at the $n^{\text{th}}$ transmit antenna and at the $k^{\text{th}}$ instant. and $r_{n,k}$ represents the the $k^{\text{th}}$ sample of received signal at the $m^{\text{th}}$ receive antenna. $N_{\text{p}}$ is the length of each preamble. Considering a training sequence as presented in Figure 27, it is clear that the received signal in each instant is only the function of one training sequence of corresponding emitting transmitter. In order to simplify notation assuming the time reference at the instant where the preamble arrives at the receivers ($\tau=0$), Assuming a flat fading channel, $r_{m,\tau}$ when the $n^{th}$ transmitter is on can be written as follows:

$$r_{m,\tau} = c_{n,\tau} h_{nm}(1) e^{j2\pi\tau\Delta f T_s} + w_{m,\tau} \tag{101}$$

Replacing (101) in (100) and we obtain:

$$y_{mn}(\tau)\Big|_{\tau=N_p-1} = y_{mn}^{\max} = \sum_{k=1}^{N_p-1} \left|c_{n,N_p-k}\right|^2 \left|c_{n,N_p-k-1}\right|^2 \left|h_{nm}(1)\right|^2 e^{-j2\pi\Delta f T_s} + w'_{m,N_p-1} \tag{102}$$

Hence, $y_{mn}(N_P\text{-}1)$ presents a sharp peak at packet arrival independent of the channel phase coefficient and frequency error. Using the principle of maximum ratio combining and since the phase effect due to channel and frequency error is already cancelled out, we can add up $y_{mn}(\tau)$ directly to form our decision function for the $n^{th}$ transmit antenna.

$$y_n(\tau) = \sum_{m=1}^{N_r} y_{mn}(\tau) \tag{103}$$

In order to take advantage of the other codes sent by the other transmit antennas, we can sum up a shifted version of all the $y_n(\tau)$ to obtain our final decision function:

$$y(\tau) = \sum_{n=1}^{N_t} y_n(\tau - (n-1)N_{train}) \tag{104}$$

Figure 29 represents $y(\tau)$. The correlation function proposed in literature is presented also to make clear the decision algorithm. As soon as maximum correlation function exceeds a fixed threshold, we search for a peak in (104) that gives the exact packet arrival.

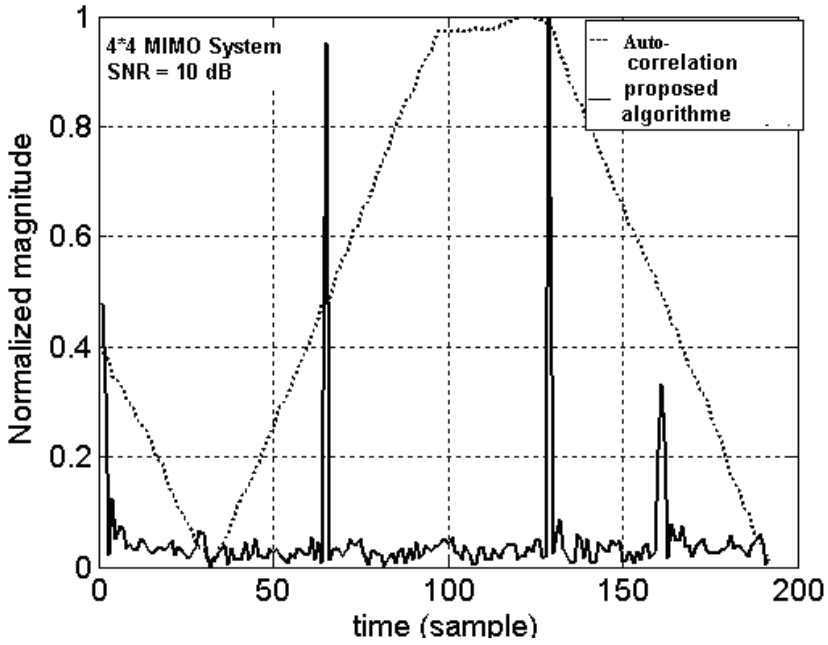


Figure 29- Peak of the $y(n)$ (equation (104)) helping to detect the exact timing

#### 4.3.2.3 Training sequence design for advanced auto correlation synchronizer

Here we describe how we can construct the training sequence for the first transmit antenna. The training sequence should be designed so that it has a maximum in $\tau = N_P - 1$ according to equation (102) and it is about zero near the maximum. Defining $c'_{1,k} = c_{1,k} c^*_{1,k-1}$, the required condition is:

$$\sum_{\substack{p=0 \\ q\neq 0}}^{N_P-1} c'_{1,p} {c'_{1,(p+q) \bmod N_P}}^* = 0 \qquad (105)$$

It guaranties a sharp peak of the function defined by (100) and consequently by (104). Applying this code to $c'(p)$, the training sequence $c(p)$ can be calculated by means of a simple recursion algorithm as follow:

- $c_{1,0} = 1$
- $c_{1,k} = \dfrac{{c'_{1,k-1}}^*}{c^*_{1,k-1}}$

#### 4.3.2.4 Advanced Autocorrelation method in frequency selective channel

Although our algorithm is designed for flat fading channels, we wanted to quantify the performance degradation when this algorithm is applied directly to frequency selective channels. The performance is measured by the probability of time synchronization failure, which can be compared with the results of autocorrelation time synchronization algorithm presented in [48].

Table 3 shows the results [8]. The simulations have been performed for a 2×2 MIMO system with SNR=6 dB. Exponential decay channel was considered with Rayleigh fading coefficients. The rms delay spread range varies between 50 and 250 ns and sampling rate is fixed to 20 MHz. As it can be seen from the table, this algorithm quite outperforms the autocorrelation method especially in low dispersive channel, as expected.

Table 3: Synchronization failure probability in autocorrelation and proposed algorithms

| $\tau_{rms}$ | $P_{fail\ in\ conventional\ time\ synchronization\ algorithm}$ | $P_{fail\ in\ proposed\ time\ synchronization\ algorithm}$ |
|---|---|---|
| 50 ns | $9.4 \times 10^{-3}$ | $1.7 \times 10^{-3}$ |
| 100 ns | $6.4 \times 10^{-2}$ | $1.7 \times 10^{-2}$ |
| 150 ns | $1.2 \times 10^{-1}$ | $6.1 \times 10^{-2}$ |
| 200 ns | $1.8 \times 10^{-1}$ | $1.0 \times 10^{-1}$ |
| 250 ns | $2.4 \times 10^{-1}$ | $2.0 \times 10^{-1}$ |

## 4.4 MIMO OFDM symbol timing

To overcome the Inter-Symbol-Interference (ISI) most OFDM based systems apply a cyclic prefix (CP) in front of every symbol. This CP, often also referred to as guard interval (GI), introduces redundancy, which is removed in the receiver, before data detection. In this way the influence of the ISI caused by the multipath channel can be largely reduced.

The CP, however, also significantly decreases the effective data rate of the system. It is, therefore, important that the ratio between the length of the CP and the number of carriers is minimized. One solution is to keep the CP length low compared to the channel impulse response (CIR) length. Then, however, ISI will possibly become the performance limiting factor and the placement of the discrete Fourier transform (DFT) window within the stream of received OFDM symbols, here referred to as *symbol timing*, becomes important. Note that symbol timing in literature is sometimes referred to as fine timing, in contrast to coarse timing which indicates the packet detection.

Several frame timing approaches for single-input single output (SISO) OFDM have been proposed previously in literature. Generally, they are based on a maximizing of a timing measure which is found by either correlation between repeated dedicated training symbols, or correlation between the redundant parts in the data symbols. The limited accuracy of these algorithms makes their applicability to systems with short CP lengths questionable. In this chapter we illustrate several symbol-timing which exhibit good performance.

The symbol timing algorithms that will be presented in this section can be divided into two types. The methods that will see first use the knowledge of channel coefficients. In fact, the idea of this kind of methods firstly appeared in [38, pp 88-92]. Zelst extended the technique proposed in [38] over MIMO channels. Since, this technique relies on the knowledge of the channel response, their powers are estimated by correlating the received signals with the known training sequence and subsequently the powers of all paths in MIMO channel are summed.

Schenk in [121] elaborated on this idea and presented several symbol timing methods for MIMO OFDM systems. In the following section we will present his symbol timing methods. Then, in the next subsection a new symbol timing algorithm, introduced by us in [5], will be presented.

## 4.4.1 Methods with the knowledge of channel coefficients

### 4.4.1.1 Signal Model

Consider a MIMO OFDM system with $N_t$ transmitter (TX) and $N_r$ receiver (RX) branches, $Nc$ subcarriers and a CP of length $N_{CP}$ samples. The vector of (complex) data symbols in the frequency domain is denoted by **u** here, the size of which equals $N_cN_t\times1$. After the inverse discrete Fourier transform (IDFT) and the addition of the CP the $N_sN_t \times 1$ signal vector is given by **u**, where the symbol length is denoted by $N_s = N_c + N_{CP}$. The upconverted signal is transmitted through the wireless channel and downconverted at the RX, here modeled by transmission through the baseband equivalent channel **H**. The $N_r \times1$ received time-domain vector during sample time $k$ is expressed by:

$$\mathbf{r}(k) = \left[ r_{1,k},\ r_{2,k},\ \ldots,\ r_{N_r,k} \right]^T = \sum_{l=0}^{L-1} \mathbf{H}(l)\mathbf{u}(k-l) + \mathbf{w}(k) \qquad (106)$$

where $r_{m,k}$ represents the received signal at the $m^{\text{th}}$ receive antenna and during $k^{\text{th}}$ instant. $\mathbf{u}(k)$ and $\mathbf{r}(k)$ denote the $N_t\times1$ and $N_r\times1$ transmitted and received vector during sample time $k$, $L$ is the length of the channel and $T$ denotes the matrix transpose. For the following it is useful to group these vectors per OFDM symbol. Therefore, we introduce the following short notation for the $k^{\text{th}}$ sample of the $p^{\text{th}}$ OFDM symbol at the $m^{\text{th}}$ and $n^{\text{th}}$ receive and transmit antenna:

$r^p_{m,k} = r_{m,(p-1)Ns+k}$ and $u^p_{n,k} = u_{m,(p-1)Ns+k}$.

Using this notation, the noiseless received signal in (106) can be rewritten for the $k^{\text{th}}$ sample of the $p^{\text{th}}$ received symbol vector as:

$$\mathbf{r}^p(k) = \left[ r^p_{1,k},\ r^p_{2,k},\ \ldots,\ r^p_{N_r,k} \right]^T = \sum_{l=0}^{L-1} \mathbf{H}(l)\mathbf{u}^p(k-l) + \sum_{l=0}^{L-k-1} \mathbf{H}(l+k)\mathbf{u}^{p-1}(N_s - l) \tag{107}$$

for $k = 1, \ldots, N_s$ and where we assumed $L \leq N_s$. $\mathbf{r}^p{}_{m,k}$ represents the $k^{\text{th}}$ sample of the $p^{\text{th}}$ OFDM symbol at the $m^{\text{th}}$ receive antenna We can conclude from (107) that the received signal contains samples from the regarded and the previous symbol, i.e., the first term is the useful signal term and the second term is the ISI. The symbol timing now determines which $N_c$ out of $N_s$ samples of $\mathbf{r}^p$ are used to determine the frequency domain signal vector. This vector is then used by the MIMO processing to estimate the transmitted data.

If the symbol timing is chosen to be $k_m$, the $N_c$ –dimensional vector input to the DFT for the $m^{\text{th}}$ RX is given by:

$$\mathbf{r}^p_{m,k_m} = \left[ r^p_{m,k_m},\ \ldots,\ r^p_{m,k_m+N_c-1} \right]^T = \mathbf{r}^{p\ \ sig}_{m,k_m} + \mathbf{r}^{p\ \ isi}_{m,k_m} \tag{108}$$

where the $\mathbf{r}^{p\ \ sig}_{m,k_m}$ and $\mathbf{r}^{p\ \ isi}_{m,k_m}$ are the desired and ISI vector, respectively.

Their $k^{\text{th}}$ elements are given by:

$$\begin{aligned} \mathbf{r}^{p\ \ sig}_{m,k_m}(k) &= \sum_{l=0}^{k_m+k-1} \mathbf{H}_{:m}{}^T(l)\mathbf{u}^p_{k_m+k-l}, \\ \mathbf{r}^{p\ \ isi}_{m,k_m}(k) &= \sum_{l=0}^{L-(k_m+k)-1} \mathbf{H}_{:m}{}^T(l+k_m+k)\mathbf{u}^{p-1}_{N_s-l} \end{aligned} \tag{109}$$

for $k = 0, \ldots, N_c - 1$, respectively. Here the $m^{\text{h}}$ column of $\mathbf{H}$ is denoted by $\mathbf{H}_{:m}$ .

As a performance measure of the timing we now consider the power of the desired signal and ISI terms. Hereto we calculate the expected power value of the desired signal term for timing point $p$ and a given channel realization $\mathbf{H}$. Since the DFT is applied per RX branch, it is derived here for the $m^{\text{th}}$ RX branch. For a timing point $k$, the expected signal power is given by:

$$\begin{aligned} P^{sig}_m(k) &= \mathsf{E}\left[ \mathbf{r}^{p\ \ sig\ H}_{m,k} \mathbf{r}^{p\ \ sig}_{m,k} \,\middle|\, \mathbf{H}_{:m} \right] \\ &= N_c\sigma_u^2 \sum_{i=0}^{k} \mathbf{H}_{:m}(i)^H \mathbf{H}_{:m}(i) + \sigma_u^2 \sum_{i=1}^{N_c+1} (N_c - i)\mathbf{H}_{:m}(i+k)^H \mathbf{H}_{:m}(i+k) \end{aligned} \tag{110}$$

where the elements of $u$ are assumed to be independently and identically distributed and to have zero-mean and a variance of $\sigma_u^2$.

In a similar way the expected value of the ISI power can be derived as:

$$\begin{aligned} P_m^{isi}(k) &= \mathsf{E}\left[ \mathbf{r}_{m,k}^{p\ isi\ H} \mathbf{r}_{m,k}^{p\ isi} \,\middle|\, \mathbf{H}_{:m} \right] \\ &= N_c \sigma_u^2 \sum_{i=N_c+k}^{k} \mathbf{H}_{:m}(k)^H \mathbf{H}_{:m}(k) + \sigma_u^2 \sum_{i=1}^{N_c-1} i \mathbf{H}_{:m}(i+k)^H \mathbf{H}_{:m}(i+k) \end{aligned} \tag{111}$$

The expressions for the wanted signal and ISI power, in two previous equations are defined for $1 \le k \le N_{CP}$. Note that for $k > N_{CP}$ these expressions change slightly, since samples of the next symbol will be included in the DFT window, also causing ISI.

Clearly the expressions for the expected power of the desired signal and ISI in two previous equations depend on the channel realization.

Knowledge of the MIMO channel coefficients is, therefore, required to be able to apply these expressions for symbol timing. In a practical system, however, full knowledge of the channel is often not available and an estimate of the channel has to be acquired.

Let us now consider a packet transmission, where the channel can be assumed to be quasi static, i.e., the channel is constant during the reception of a packet. For such a system often a piece of known data, i.e., preamble is transmitted in front of the data part to enable estimation of the MIMO channel coefficients, which is also required for MIMO detection. MIMO channel estimation and the design of an efficient preamble has been the subject of many contributions over the last few years and details are, therefore, omitted here.

Although the different channel estimation/preamble combinations will result in a different performance, we can generally write the resulting estimate of the $l^{\text{th}}$ tap of the $N_t \times N_r$ MIMO channel matrix $\mathbf{H}[l]$ by $\tilde{\mathbf{H}}(l) = \mathbf{H}(l) + er(l)$ , where $er(l)$ is the estimation error in the $l^{\text{th}}$ tap of the channel coefficients, which is here assumed to be zero-mean circular symmetric complex Gaussian distributed with a variance of $\sigma^2$ . Clearly this variance decreases with increasing signal-to-noise ratio (SNR). We have to note that although only an estimate of channel is available, we will assume perfect channel knowledge in these sections.

#### 4.4.1.2 Dominant path detection

The straightforward method to determine the symbol timing is to relate the timing to the maximum path in the channel. This is based on the observation that paths in the CIR with the smallest delay will generally experience the lowest attenuation. Generally an offset of several samples is applied to take into account possible smaller taps preceding the path with the maximum power in the channel

##### 4.4.1.2.1 RX-branch timing

When this technique is applied for a MIMO system to calculate the symbol timing for one of the RX branches, i.e., *m*, the estimated symbol timing point is given by:

$$\hat{k}_m = \arg\max_k \left\{ \mathbf{H}_{:m}(k)^H \mathbf{H}_{:m}(k) \right\} - c + N_{CP} \tag{112}$$

where $c$ is the above mentioned offset parameter. Furthermore, the constant $N_{CP}$ is added since the outcome of the argmax indicates the beginning of the OFDM symbol, rather than the beginning of the DFT window. This is also the case for following equations. It is noted that for the *RX-branch timing* every RX branch uses separate symbol timing.

##### 4.4.1.2.2 Joint timing

To make the processing in the MIMO RX less complex, a common symbol timing for the whole MIMO RX can be calculated. The timing point is then found as:

$$k = \arg\max_{\tilde{k}} \left\{ \sum_{m=1}^{N_r} \mathbf{H}_{:m}(\tilde{k})^H \mathbf{H}_{:m}(\tilde{k}) \right\} - c + N_{CP} \tag{113}$$

#### 4.4.1.3 SIR (Signal to ISI Ratio) optimization

A more optimal approach to symbol timing in the case of perfect channel knowledge is proposed here. It maximizes the SIR (Signal to ISI ratio) at the input of the DFT, i.e., the method finds a timing point that minimizes the amount of ISI and maximizes the amount of signal falling into the DFT window.

##### 4.4.1.3.1 RX-branch timing

For this approach the symbol timing point for the $n_r^{\text{th}}$ RX branch is found by:

$$\hat{k}_m = \arg\max_k \left\{ P_m^{sig}(k) / P_m^{isi}(k) \right\} \tag{114}$$

where the expected signal and ISI power are given by (110) and (111), respectively. An alternative approach would be to maximize the desired signal power in (110) or to minimize ISI power in (111).

##### 4.4.1.3.2 Joint timing

For the case of joint symbol timing for the entire MIMO receiver, the ratio of the total signal power and total ISI power is calculated. The symbol timing point is then found by:

$$\hat{k} = \arg\max_{\tilde{k}} \left\{ \sum_{m=1}^{N_r} P_m^{sig}(\tilde{k}) \Big/ \sum_{m=1}^{N_r} P_m^{isi}(\tilde{k}) \right\} \tag{115}$$

#### 4.4.1.4 Reduced complexity algorithm

A less computational complex algorithm, i.e., requiring less operations for implementation, and finds the maximum of the convolution of the CIR powers with a rectangular window of length $N_{CP}$, i.e., the length of the guard interval. In doing so, the algorithm attempts to maximize the amount of ISI within the CP, i.e., minimizing the ISI within the DFT window.

#### 4.4.1.4.1 RX-branch timing

In the extension for MIMO systems, the resulting timing point for the $n_r^{\text{th}}$ RX branch is:

$$\hat{k}_m = \arg\max_{\tilde{k}} \left\{ \sum_{i=\tilde{k}}^{\tilde{k}+N_{CP}} \mathbf{H}_{:m}(i)^H \mathbf{H}_{:m}(i) \right\} + N_{CP} \tag{116}$$

#### 4.4.1.4.2 Joint timing

The extension for joint timing is given by:

$$\hat{k} = \arg\max_{\tilde{k}} \left\{ \sum_{m=1}^{N_r} \sum_{i=\tilde{k}}^{\tilde{k}+N_{CP}} \mathbf{H}_{:m}(i)^H \mathbf{H}_{:m}(i) \right\} + N_{CP} \tag{117}$$

### 4.4.1.5 Simulation results

Here, our aim is to compare the different approaches of the methods with the knowledge of channel to find the best algorithm between them. The performance of the algorithms are evaluated by Monte Carlo simulations for a 4×4 MIMO extension of the IEEE 802.11a standard, i.e., for $N_c = 64$ and $N_{CP} = 16$.

In the following we will derive the performance of the symbol timing as function of the accuracy of the channel estimation. Therefore, we define the Channel estimation -to-Channel estimation Error Ratio (CCER) for the $n_r^{\text{th}}$ RX branch as:

$$CCER_m = \frac{E\left[ \sum_{l=0}^{L-1} \mathbf{H}_{:m}(l)^H \mathbf{H}_{:m}(l) \right]}{E\left[ \sum_{l=0}^{L-1} er_{:m}(l)^H er_{:m}(l) \right]} \tag{118}$$

which in fact equals the inverse of the normalized mean-squared-error (MSE) of the MIMO channel estimate for the $m^{\text{th}}$ branch. The CCER averaged over the different RX antennas is then denoted by CCER.

Although the MSE in symbol timing gives a measure for the error in the timing, it does not show the degradation in performance due to this error. Therefore, the following figure will

report the achieved SIR (Signal to ISI ratio) with a certain timing algorithm as function of the CCER. When the SIR has the same order of magnitude as or is larger than the experienced SNR, the system performance will be limited by the ISI. In Figure 30 the SIR performance is given for the channel delay equal to $\tau = 3N_{CP}/8$. The optimal SIR values obtained with perfect timing are depicted as dashed lines. It can be observed that the reduced complexity model achieves the bound for the lowest CCER value, closely followed by the SIR maximization method. It is observed that the dominant path search method shows flooring below the optimum, while it performs better than the SIR maximization algorithm for very inaccurate channel estimates.

By comparison of the performance of the joint (dashed line) and RX timing (solid line), it can be concluded that for the SIR maximization the joint timing performs a little worse than the separate timing for low CCER values. For the reduced complexity and dominant path detector this is the other way around.

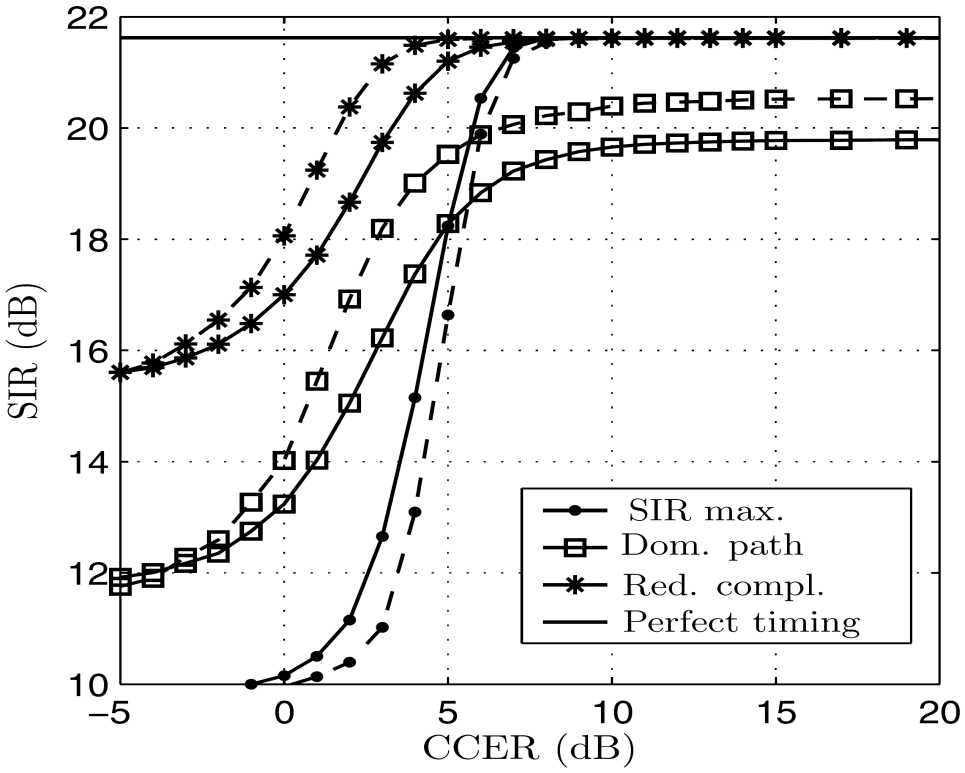


Figure 30: Comparison of symbol timing results obtained by the methods with the knowledge of channel

## 4.4.2 ML MIMO-OFDM symbol timing method

In ML MIMO OFDM symbol timing, we have assumed that the arrival of the preamble can be identified by detecting the received signal energy in the frame synchronization procedure. It means that, the coarse frame synchronization has succeeded to detect a packet arrival but the precise beginning is not known. This could be due to dispersive effect of the propagation channel and the use of a simple autocorrelation algorithm.

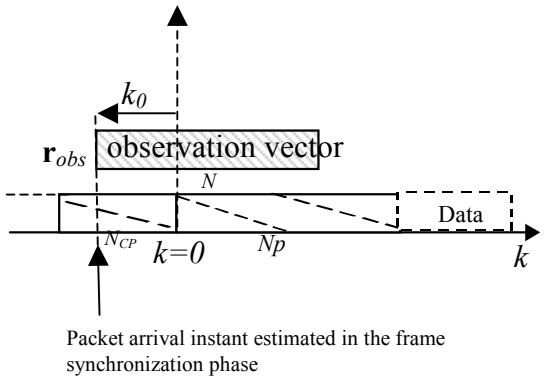


Figure 31: Illustration of symbol timing problem

Suppose the beginning of the preamble, as shown in Figure 31, is chosen as the time reference, i.e. $k=0$. The receiver takes a signal vector of size $N$ from estimated instant in frame synchronization procedure as the observation vector ($\mathbf{r}_{obs}$). Assume $k_0$ as time offset of observation vector with respect to beginning of the training sequence (Figure 31). We have to perform symbol timing in order to find the exact place of the beginning of the training sequence packet. By knowing the exact place of the training sequence packet, we are able to align the start of FFT window within the OFDM symbol. The received observation vector at the $m^{\text{th}}$ receive antenna, $\mathbf{r}_{m,obs}$, begins from $k_0^{\text{th}}$ sample from the beginning of the training sequence packet, in other words $\mathbf{r}_{m,obs} = [r_{m,k_0} \ \ldots \ r_{m,k+N-1}]^T_{N\times 1}$. The objective of our algorithm is to estimate $k_0$ from the observation vector $\mathbf{r}_{m,obs}$.

Suppose $k_0 \in [-N_{CP},\ N_P - N]$ the training sequence of $n^{\text{th}}$ transmit antenna is $\begin{bmatrix} c_{n,0} & c_{n,2} & \dots & c_{n,N_P-1} \end{bmatrix}$ where $N_P$ is the length of training sequence and the length of cyclic prefix is $N_{CP}$, the joint ML estimate of $k_0$ and $\mathbf{h_0}$ can be obtained by maximizing:

$$p(\mathbf{r_{obs}} \left| k, \mathbf{h} \right.) = \frac{1}{(\pi \sigma_w^{\ 2})^N} \exp\left\{ - \frac{\sum_{m=1}^{N_r} \left\| \mathbf{r}_{m,obs} - \sum_{n=1}^{N_t} \mathbf{C}_{n,k} \mathbf{h}_{nm} \right\|^2}{\sigma_w^{\ 2}} \right\} \tag{119}$$

where:

$$\mathbf{C}_{n,k} = \begin{bmatrix} c_{n,mod(k,N_P)} & c_{n,mod(k-1,N_P)} & \cdots & c_{n,mod(k-L+1,N_P)} \\ c_{n,mod(k+1,N_P)} & c_{n,mod(k,N_P)} & \cdots & c_{n,mod(k-L+2,N_P)} \\ . & . & . & . \\ c_{n,mod(k+N-1,N_P)} & c_{n,mod(k+N-2,N_P)} & \cdots & c_{n,mod(k-L+N,N_P)} \end{bmatrix}_{N \times L} \tag{120}$$

$\mathbf{h}_{nm} = \begin{bmatrix} h_{nm}(0) & h_{nm}(1) & \dots & h_{nm}(L-1) \end{bmatrix}^T_{L \times 1}$ and $h_{nm}(l)$ is the equivalent channel between $n^{\text{th}}$ transmit antenna and $m^{\text{th}}$ receive antenna.

The development of this algorithm is somewhat similar to the ML-MIMO-OFDM joint algorithm that will be presented in the next chapter. Therefore, we avoid here describing the mathematical detail of algorithm. By maximizing (46) we find the following criterion to estimate $k_0$:

$$\hat{k}_0 = \arg\max_k \Psi(k) \tag{121}$$

where:

$$\Psi(k) = \sum_{m=1}^{Nr} \mathbf{r}_{m,obs}^H \mathbf{C}_k (\mathbf{C}_k^H \mathbf{C}_k)^{-1} \mathbf{C}_k^{\ H} \mathbf{r}_{m,obs} \tag{122}$$

**SYMBOL SYNCHRONIZATION PERFORMANCE CRITERION**

In [28] Erchin Serpedin states that since in Rayleigh multipath fading channel, the channel may contain some small taps at the beginning, the starting position of the channel is not clear. So, the beginning of the packet can be defined in so many ways: as the first non-zero tap of the channel, as the first tap with energy larger than a certain threshold, as the position of the

strongest path or any other definition. In MIMO case the problem is even more complicated because we have several different paths. Therefore, the symbol boundary of a received OFDM symbol is not well defined. According to [28], even if we choose one of the above definitions as the reference position, there is no guarantee that a certain synchronization algorithm giving estimates close to the reference position would provide good performance in OFDM systems. Moreover in OFDM systems, due to the existence of cyclic prefix, some timing offset can be tolerated as long as the samples within the FFT window are influenced by only one transmitted OFDM symbol. Therefore the criterion that the synchronization error has to be within certain limits of a fixed reference point is not an appropriate performance measure for OFDM systems in frequency selective fading channels. Hence, Erchin Serpedin uses the loss in system performance due the synchronization error as a criterion for symbol timing. Serpedin derives his criterion for the SISO systems. Here, we extend this criterion on the MIMO systems and then we evaluate the performance of our algorithm by using this criterion.

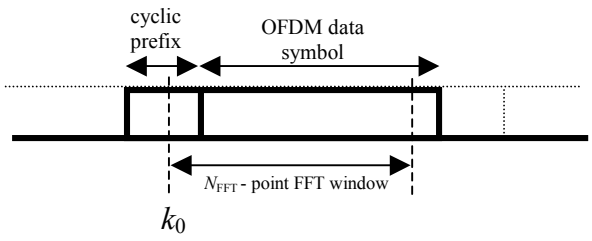


Figure 32: OFDM symbol and FFT position

Suppose that the fast Fourier transform (FFT) window starts at position $k_0$ (Figure 32), the signal at the sub-carrier $p$ after FFT operation has been written in [101] for SISO case. Using the linearity properties we extend it over MIMO system, so the signal at the sub-carrier $p$, at the $m^{\text{th}}$ receiver, after FFT operation, $z_m[p]$, can be written as:

$$z_m[p] = e^{j2\pi(p/N_{FFT})k_0} \sum_{n=1}^{N_t} \alpha_{nm}(k_\eta) a_n[p] H_{nm}[p] + \mathrm{I}_m[p] + v_m[p] \qquad (123)$$

where $a_n[p]$ is the transmitted data from $n^{\text{th}}$ transmit antenna at sub-carrier $p$, $H_{nm}[p]$ is the channel transfer function between $n^{\text{th}}$ transmit and $m^{\text{th}}$ receive antenna at sub–carrier $p$, $w_m[p]$

is the noise sample at sub-carrier $p$ and $m^{th}$ receive antenna. $N_{FFT}$ is the number of FFT points in the OFDM system. $\alpha_{nm}(k_0)$ is the attenuation caused by the synchronization error, which is approximated for SISO in [101] and we use it for MIMO system as well:

$$\alpha_{nm}(k_0) = \sum_{l=0}^{L-1} |h_{nm}(l)|^2 \frac{N_{FFT} - \Delta\kappa_l}{N_{FFT}} \tag{124}$$

where:

$$\Delta\kappa_l = \begin{cases} k_0 - l & k_0 > l \\ l - N - k_0 & k_0 < -(N-l) \\ 0 & otherwise \end{cases} \tag{125}$$

and $\mathrm{I}_m[p]$ is the ISI plus inter-carrier interference (ICI) term at sub-carrier $p$ and $m^{th}$ receive antenna because of timing offset that can be well approximated by Gaussian noise with power [101]:

$$\sigma^2_{I_m}(k_0) = \sum_{n=1}^{N_t} \sum_{l=0}^{L-1} |h_{nm}(l)|^2 \left( 2\frac{\Delta\kappa_l}{N_{FFT}} - (\frac{\Delta\kappa_l}{N_{FFT}})^2 \right) \tag{126}$$

The signal to interference-plus-noise ratio (SINR) per receiver can be written as:

$$SINR(k_0) = \frac{1}{N_r} \sum_{m=1}^{N_r} \frac{\sum_{n=1}^{N_t} \alpha^2_{nm}(k_0) E\left\{ |a_n[p] H_{nm}[p]|^2 \right\}}{\sigma^2_{I_m}(k_0) + \sigma^2_w} \tag{127}$$

We can rewrite also SINR by using (137):

$$SINR(k_0) = \frac{1}{N_r} \sum_{m=1}^{N_r} \frac{E\left\{ |z_m[p]|^2 \right\} - \sigma^2_{I_m}(k_0) - \sigma^2_w}{\sigma^2_{I_m}(k_0) + \sigma^2_w} \tag{128}$$

When we know the channel coefficients, which is the ideal case, we can calculate $k_0$ for the ideal MIMO synchronizer by maximizing SINR. Note that for each of receive antennas ($m$), $E\{|z_m[p]|^2\}$ is independent of $m$ and $p$ and thus we can maximize the following term to maximize SINR.

$$\Gamma(k_0) = \sum_{m=1}^{N_r} \frac{1}{\sigma_{I_m}^2(k_0) + \sigma_w^2} \tag{129}$$

Using the equations (128), (124)-(126) we can calculate $k_0$ of the ideal MIMO synchronizer numerically by maximizing $\Gamma(k_0)$. The ideal symbol synchronizer can be served as a reference to our practical synchronization algorithms.

For a particular realization of channel, let $k_0$ be the start of FFT window estimated by our symbol synchronization algorithm and $k_{id}$ be that of the ideal symbol synchronizer. Then the loss of SINR, defined as the ratio of SINR obtained from the ideal symbol synchronizer to that from non-ideal synchronizer is given by:

$$SINR_{loss}(k_0) = \frac{SINR_{id}}{SINR(k_0)} \tag{130}$$

Noting that $E\left\{\left|a_n[p]\right|^2\right\}$ is independent of $n$ and $E\left\{\left|H_{nm}[p]\right|^2\right\} = \sum_{l=0}^{L-1} h_{nm}^2(l)$ by using (127) we have:

$$SINR_{loss}(k_0) = \sum_{m=1}^{N_r} \frac{\sum_{n=1}^{N_t} \alpha_{nm}^2(k_{id})}{\sigma_{I_m}^2(k_{id}) + \sigma_w^2} \Bigg/ \sum_{m=1}^{N_r} \frac{\sum_{n=1}^{N_t} \alpha_{nm}^2(k_0)}{\sigma_{I_m}^2(k_0) + \sigma_w^2} \tag{131}$$

As In [28] and [43] we assume that a synchronization failure happens when the probability of loss in SINR is greater than a certain thresholds, that is:

$$P_f(\Delta\gamma) = P(10\log_{10}(SINR_{loss}) > \Delta\gamma) \tag{132}$$

where $P_f(\Delta\gamma)$ is the probability of synchronization failure when the tolerable system degradation (in dB) is $\Delta\gamma$.

**SIMULATION RESULTS**

We provide a few Monte Carlo simulation results to illustrate the effectiveness of this algorithm. In all of the simulations, the channel between a certain transmit and receive antenna is modelled using an exponentially decaying power delay profile (PDP) with independent Rayleigh fading on every tap. We assume that each channel has $L = 15$ and the channel delay is 50 nS, and a. The channel is fixed during transmission of one packet and independent of that of

another packet. Furthermore it is assumed that the length of FFT window or the OFDM symbol is 64 and there is a cyclic prefix for each OFDM symbol with length 16.

As preamble, we use a single sequence with the length of an OFDM symbol and we use a cyclic prefix of length 32 for our preamble. The preamble is generated randomly for each transmit antenna. Our sampling vector length, $N$, is 64. So, the index $k$ in (121) belongs to [32,0] and also $k_0$ is treated as a uniform random variable over this interval and value of $k_0$ was randomly generated in each iteration. For each simulation run, the loss of SINR is calculated using (131), where the ideal symbol synchronizer uses $k_{id}$ such that (129) is maximized. Each point of result is obtained by averaging over $10^4$ Monte-Carlo runs.

Since this algorithm is somewhat similar to the algorithms presented [28] and [27], we compare it with them.

Figure 33 represents the result of [28] and [27] for IEEE 802.11a specification SISO system. The channel models are considered with 100nS and 30nS delay spread respectively.

Figure 34 presents our simulation results for a $4 \times 4$ MIMO system, but in comparison with

Figure 33 we have considered the channel models with 100 and 150 nS delay spread and the curve shows the $P_f\,(0.05dB)$ in contrast with $P_f\,(0.5\text{dB})$ for the case of SISO. As it can be seen in the figure, the probabilities of failure are very small. Note that as it is described in [28] the curves of $P_f$ in general have a U shape form. This is because: At low SNRs, due to high level of noise the synchronizer is not accurate and the error in time leads to a reduction in $\alpha_{nm}$ and thus an increase in the amount of loss (see(131)). At low SNRs $\alpha_{nm}$ has a major role in the increase of loss while at high SNR, in addition to $\alpha_{nm}$, $\sigma^2_{I_m}$ has a considerable effect on the increase of loss. In fact, in high SNRs $\sigma^2_{I_m}$ is greater than $\sigma^2_w$ and thus $\sigma^2_{I_m}$ will be directly proportional to the increase in loss. Therefore, eve though at high SNRs the estimated positions can be quite accurate, a small error with respect to ideal position leads to a large amount of loss in SINR.

Finally we want to mention that although we can not compare directly our results with that of [28] and [27] because of different assumptions, but it is evident from the figures that the performance of the proposed algorithm is much better than the SISO case. This improvement in performance is quite reasonable since we have used a 4×4 MIMO system instead of SISO one.

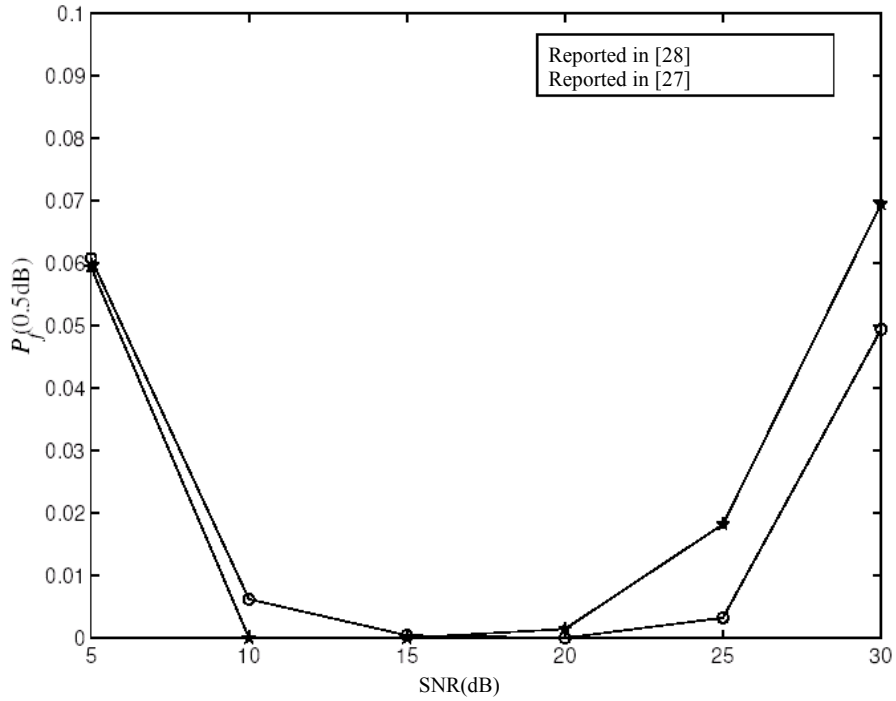


Figure 33 : $P_f(0.5)$ for the algorithms in [28] and [27] as a function of SNR

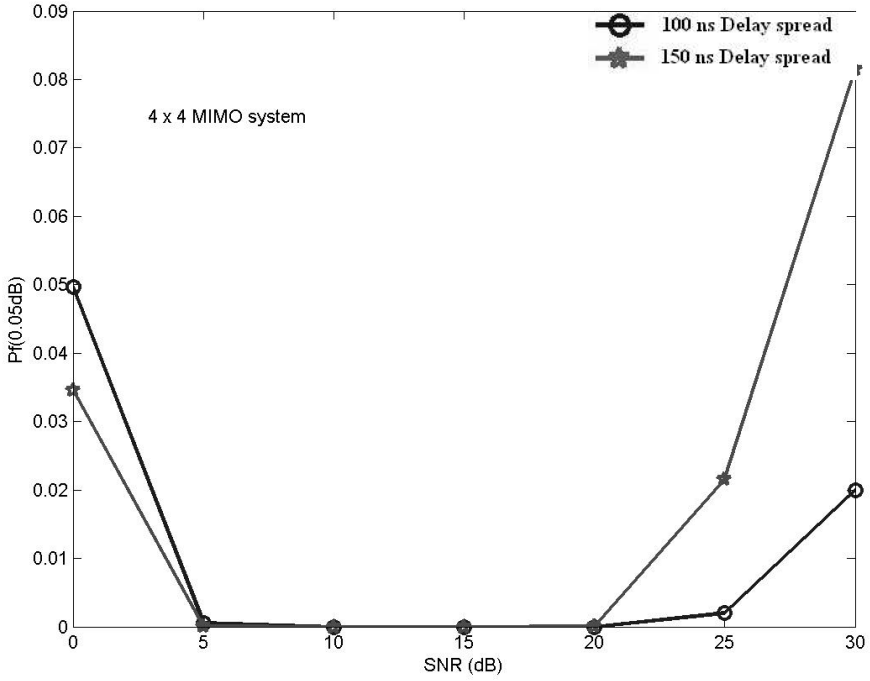


Figure 34: $P_f$(0.05dB) for the proposed algorithm in 4×4 MIMO system with 100 and 150 ns delay spread as a function of SNR

*Chapter*


# JOINT TIME-FREQUENCY MIMO–OFDM SYNCHRONIZATION

## 5.1 Introduction

We saw in the previous chapter that a number of methods for OFDM synchronization have been proposed in the literature. The algorithms used for frame synchronization are not able to determine the precise packet time arrival. This is a critical step since we have to perform symbol timing after these algorithms. This chapter is our effort to establish a more sophisticated time synchronization algorithm in MIMO–OFDM systems for both frame synchronization and symbol timing. In addition to time synchronization, frequency synchronization and channel estimation are addressed in this chapter. Part of this chapter has been presented in [4] but the most part of this chapter will be submitted in [2].

# 5.2 Signal model

We consider a MIMO-OFDM communication system with $N_t$ transmit and $N_r$ receive antennas. At each receiving antenna, a superposition of faded signals from all transmit antennas plus noise is received. Let $s_n(t)$ be the baseband-equivalent signal of the preamble at nth transmitter. This signal will be passed through a transmission filter and it will be up converted and sent to the multipath fading channel. At the receiver, the signal is downconverted into baseband and it will be passed through a receive filter. Let $h_{nm}(t)$ be the equivalent channel between the $n^{\text{th}}$ transmit antenna and the $m^{\text{th}}$ receive antenna receive antenna which is quasi-static, i.e. it is fixed during transmission of one packet and varies independently for each packet. The received signal $r_m(t)$ at $m^{\text{th}}$ receive antenna can be written as:

$$r_m(t) = \sum_{n=1}^{N_t} \int_{-\infty}^{+\infty} e^{j2\pi t\Delta f} s_n(t-u) h_{nm}(u) du + w_m(t) \qquad (133)$$

where the term $e^{j2\pi t\Delta f}$ denotes the phase rotation due to Carrier Frequency Offset (CFO).

As the previous chapters, we denote ε as: $\varepsilon = \Delta f \times T_s$ ($T_s$ is the sampling rate) and we use the term CFO equivalent with $\varepsilon$. It is worthy to mention that theoretically each transmit–receive antenna pair may have a different CFO [141]. In practice, however, the difference among these CFOs is usually negligible, because, according to Giannakis [117]: 1) even if collocated antennas do not share a radio frequency (RF) oscillator, achieving frequency synchronization among collocated oscillators is relatively easy; and 2) in a number of applications, the difference between Doppler shifts among all transmit–receive antenna pairs are approximately negligible, so, similar to some other researchers like [114] [117] [142], we have considered in our simulations a MIMO-OFDM system experiencing a single common CFO among transmit–receive antennas.

Now we suppose that the received signal is sampled at $t=kT_s+\eta_0 T_s$, , where $k$ is the number of sampled symbol and $\eta_0 \in [0,1)$ is the unknown time offset induced by the combination of channel delay and the sampling phase offset, so we have:

$$r_m(kT_s + \eta_0 T_s) = \sum_{n=1}^{N_t} \int_{-\infty}^{+\infty} e^{j2\pi k\varepsilon} s_n(kT_s - u') h_{nm}(u') du' + w_m(kT_s + \eta_0 T) \quad (134)$$

where $u' = u - \eta_0 T_s$. In an OFDM system due to FFT a delay in the time domain will be converted into an offset in phase in frequency domain. Therefore, $\eta_0$ will be appeared as a channel phase and we do not need to estimate $\eta_0$. Hence, the question of time synchronization in OFDM systems is reduced to estimate *k*. According to [143], if the equivalent channel bandwidth ($B_h$) satisfies $B_h < 1/T_s - B_s$ ($B_s$ is the bandwidth of $s(t)$ ), then by the equivalence of digital and analogue filtering for band-limited signals, the sampled received signal can be expressed as:

$$r_{m,k} = \sum_{n=1}^{N_t} \sum_{l=-\infty}^{\infty} e^{j2\pi k\varepsilon} s_n(kT_s - lT_s) h_{nm}(lT_s) + w_{m,k} \quad (135)$$

where $r_{m,k} \blacktriangleright r_m(kT_s + \eta_0 T_s)$, $w_{m,k} \blacktriangleright w_m(kT_s + \eta_0 T_s)$ and $w_m(t)$ is the stationary additive Gaussian noise at the $m^{\text{th}}$ receive antenna which is independent of the other antennas.

The fading multi-path channel is considered to be quasi static. Supposing the channel length equal to *L*, we have $N_t \times N_r$ paths, each of which can be modelled by an equivalent FIR complex filter of order $L-1$ with $h_{nm}(l) = h_{nm}(lT_s)$ as the taps with $l = [0, 1, \ldots, L-1]$. These taps are assumed to be independent zero mean complex Gaussian random variables with variance $1/2.P(l)$ per dimension. The set $P(l)$ with $l = [0, 1, \ldots, L-1]$ is called the power delay profile (PDP) of the channel and its total power is assumed to be normalized to $\sigma_c^2 = 1$, which is the average total channel power. Therefore (135) can be written as:

$$r_{m,k} = \sum_{n=1}^{N_t} \sum_{l=0}^{L-1} e^{j2\pi k\varepsilon} s_n(kT_s - lT_s) h_{nm}(l) + w_{m,k} \quad (136)$$

Let $\mathbf{r}_{m,k}$ be a received-signal vector of size *N*, sampled from time *k* to $N+k-1$ at $m^{\text{th}}$ antenna:

$$\mathbf{r}_{m,k} = \begin{bmatrix} r_{m,k} & r_{m,k+1} & \ldots & r_{m,k+N-1} \end{bmatrix}^T_{N \times 1} \quad (137)$$

Let the training sequence of $n^{\text{th}}$ transmit antenna be $\begin{bmatrix} c_{n,0} & c_{n,1} & \ldots & c_{n,N_P-1} \end{bmatrix}$ where $N_P$ is the length of training sequence. If the length of the cyclic prefix for the OFDM word is $N_{CP}$, we take $\begin{bmatrix} c_{n,N_P-N_{CP}} & \ldots & c_{n,N_P-1} \end{bmatrix}$ as the cyclic prefix of training sequence. Supposing $-N_{CP}+L-1 \le k \le N_P - N$, we have:

$$\mathbf{r}_{m,k} = \sum_{n=1}^{N_t} \mathbf{E}_k \mathbf{C}_{n,k} \mathbf{h}_{nm} + \mathbf{w}_{m,k} \tag{138}$$

where:

$$\mathbf{C}_{n,k} = \begin{bmatrix} c_{n,mod(k,N_P)} & c_{n,mod(k-1,N_P)} & \cdots & c_{n,mod(k-L+1,N_P)} \\ c_{n,mod(k+1,N_P)} & c_{n,mod(k,N_P)} & \cdots & c_{n,mod(k-L+2,N_P)} \\ . & . & . & . \\ c_{n,mod(k+N-1,N_P)} & c_{n,mod(k+N-2,N_P)} & \cdots & c_{n,mod(k-L+N,N_P)} \end{bmatrix}_{N \times L} \tag{139}$$

$$\mathbf{h}_{nm} = \begin{bmatrix} h_{nm}(0) & h_{nm}(1) & \cdots & h_{nm}(L-1) \end{bmatrix}^T_{L \times 1} \tag{140}$$

matrix $\mathbf{E}_k$ denotes the phase rotation due to Carrier Frequency Offset (CFO) as follows:

$$\mathbf{E}_k = \mathrm{diag}\begin{bmatrix} e^{j2\pi k\varepsilon} & e^{j2\pi(k+1)\varepsilon} & \cdots & e^{j2\pi(k+N-1)\varepsilon} \end{bmatrix}_{N \times N} \tag{141}$$

Index $k$ adds a constant phase that can be absorbed in channel coefficients, so it can be removed without loosing generality. $\mathbf{w}_{m,k}$ is a column vector containing the noise samples at the $m^{\text{th}}$ receive antenna with covariance matrix $\sigma^2_w \mathbf{I}_N$ ($\mathbf{I}_N$ is the $N \times N$ identity matrix).

Stacking the received vectors from all the $N_r$ receive antennas and using Kronecker product, we have:

$$\mathbf{r}_k = (\mathbf{I}_{N_r} \otimes \mathbf{E}_0 \mathbf{C}_k)\mathbf{h}_0 + \mathbf{w}_k \tag{142}$$

where:

$$\mathbf{r}_k = \begin{bmatrix} \mathbf{r}^T_{1,k} & \mathbf{r}^T_{2,k} & \cdots & \mathbf{r}^T_{N_r,k} \end{bmatrix}^T_{NN_r \times 1} \tag{143}$$

$$\mathbf{C}_k = \begin{bmatrix} \mathbf{C}_{1,k} & \mathbf{C}_{2,k} & \cdots & \mathbf{C}_{N_t,k} \end{bmatrix}_{N \times LN_t} \tag{144}$$

$$\mathbf{h}_0 = \begin{bmatrix} \mathbf{h}^T_{:1} & \mathbf{h}^T_{:2} & \cdots & \mathbf{h}^T_{:N_r} \end{bmatrix}^T_{LN_tN_r \times 1} \tag{145}$$

$$\mathbf{h}_{:m} = \begin{bmatrix} \mathbf{h}^T_{1m} & \mathbf{h}^T_{2m} & \cdots & \mathbf{h}^T_{N_tm} \end{bmatrix}^T_{LN_t \times 1} \tag{146}$$

$$\mathbf{w}_k = \left[ \mathbf{w}_{1,k}^T \quad \mathbf{w}_{2,k}^T \quad \ldots \quad \mathbf{w}_{N_r,k}^T \right]^T_{NN_r \times 1} \tag{147}$$

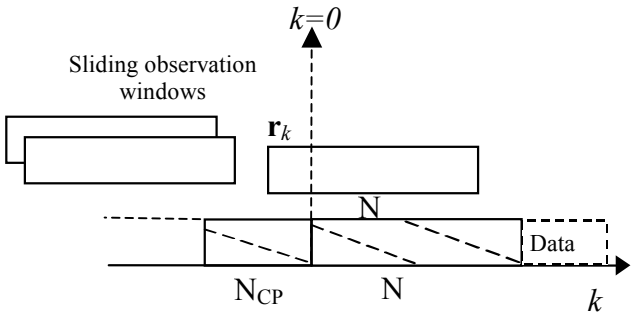


Figure 35- Finding the correct time instant ($k$=0) assuming sliding windows observations

# 5.3 ML Synchronizer

## 5.3.1 Packet arrival instant and initial CFO estimation

As already mentioned in literature, two stages of time synchronization for an OFDM system are normally distinguished. The first one is frame synchronization or coarse timing synchronization and the other one is symbol timing or fine timing synchronization. In this section we propose a new single-stage algorithm for time synchronization, that is, in our algorithm the result of frame synchronization is accurate enough to indicate the fine starting point of the packet. Hence, in our synchronization method, the two stages of synchronization are merged into a single stage.

The idea of calculation of complex correlation of two preambles, proposed in literature, is a good way to detect packet arrival. This method calculates the correlation of received signal by a delayed version of itself. The drawback of this method is that the correlation function does not produce a sharp peak at the arrival of the data packet and thus, during the cyclic prefix period, the correlation function remains constant and there is no abrupt falling edge outside this period. Due to this phenomenon, which is inherent to the structure of training sequence, it is difficult to find the fine packet time arrival instant. On the other hand, the performance of cross correlation

methods, which produces a sharp peak, is poor in dispersive and unknown channels [28]. Moreover, a frequency offset can degrade these methods considerably. We propose, hence, a new robust solution to find the exact time of packet arrival in unknown dispersive channels and in the presence of CFO.

Suppose we have a sliding window with length *N,* which starts from *k* to *k*+*N*. Hence, $\mathbf{r}_k$ is the observed received vector (Figure 35). We suppose also that the beginning of the preamble, as shown in (Figure 35), is chosen as the time reference, i.e. *k*=0. The objective, then, is to find the correct time instant *k*=0. To this purpose, we have to maximize the probability of receiving $\mathbf{r}_k$, given the preamble, the carrier frequency offset, the time offset and the channel coefficients. This probability can be calculated by using (142) as follows:

$$p(\mathbf{r}_k \mid k, \mathbf{h}, \varepsilon) = \frac{1}{(\pi\sigma_w^{\,2})^{N_r N}} \exp\left\{ -\frac{\left\| \mathbf{r}_k - (\mathbf{I}_{N_r} \otimes \mathbf{E}_0\mathbf{C}_0)\mathbf{h}_0 \right\|^2}{\sigma_w^{\,2}} \right\} \tag{148}$$

where $C_0$ is the training sequence matrix given in (144) can be written as:

$$\mathbf{C}_0 = \left[ \mathbf{C}_{1,0} \quad \mathbf{C}_{2,0} \quad \ldots \quad \mathbf{C}_{N_t,0} \right]_{N \times LN_t} \tag{149}$$

By varying *k* and maximizing the probability of equation (148) the arrival instant of the beginning of the packet can be estimated.

To estimate the arrival instant, instead of maximizing the probability we can equivalently minimize following metric:

$$J(\mathbf{r}_k \mid k, \mathbf{h}, \varepsilon) = (\mathbf{r}_k - (\mathbf{I}_{N_r} \otimes \mathbf{E}_0\mathbf{C}_0)\mathbf{h})^H (\mathbf{r}_k - (\mathbf{I}_{N_r} \otimes \mathbf{E}_0\mathbf{C}_0)\mathbf{h}) \tag{150}$$

Setting the partial derivative of $J(\mathbf{r} \mid \varepsilon, k, \mathbf{h})$ with respect to **h** to zero, the ML estimate for **h** (when *k* is fixed) is obtained as :

$$\hat{\mathbf{h}} = [(\mathbf{I}_{N_r} \otimes \mathbf{E}_0\mathbf{C}_0)^H (\mathbf{I}_{N_r} \otimes \mathbf{E}_0\mathbf{C}_0)]^{-1} (\mathbf{I}_{N_r} \otimes \mathbf{E}_0\mathbf{C}_0)^H \mathbf{r}_k \tag{151}$$

In order to have a unique solution for this equation, the number of observation points must be more than the number of unknown channel coefficients i.e.: $N \times N_r > L \times N_t \times N_r$.

Substituting equation (151) into equation (150), after some straightforward manipulations and dropping the irrelevant terms, the packet arrival time is estimated by maximizing the following likelihood function:

$$\Psi(k,\varepsilon)=\mathbf{r}_k^H(\mathbf{I}_{N_r}\otimes\mathbf{E}_0\mathbf{C}_0)[(\mathbf{I}_{N_r}\otimes\mathbf{E}_0\mathbf{C}_0)^H(\mathbf{I}_{N_r}\otimes\mathbf{E}_0\mathbf{C}_0)]^{-1}(\mathbf{I}_{N_r}\otimes\mathbf{E}_0\mathbf{C}_0)^H\mathbf{r}_k \tag{152}$$

Using the well-known properties of the Kronecker product $(A\otimes B)^{-1}=A^{-1}\otimes B^{-1}$, $(A\otimes B)^H=A^H\otimes B^H$ and $(A\otimes B)(C\otimes D)=(AC)\otimes(BD)$, we have:

$$(I_{N_r}\otimes\mathbf{E}_0\mathbf{C}_0)[(\mathbf{I}_{N_r}\otimes\mathbf{E}_0\mathbf{C}_0)^H(\mathbf{I}_{N_r}\otimes\mathbf{E}_0\mathbf{C}_0)]^{-1}(\mathbf{I}_{N_r}\otimes\mathbf{E}_0\mathbf{C}_0)^H=\mathbf{I}_{N_r}\otimes\mathbf{E}_0\mathbf{C}_0(\mathbf{C}_0^H\mathbf{C}_0)^{-1}\mathbf{C}_0{}^H\mathbf{E}_0^H \tag{153}$$

Substituting this result into equation (152) the likelihood function would be:

$$\Psi(k,\varepsilon)=\mathbf{r}_k{}^H(\mathbf{I}_{N_r}\otimes\mathbf{E}_0\mathbf{C}_0(\mathbf{C}_0^H\mathbf{C}_0)^{-1}\mathbf{C}_0{}^H\mathbf{E}_0^{\mathbf{H}})\mathbf{r}_k=\sum_{m=1}^{N_r}\mathbf{r}_{m,k}^H\mathbf{E}_0\mathbf{C}_0(\mathbf{C}_0^H\mathbf{C}_0)^{-1}\mathbf{C}_0{}^H\mathbf{E}_0^{\mathbf{H}}\mathbf{r}_{m,k} \tag{154}$$

The arrival packet instant ($k$=0) and the carrier frequency offset $\varepsilon$ can be calculated from the received signal vector $\mathbf{r}_k$ as follows:

$$(\hat{k},\hat{\varepsilon})=\arg\max_{k,\varepsilon}\Psi(k,\varepsilon) \tag{155}$$

Here we have to maximize a two dimensional function where one of the parameters is discrete and the other is continuous. We use the following approach: we know that $k$ can only take on values in a finite set $\mathbf{K}$, so we keep $k$ fixed to some value $\tilde{k}\in\mathbf{K}$, and we calculate $\varepsilon$:

$$\hat{\varepsilon}(\tilde{k})=\arg\max_{\varepsilon}\Psi\left(\tilde{k},\varepsilon\right) \tag{156}$$

We will shortly explain how we can calculate $\hat{\varepsilon}$ from equation (156) . The final estimate of $k$ then becomes:

$$\hat{k}=\arg\max_{\tilde{k}\in\mathbf{K}}\Psi\left(\tilde{k},\hat{\varepsilon}(\tilde{k})\right) \tag{157}$$

Since $k$ is a discrete parameter within a finite set we can estimate it by a numeric search. We propose the following method to calculate continuous parameter $\varepsilon$ from equation (156). We rewrite equation (156) by using equation (154):

$$\hat{\varepsilon}(\tilde{k}) = \arg\max_{\varepsilon} \Psi\left(\tilde{k},\varepsilon\right) = \arg\max_{\varepsilon}\left(\sum_{m=1}^{N_r} \mathbf{r}_{m,\tilde{k}}^{H}\mathbf{E}_0 \underbrace{\mathbf{C_0}(\mathbf{C_0}^{H}\mathbf{C_0})^{-1}\mathbf{C_0}^{H}}_{\Phi}\mathbf{E}_0^{H}\mathbf{r}_{m,\tilde{k}}\right)$$
$$= \arg\max_{\varepsilon}\left(\sum_{m=1}^{Nr}\sum_{p=0}^{N-1} r^{*}_{m,p+\tilde{k}}\sum_{q=0}^{N-1}[\phi]_{p,q}\, r_{m,q+\tilde{k}} e^{-j2\pi\varepsilon(q-p)}\right) \quad (158)$$

Considering Hermitain symmetry of $\Phi$ we have:

$$\Psi(\tilde{k},\varepsilon) = \sum_{m=1}^{N_r}\left\{\sum_{p=0}^{N-1}[\phi]_{p,p}\left|r_{m,p+\tilde{k}}\right|^2 + \underbrace{2\Re\left(\sum_{p=0}^{N-2}\sum_{q=p+1}^{N-1}[\phi]_{p,q}\, r^{*}_{m,p+\tilde{k}} r_{m,q+\tilde{k}} e^{-j2\pi\varepsilon(q-p)}\right)}_{\Psi'(\tilde{k},\varepsilon)}\right\} \quad (159)$$

$\Re(.)$ denotes the real part of a given quantity. Note that the first term in (159) is independent of $\varepsilon$ and has no role in the maximization of equation (158) so, to estimate $\hat{\varepsilon}(\tilde{k})$, we have to maximize the second term. By denoting $s = q - p$, we can further simplify the second term:

$$\Psi'(\tilde{k},\varepsilon) = \Re\left\{\left(\sum_{s=1}^{N-1} r'_s e^{-j2\pi\varepsilon s}\right)\right\} \quad (160)$$

Where we have $r'_s = \sum_{m=1}^{N_r}\sum_{p=0}^{N-1-s}[\phi]_{p,p+s} r^{*}_{m,p+\tilde{k}} r_{m,p+s+\tilde{k}}$. If we assume $r'_0$=0, equation (160) can be easily computed by using the FFT algorithm:

$$\hat{\varepsilon}(\tilde{k}) = \arg\max_{\varepsilon} \Psi'\left(\tilde{k},\varepsilon\right) = \arg\max_{\varepsilon}\left(FFT(r'_s)\right) \quad (161)$$

By using the estimated $\varepsilon$, the packet time arrival index $\hat{k}$ can be found from equation (157) . We make the following remarks:

- In our algorithm there is no constraint over training sequence structure. It means that we can use one preamble or two consecutive repeated preambles
- Better frequency resolution can be achieved by zero-padding to effectively increase the FFT length.
- FFT-based implementation, will significantly accelerate the search algorithm for $\hat{\varepsilon}$
- The orthogonality between the preambles of different antennas is not assumed.

- By calculating arrival instant, the channel can be estimated using (151)

## 5.3.2 Residual CFO estimation

So far, by using the algorithm presented in section5.3.1, the arrival packet and CFO and channel can be estimated. As we have seen just before, the frequency offset is estimated by equation (156) and to solve the problem of maximization we have presented an FFT-based algorithm. But, in the FFT algorithm, the accuracy of the frequency estimation is limited by the length of FFT. The more accuracy we need, the longer the FFT is and thus the more complexity is required.

It is possible however to propose two methods to solve the problem of maximization in (156) when the carrier frequency offset is small. Hence, in this section we are going to present a new simple iterative algorithm to estimate small carrier frequency offset in OFDM systems. we suppose that carrier frequency offset estimation is performed in two levels. In the first level, a coarse estimation is preformed by the FFT algorithm. As it will be shown shortly, a coarse frequency estimate is sufficient for time synchronization. In the second level, after compensating the frequency offset, we estimate CFO using more precise method. At this level, the problem of time synchronization does not exist any more.

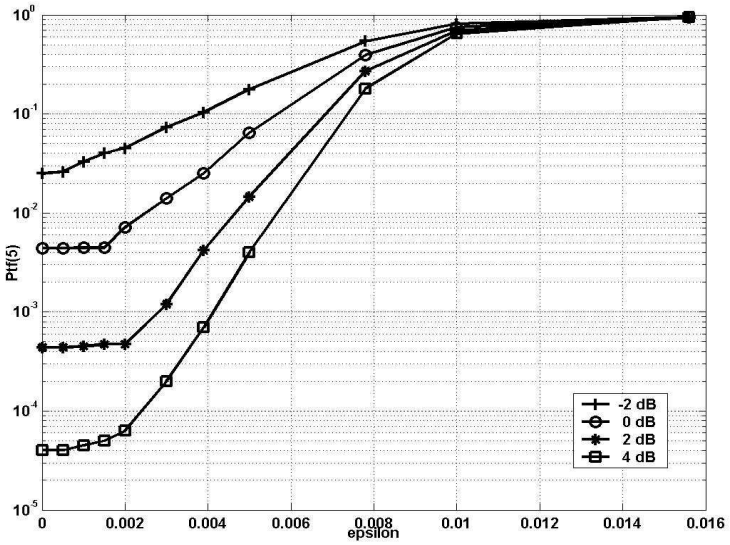


Figure 36: Timing failure probability resulted in time synchronizer without compensating CFO with respect to $\varepsilon$. Various SNRs are considered

Figure 36 represents the probability of time failure for a 4×4 MIMO system as a function of ε when the system lacks frequency estimation and compensation. In this system the arrival packet time instant is estimated by following equation:

$$\hat{\varepsilon} = \arg\max_{\varepsilon} \Psi(k) \tag{162}$$

where $\Psi(k) = \sum_{m=1}^{N_r} \mathbf{r}_{m,k}^H \mathbf{C_0} (\mathbf{C_0}^H \mathbf{C_0})^{-1} \mathbf{C_0}^H \mathbf{r}_{m,k}$ (as in (154) with ε = 0). The breaking point in Figure 36 denotes the maximum residual CFO that has no effect on time synchronization algorithm, and this is done for each SNR. For example, for SNR ≥ 0dB, the residual $\varepsilon$ can be up to 0.002 without significant loss in timing estimation. Thus, the resolution of our frequency estimator should be at least about 0.002 corresponding to a 512-point FFT. Therefore, the arrival packet can be estimated by using only a 512-point FFT to compensate CFO without the loss of performance. When the arrival packet and channel coefficient and initial CFO are estimated, we can then use the following method to improve CFO estimation. In the following frequency synchronization algorithms, therefore, we suppose that the frame synchronization has been performed perfectly, i.e. $\hat{k} = 0$.

***A) Residual frequency estimation --- method A***

The goal is to maximize the criteria presented in equation (160). To this aim we set its derivative with respective to CFO to zero that is:

$$\frac{d\Psi'(0,\varepsilon)}{d\varepsilon} = 0 \tag{163}$$

And we obtain the following equation:

$$\mathrm{Image}\left\{ \left( s \sum_{s=1}^{N-1} r_s' e^{-j2\pi\varepsilon s} \right) \right\} = 0 \tag{164}$$

And so:

$$\sum_{s=1}^{N-1} s \left| r' \right| \sin(\arg(r_s') - 2\pi\varepsilon s) = 0 \tag{165}$$

If we were to use the approximation $\arg(r_s') \approx 2\pi\varepsilon s$, the problem could be solved very easily. However it would be better if we find a better estimation for $\arg(r_s')$. Since we know that

$r'_s = \sum_{m=1}^{N_r} \sum_{p=0}^{N-1-s} [\phi]_{p,p+s} r^*_{m,p} r_{m,p+s}$ and $r_{m,p} = e^{j2\pi\varepsilon p} \left[\mathbf{C}_0^H\right]_{p,:} \mathbf{h}_{:m}^H + noise$, where $\left[\mathbf{C}_0^H\right]_{p,:}$ is $p^{th}$ row of matrix $C_0$ (eq.(149)), one can infer that $\arg(r'_s) = 2\pi\varepsilon s + \beta_s + noise$ where $\beta_s$ is the angle of $\sum_{m=1}^{N_r} \sum_{p=0}^{N-1-s} [\phi]_{p,p+s} (\left[\mathbf{C}_0^H\right]_{p,:} \mathbf{h}_{:m}^H)^* (\left[\mathbf{C}_0^H\right]_{p+s,:} \mathbf{h}_{:m}^H)$. By calculating $\beta_s$ we can rewrite (165) as:

$$\sum_{s=1}^{N-1} s |r'_s| \sin(\underbrace{\arg(r'_s) - \beta_s - 2\pi\varepsilon s}_{very\ small} + \beta_s) = 0 \tag{166}$$

And by using the approximation $\sin(x) \approx x$ we can find CFO as:

$$\hat{\varepsilon} = \frac{1}{2\pi} \frac{\sum_{s=1}^{N-1} s |r'_s| \left(\cos(\beta_s)(\arg(r'_s) - \beta_s) + \sin(\beta_s)\right)}{\sum_{s=1}^{N_p - 1} s^2 |r'_s|}_{s} \tag{167}$$

Therefore one can use the above equation to improve the frequency estimation. Equation (167) is a bit complex, however by using a simple method we can achieve to the same performance as equation (167). We will discuss about this new method in the following section (method B). Moreover, it is worth consideration that the method B has exactly the same performance as method A, while the method B is less complex.

***B) Residual frequency estimation --- method B***

The classical method to estimate the carrier frequency offset is to use cross correlation between received signal and noiseless received vector but in this method channel knowledge is required. Let us suppose that we have channel knowledge, CFO estimation can be derived as follows:

$$\begin{aligned} \hat{\varepsilon} &= \arg\max_{\varepsilon} \left| \sum_{m=1}^{N_r} (\mathbf{E}_0 \mathbf{C}_0 \mathbf{h}_{:m})^H \mathbf{r}_{m,0} \right|^2 \\ &= \arg\max_{\varepsilon} \left| \sum_{m=1}^{N_r} \mathbf{h}_{:m}^H \mathbf{C}_0^H \mathbf{E}_0^H \mathbf{r}_{m,0} \right|^2 \\ &= \arg\max_{\varepsilon} \left| \sum_{m=1}^{N_r} \sum_{s=0}^{N-1} \mathbf{h}_{:m}^H \left[\mathbf{C}_0^H\right]_{s,:} \mathbf{r}_{m,0}(s) e^{-j2\pi\varepsilon s} \right|^2 \end{aligned} \tag{168}$$

where $\left[\mathbf{C}_0^H\right]_{s,:}$ is $s^{\text{th}}$ row of matrix $\mathbf{C}_0$ (eq. (144)) and $\mathbf{r}_{m,0}(s)$ denotes the $s^{\text{th}}$ element of vector $\mathbf{r}_{m,0}$. If we define $\Delta(s) = \sum_{m=1}^{N_r} \mathbf{h}_{:m}^H \left[\mathbf{C}_0^H\right]_{s,:} \mathbf{r}_{m,0}(s)$ we have:

$$\hat{\varepsilon} = \arg\max_{\varepsilon} \left| \sum_{s=0}^{N-1} \Delta(s) e^{-j2\pi\varepsilon s} \right|^2 = \arg\max_{\varepsilon} \Re\left( \sum_{s=1}^{N-1} r''(s) e^{-j2\pi\varepsilon s} \right) \quad (169)$$

where $r''(s) = \sum_{p=0}^{N-1-s} \Delta(p)\Delta^*(p+s)$. Setting the derivative of (169) with respect to $\varepsilon$ equal to zero, we have:

$$\Im m\left( \sum_{s=1}^{N-1} s r''(s) e^{-j2\pi\hat{\varepsilon} s} \right) = 0 \quad (170)$$

Or equivalently:

$$\sum_{s=1}^{N-1} s \left| r''(s) \right| \sin\left( \arg(r''(s)) - 2\pi\hat{\varepsilon} s \right) = 0 \quad (171)$$

*Lemma 1*: $\arg(r''(s)) \approx j2\pi\varepsilon s$

*Proof:* We have defined $\Delta(s) = \sum_{m=1}^{N_r} \mathbf{h}_{:m}^H \mathbf{C}_0^H(s,:) \mathbf{r}_{m,0}(s)$

We know from equation (138) $\mathbf{r}_{m,k} = \sum_{n=1}^{N_t} \mathbf{E}_k \mathbf{C}_{n,k} \mathbf{h}_{n,m} + \mathbf{w}_{m,k}$ and so: $\mathbf{r}_{m,0}(s) = e^{j2\pi(s-1)\varepsilon} \mathbf{C}_0(s,:)\mathbf{h}_{:m} + \mathbf{w}_{m,k}(s)$

Therefore: $\Delta(s) = \sum_{m=1}^{N_r} \mathbf{h}_{:m}^H \mathbf{C}_0^H(s,:) \left( e^{j2\pi(s-1)\varepsilon} \mathbf{C}_0(s,:)\mathbf{h}_{:m} + \mathbf{w}_{m,k}(s) \right) = e^{j2\pi(s-1)\varepsilon} \sum_{m=1}^{N_r} \left\| \mathbf{h}_{:m} \mathbf{C}_0(s,:) \right\|^2 + noise$

and

$$r''(s) = \sum_{p=0}^{N-1-s} \Delta^*(p)\Delta(p+s) = e^{+j2\pi s\varepsilon} \sum_{p=0}^{N-1-s} \sum_{m=1}^{N_r} \sum_{m'=1}^{N_r} \left\| \mathbf{h}_{:m} \mathbf{C}_0(p,:) \right\|^2 \left\| \mathbf{h}_{:m'} \mathbf{C}_0(p+s,:) \right\|^2 + noise$$

So, we can conclude that $\arg(r''(s)) \approx j2\pi\hat{\varepsilon} s$ . The more the power of noise is lower or the more $N$ is greater, the more this approximation is accurate.

Here, in contrast with *method A*, based on lemma 1, if the training sequence is sufficiently long we have: $\arg(r''(s)) \approx j2\pi\hat{\varepsilon} s$. Therefore, using the approximation $\sin(x) \approx x$, $\varepsilon$ can be calculated as follows:

$$\hat{\varepsilon} = \frac{1}{2\pi} \frac{\sum_{s=1}^{N-1} s \left| r''(s) \right| \arg(r''(s))}{\sum_{s=1}^{N_p-1} s^2 \left| r''(s) \right|} \tag{172}$$

We have to note that due to the presence of function arg(.) in (172), $\hat{\varepsilon}$ is ambiguous when $\left| \arg(r''(s)) \right|$ exceeds π. This limits $\hat{\varepsilon}$ to be less than $1/2N_p$. For $N_p$=64, $\hat{\varepsilon}$ have to be less than 1/128, therefore, if in the coarse frequency estimation we use an FFT with length 512, the residual carrier frequency offset would be less than 1/128.

So our algorithm can be expressed as follows: in the first stage we estimate arrival packet instant, the channel coefficients, and initial CFO (as described in method A.). After compensating initial CFO by using channel coefficients estimated in the first stage, the residual CFO is estimated by (172). Having estimated residual CFO, the channel estimation is updated by (151), and by using this new channel coefficients, residual CFO estimate is updated by (172). This procedure is repeated in an iterative manner.

# 5.4 Simulation results and complexity

## 5.4.1 Cramer-Rao Lower Bound

Stoica in [123] based on [122] has derived the Cramer-Rao lower bound for frequency estimation in a SISO system. Following the same approach, we develop the Cramer-Rao lower bound for carrier frequency offset in the MIMO systems:

Considering equation (142), the received signal **r** is a complex valued circularly symmetric Gaussian vector with mean $\mathbf{M} = (\mathbf{I}_{N_r} \otimes \mathbf{E}_0 \mathbf{C}_k)\mathbf{h}_0$ (since we assume a perfect time synchronization we can replace *k* by 0) and covariance matrix $\sigma_w^2 \mathbf{I}$. The parameter vector of interest is $\mathbf{\Omega} = \left[ \varepsilon \;\; \mathbf{h}_R \;\; \mathbf{h}_I \right]^T$ where $\mathbf{h}_R$ and $\mathbf{h}_I$ stand for real and imaginary part of $\mathbf{h}_0$. For this type of problem the estimation of $\sigma_w^2$ is decoupled from that of $\mathbf{\Omega}$ [144][123].

The Ficher Information Matrix (FIM) for $\mathbf{\Omega}$ is given by [144][123]:

$$\mathbf{F} = \frac{2}{\sigma_w^2}\mathrm{Re}\left[\frac{\partial \mathbf{M}^H}{\partial \mathbf{\Omega}}\frac{\partial \mathbf{M}}{\partial \mathbf{\Omega}^T}\right] \tag{173}$$

Besides, we have:

$$\begin{aligned}
\frac{\partial \mathbf{M}}{\partial \varepsilon} &= j2\pi(\mathbf{I}_{N_r} \otimes \mathbf{D}\mathbf{E}_0\mathbf{C}_0)\mathbf{h}_0 \\
\frac{\partial \mathbf{M}^H}{\partial \mathbf{h}_R} &= (\mathbf{I}_{N_r} \otimes \mathbf{C}_0^H\mathbf{E}_0^H) \\
\frac{\partial \mathbf{M}^H}{\partial \mathbf{h}_I} &= -j(\mathbf{I}_{N_r} \otimes \mathbf{C}_0^H\mathbf{E}_0^H)
\end{aligned} \tag{174}$$

where $\mathbf{D} = diag(0,1...,N-1)$. Using of (174) in (173) as well as using the well-known properties of the Kronecker product yields:

$$\mathbf{F} = \frac{2}{\sigma_w^2}\begin{bmatrix}
4\pi^2\mathbf{h}_0^H(\mathbf{I}_{N_r} \otimes \mathbf{C}_0^H\mathbf{D}^2\mathbf{C}_0)\mathbf{h}_0 & 2\pi\mathrm{Im}\left[\mathbf{h}_0^H(\mathbf{I}_{N_r} \otimes \mathbf{C}_0^H\mathbf{D}\mathbf{C}_0)\right] & 2\pi\mathrm{Re}\left[\mathbf{h}_0^H(\mathbf{I}_{N_r} \otimes \mathbf{C}_0^H\mathbf{D}\mathbf{C}_0)\right] \\
-2\pi\mathrm{Im}\left[(\mathbf{I}_{N_r} \otimes \mathbf{C}_0^H\mathbf{D}\mathbf{C}_0)\mathbf{h}_0\right] & \mathrm{Re}\left[(\mathbf{I}_{N_r} \otimes \mathbf{C}_0^H\mathbf{C}_0)\right] & -\mathrm{Im}\left[(\mathbf{I}_{N_r} \otimes \mathbf{C}_0^H\mathbf{C}_0)\right] \\
2\pi\mathrm{Re}\left[(\mathbf{I}_{N_r} \otimes \mathbf{C}_0^H\mathbf{D}\mathbf{C}_0)\mathbf{h}_0\right] & \mathrm{Im}\left[(\mathbf{I}_{N_r} \otimes \mathbf{C}_0^H\mathbf{C}_0)\right] & \mathrm{Re}\left[(\mathbf{I}_{N_r} \otimes \mathbf{C}_0^H\mathbf{C}_0)\right]
\end{bmatrix} \tag{175}$$

The Cramer-Rao lower bound is obtained as the inverse of the FIM. For CRB of frequency offset we only need to calculate the first element of the FIM's inverse matrix.

In [123] stoica through a lemma ([123]-App. A) shows that if we have $\mathbf{F} = \frac{2}{\sigma_w^2}\begin{bmatrix} F_1 & \alpha^H \\ \alpha & F_h \end{bmatrix}$ the first element of $\mathbf{F}^{-1}$ will be:

$$\mathbf{F}^{-1}{}_{11} = \frac{\sigma_w^2}{2}(F_1 - \mathbf{\alpha}^T\mathbf{F}_h^{-1}\mathbf{\alpha})^{-1} \tag{176}$$

Here we have:

$F_1 = 4\pi^2\mathbf{h}_0^H(\mathbf{I}_{N_r} \otimes \mathbf{C}_0^H\mathbf{D}^2\mathbf{C}_0)\mathbf{h}_0$, $\mathbf{\alpha} = \begin{bmatrix} -2\pi\mathrm{Im}\left[(\mathbf{I}_{N_r} \otimes \mathbf{C}_0^H\mathbf{D}\mathbf{C}_0)\mathbf{h}_0\right] \\ 2\pi\mathrm{Re}\left[(\mathbf{I}_{N_r} \otimes \mathbf{C}_0^H\mathbf{D}\mathbf{C}_0)\mathbf{h}_0\right] \end{bmatrix}$ and

$$\mathbf{F}_h = \begin{bmatrix} \mathrm{Re}\left[(\mathbf{I}_{N_r} \otimes \mathbf{C}_0^H \mathbf{C}_0)\right] & -\mathrm{Im}\left[(\mathbf{I}_{N_r} \otimes \mathbf{C}_0^H \mathbf{C}_0)\right] \\ \mathrm{Im}\left[(\mathbf{I}_{N_r} \otimes \mathbf{C}_0^H \mathbf{C}_0)\right] & \mathrm{Re}\left[(\mathbf{I}_{N_r} \otimes \mathbf{C}_0^H \mathbf{C}_0)\right] \end{bmatrix}$$

It can be proved that $\mathbf{F}_h^{-1} = \begin{bmatrix} \mathrm{Re}\left[(\mathbf{I}_{N_r} \otimes \mathbf{C}_0^H \mathbf{C}_0)^{-1}\right] & -\mathrm{Im}\left[(\mathbf{I}_{N_r} \otimes \mathbf{C}_0^H \mathbf{C}_0)^{-1}\right] \\ \mathrm{Im}\left[(\mathbf{I}_{N_r} \otimes \mathbf{C}_0^H \mathbf{C}_0)^{-1}\right] & \mathrm{Re}\left[(\mathbf{I}_{N_r} \otimes \mathbf{C}_0^H \mathbf{C}_0)^{-1}\right] \end{bmatrix}$

Moreover, for any hermitian matrix $\mathbf{O}$ (e.g. $(I_{N_r} \otimes \mathbf{C}_0^H \mathbf{C}_0)^{-1}$) and any vector $\boldsymbol{x} = \boldsymbol{x}_R + \boldsymbol{x}_I$ (R and I stand for real and imaginary part) it holds that $\mathbf{x}^H \mathbf{O} \mathbf{x} = \begin{bmatrix} \mathbf{x}_R^T & \mathbf{x}_I^T \end{bmatrix} \begin{bmatrix} \mathbf{C}_R & -\mathbf{C}_I \\ \mathbf{C}_I & \mathbf{C}_R \end{bmatrix} \begin{bmatrix} \mathbf{x}_R \\ \mathbf{x}_I \end{bmatrix}$

Therefore, the CRB for frequency offset which is equal to $\mathbf{F}^{-1}_{11}$ is:

$$\begin{aligned} CRB(\varepsilon) &= \frac{\sigma_w^2}{2}\left(4\pi^2 \mathbf{h}_0^H (\mathbf{I}_{N_r} \otimes \mathbf{C}_0^H \mathbf{D}^2 \mathbf{C}_0)\mathbf{h}_0 - 4\pi^2 \mathbf{h}_0^H (\mathbf{I}_{N_r} \otimes \mathbf{C}_0^H \mathbf{D}\mathbf{C}_0)(\mathbf{I}_{N_r} \otimes \mathbf{C}_0^H \mathbf{C}_0)^{-1}(\mathbf{I}_{N_r} \otimes \mathbf{C}_0^H \mathbf{D}\mathbf{C}_0)\mathbf{h}_0\right)^{-1} \\ &= \frac{\sigma_w^2}{8\pi^2}\left(\mathbf{h}_0^H (\mathbf{I}_{N_r} \otimes \mathbf{C}_0^H \mathbf{D}^2 \mathbf{C}_0)\mathbf{h}_0 - \mathbf{h}_0^H (\mathbf{I}_{N_r} \otimes \mathbf{C}_0^H \mathbf{D}\mathbf{C}_0 \left(\mathbf{C}_0^H \mathbf{C}_0\right)^{-1} \mathbf{C}_0^H \mathbf{D}\mathbf{C}_0)\mathbf{h}_0\right)^{-1} \\ &= \frac{\sigma_w^2}{8\pi^2}\left[\mathbf{h}_0{}^H \left(\mathbf{I}_{N_r} \otimes \mathbf{C}_0^H \mathbf{D}\left(\mathbf{I} - \mathbf{C}_0 (\mathbf{C}_0^H \mathbf{C}_0)^{-1}\right)\mathbf{D}\mathbf{C}_0\right)\ \mathbf{h}_0\right]^{-1} \end{aligned} \quad (177)$$

Equation (177) can be written as:

$$CRB(\varepsilon) = \frac{\sigma_w^2}{8\pi^2}\left[\sum_{m=1}^{N_r} \mathbf{h}_{:\mathbf{m}} \mathbf{C}_\mathbf{0}^H \mathbf{D}\Theta \mathbf{C}_\mathbf{0} \mathbf{D}\mathbf{h}_{:\mathbf{m}}\right]^{-1} \quad (178)$$

where:

$$\Theta = I_N - \mathbf{C}_\mathbf{0}(\mathbf{C}_\mathbf{0}^H \mathbf{C}_\mathbf{0})^{-1} \mathbf{C}_\mathbf{0}^H \quad (179)$$

## 5.4.2 Algorithm complexity

To evaluate the complexity of the algorithm, we calculate the number of complex multiplications which are needed to implement the algorithm. The algorithm is composed of two stages. At the first stage initial frequency and timing offset and channel coefficients are calculated. The second stage deals with the estimation of residual frequency error.

For frequency estimation using FFT algorithm, the needed number of complex multiplications is equal to $\mathrm{O}\left((N^2-N)N_r+2R\log R\right)$, where $R$ is the length of FFT. For the estimation of the packet arrival instant the number of needed multiplications would be $\mathrm{O}\left((N^2+3N)N_rD\right)$ where $D$ is the interval width in which we search for the arrival of packet. Although it seems that the complexity of timing synchronization is rather high, but we have to note that its complexity is at the same order as the complexity of classical approach in time synchronization. The complexity of residual frequency estimation is equal to $\mathrm{O}\left(\frac{(N^2+N)}{2}+NN_rN_tL\right)$ in terms of the number of multiplications. Our algorithm is a bit complex. But this complexity is the price to pay to achieve high performance.

## 5.4.3 Simulation results

In practice, in a Rayleigh multipath fading environment, the channel may contain some small taps at the beginning, so the starting position of the channel is not clear. Taking this into account and in order to evaluate the performance of our ML time synchronization algorithm, we introduce the criterion to reveal the probability that how far our time synchronizer calculated instant is from the exact instant of packet arrival. Timing failure ($t_f$) probability is introduced similar to [25] as an indicator to illustrate the robustness of our algorithm. $P_{tf}(p)$ is the probability that the frame synchronizer misses an interval of $2p+1$ samples centered at $k=0$. This probability is expressed as follows:

$$P_{tf}(p)=\Pr\left\{\left|\hat{k}\right|>p\right\} \tag{180}$$

For the simulation setup we use a MIMO-OFDM system with QPSK modulation, the length of an OFDM word is 64 symbols and there is a cyclic prefix for each OFDM word with length 8. Exponentially decaying channel power delay profile of length 8 is used, decaying 3 dB per tap in positive time direction. The channel is fixed during transmission of one packet and independent of those of other packets. It is assumed that the average total TX power $P$ is distributed among the TX antennas such that $\sigma_u^2 = P/N_t$. The SNR per receive antenna is $P/\sigma_w^2=N_t\sigma_u^2/\sigma_w^2$.

As preamble, we use a single sequence with the length of an OFDM word ($N_P$=64) and a cyclic prefix; preambles are generated randomly for each transmit antenna. Our observation vector length $N$ is 64. The simulated burst consists of a preamble plus cyclic prefix, surrounded by two random 64-carrier OFDM symbols ($N_c$=64) plus cyclic prefix. Hence, the synchronizer have to search for the frame beginning at an interval of $N_c+2N_{cp}$=80 samples to the left and $N_c+N_{cp}$=72 samples to the right of the correct frame synchronization instant at $k$=0. Each point of result is obtained by averaging over $10^5$ Monte-Carlo runs.

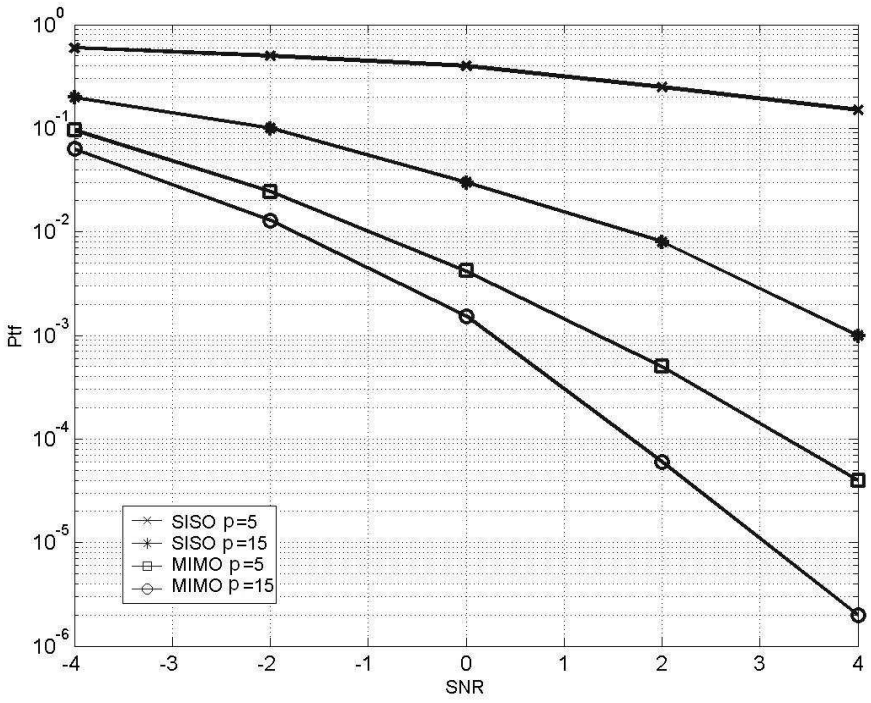


Figure 37: simulation results for the probabilities of missing a $\pm p$ interval

Figure 37 presents our simulation results for a $4 \times 4$ MIMO together with the reported results in [25]. The probability to miss a ±5 and a ±15 interval is plotted in the figure. It is clear from the figures that the performance of the proposed algorithm is better than the SISO case. Simulation results of lock-in probabilities are given in Figure 38 for various SNR. The asymmetry of the

figure is due to the channel length, that is, because Lock-in probability is represented mathematically as follows:

$$P_{lock-in} = \Pr\left\{\hat{k} = 0\right\} \tag{181}$$

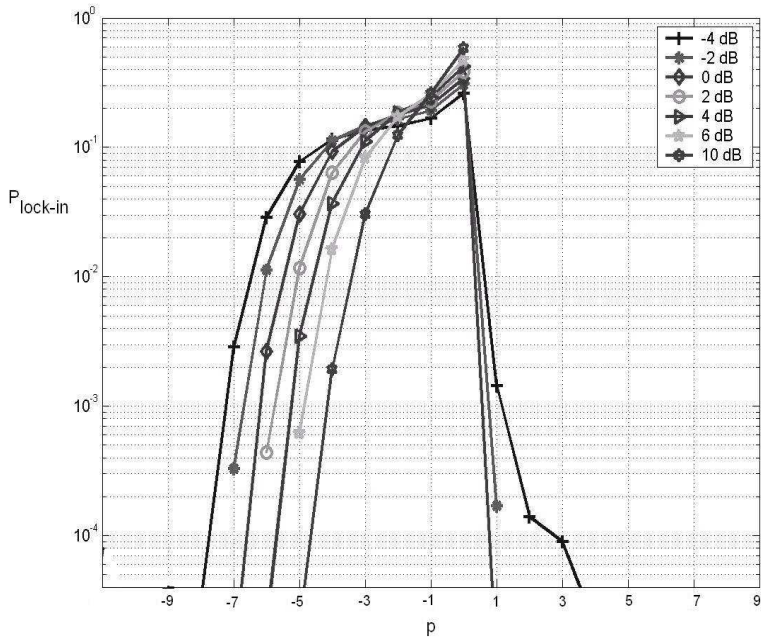


Figure 38: Simulation results for the lock-in probabilities in a dispersive channel. The simulation are performed for various SNR

Figure 39 compares the performance of frequency estimation for this method with the different method presented in chapter 3.2 together with Cramer-Rao Lower bound. The simulation setup used for different methods is as follows:

To be able to compare our results with the presented results in the literature we use a 2×2 MIMO-OFDM system to illustrate the results of frequency synchronization. The other specifications are the same as before. The carrier frequency offset ($\varepsilon$) is set to 0.1.

For FFT method (3.2.1), we used the same preamble as the preamble used by [118] (It was proposed by [62]). For hopping-pilots method (section 3.2.2), in each OFDM transmitted block we insert one pilot plus one zero symbol serving as a null subcarrier, and 64 OFDM blocks are used for CFO estimation. By such a preamble, we are able to compare this method with the

others. For autocorrelation method (section 3.2.4), we use two same consecutive preambles with length 32, each are made by a shift orthogonal code (Figure 18).

For FFT method a 1024 FFT are used. For our method that is the method presented in chapter (so called iterative method) for initial estimation a 512 FFT is used. In hopping pilots the resolution is set to .001.

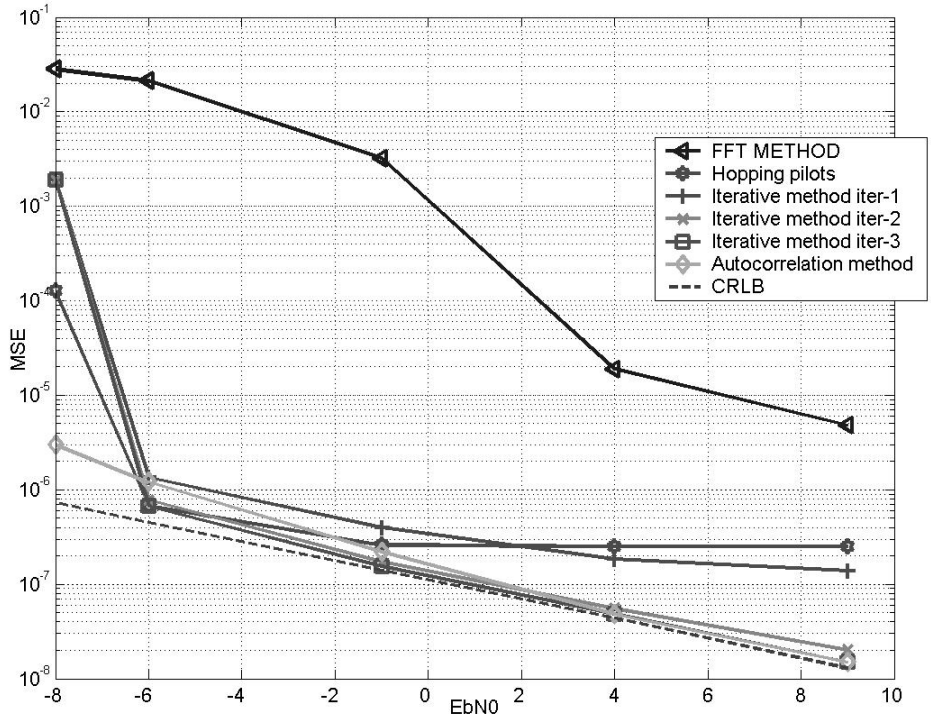


Figure 39: Mean-Square-Error of Carrier Frequency Offset (CFO) for four presented methods together with Cramer-Rao-Lower-Bound (CLRB)

All the diagrams have plotted in terms of Eb/N0. One can write Eb/N0 in terms of SNR as follows:

$$SNR=E_b/N_0+10\times\log(Rate)-10\times\log(.5) \quad (182)$$

where Rate is the modulation rate, for example we use QPSK modulation so Rate=2.

Note that we have the floor effect of MSE at high SNR for FFT method and Hopping pilots method. This effect, in FFT method, is due to the limited frequency resolution of FFT, which is

determined by the size of FFT, and in the Hopping pilots method is due to resolution of our search. The next three curves residual CFO is estimated in an iterative manner as presented in section 5.3.2. The last curve, which is the mean of Cramer-Rao lower bound for $\varepsilon$ which is estimated in (178)

Based on Figure 39, excluding FFT method, the three other methods have approximately same result (Apart from consideration of the floor effect of hopping pilots method). So for comparison between these algorithms we will need other criteria than the Minimum Square Error.

In addition, Figure 15 showed us that a CFO less than $10^{-4}$ assures that we have no degradation due to frequency error in a 2x2 MIMO OFDM systems. On the other hand by considering CRB curve in Figure 39, one can find that for a large range of $E_b/N_0$, the residual CFO is more than $10^{-4}$. Therefore, we simulated the iterative frequency synchronizer with various training sequence length. The result of this simulation (Figure 40) shows us that either training sequence length must be more than 64 or we have to accumulate the result of at least two packets to estimate CFO.

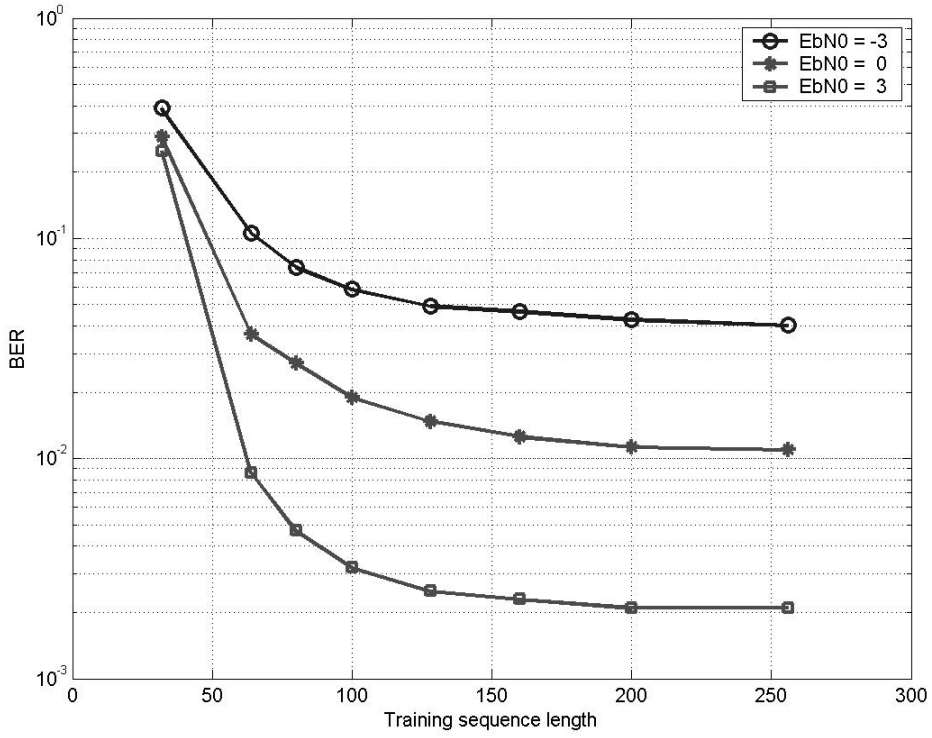


Figure 40: BER of an iterative synchronizer with respect to various training sequence length.

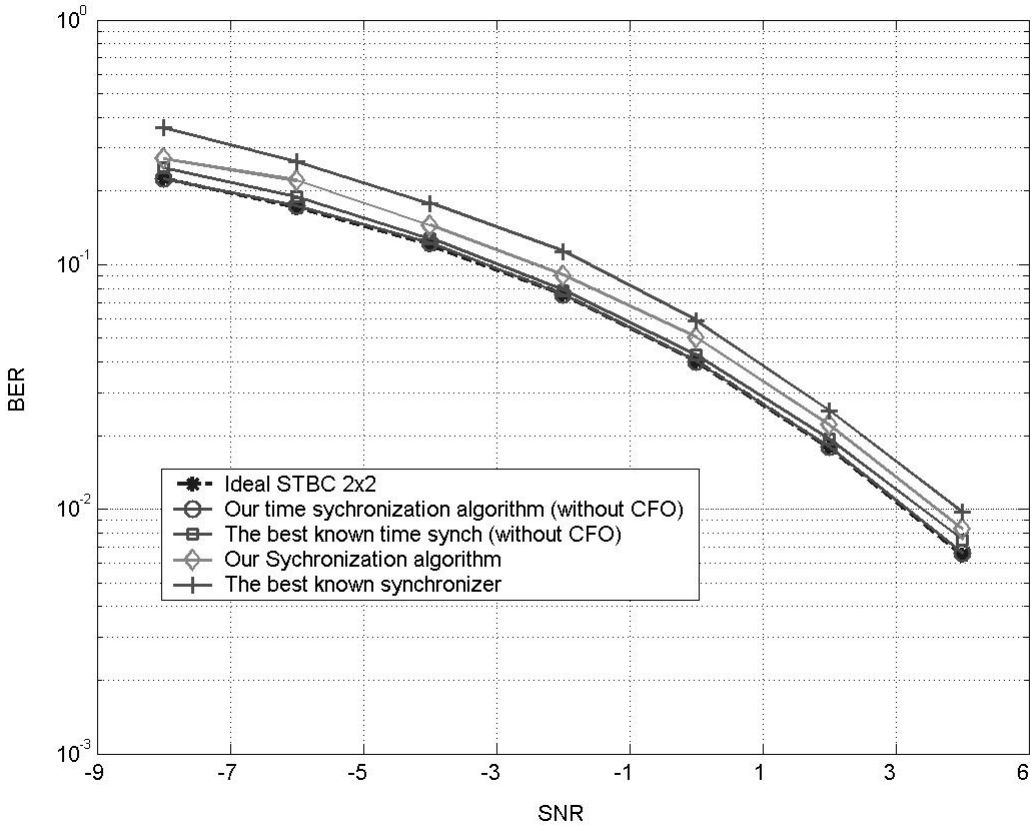


Figure 41: Comparison of our synchronizer and the best known synchronizer for a 2×2 MIMO-OFDM system

Figure 41 represents the comparison of our ML-MIMO-OFDM synchronizer with the best synchronizer presented in literature. To the best of our knowledge, one of the best synchronizer presented in literature can be achieved by using the autocorrelation method as both frame synchronizer and frequency synchronizer (we have discussed about them in section 4.3.1 section 3.2.4, respectively) and by using the algorithm presented in 4.4.1.4 for symbol timing estimation.

In Figure 41 the dashed line is the bit error rate for perfect synchronized 2×2 MIMO-OFDM system. We compare our algorithm with the known synchronizer in two different cases: the case of lacking carrier frequency offset and the case of existing frequency offset. In the case of

lacking CFO the best known algorithm is composed of autocorrelation method (section 4.3.1) plus fine synchronization method 4.4.1.4). In the case of the presence of CFO the frequency synchronization is performed with the method presented in section 4.3.1. As we can see from the figure, in both cases the performance of our algorithm outperforms the performance of the best known synchronizer however this is achieved at the price of higher complexity

*Chapter*


# EM-BASED MIMO-OFDM SYNCHRONIZATION

## 6.1 Introduction

All of the algorithms presented in the previous chapters use a training sequence to perform synchronization and thus can be categorized as data-aided algorithms. There are two problems with data-aided algorithms: first, there would be a loss in spectral efficiency and second, data-aided algorithms cannot be used in transmission mode and so they cannot be used for tracking the synchronization parameters. To avoid these two problems so-called code-aided algorithms can be used. As a matter of fact in these algorithms both frequency offset estimation and symbol timing estimation can be performed by using the data portion of the frame containing the information-bearing symbols. Recently, a great deal of papers has attempted to develop

code-aided (CA) estimation techniques. By using code-aided estimation one can use soft information provided from the decoding process, in order to fully exploit the code properties during estimation. Expectation-Maximization (EM) algorithm [124]-[128] is an efficient tool for iterative joint estimation and decoding. There is much literature [129]-[132] on using EM for channel estimation and decoding, whereas literature on code-aided synchronization is less [131][132] and in particular, to the best of our knowledge, there is no paper dealing with code-aided MIMO-OFDM synchronization. In this chapter we propose a code aided MIMO-OFDM synchronizer by which carrier frequency offset and symbol timing as well as channel coefficients can be estimated. This algorithm can be used to track the change of synchronization parameters as well as an estimation of residual parameters. This algorithm has been developed by us and is presented in [9] and submitted as a journal paper in [3].

## 6.2 System model

A MIMO-OFDM transmitter and receiver using EM synchronizer are shown in Figure 42. Let us consider such a MIMO-OFDM communication system with $N_t$ transmit and $N_r$ receive antennas.

Let the baseband equivalent signal at $n^{\text{th}}$ transmitter be $s_n(t)$, and $h_{nm}(t)$ be the equivalent channel between $n^{\text{th}}$ transmit antenna and $m^{\text{th}}$ receive antenna. The received signal $r_m(t)$ at $m^{\text{th}}$ receive antenna is sampled at $t=kT_s+\eta_0 T_s$ , where $T_s$ is the sampling time, and $\eta_0 \in [0,1)$ is the unknown time offset induced by the combination of the channel first path delay and the sampling phase offset. In OFDM systems $\eta_0$ can be absorbed in channel coefficient and so the sampled received signal can be expressed as:

$$r_{m,k} = \sum_{n=1}^{N_t} \sum_{l=0}^{L-1} s_n(kT_s - lT_s) h_{nm}(l) e^{j2\pi k\varepsilon} + w_{m,k} \qquad (183)$$

where $r_{m,k} \blacktriangleright\ r_m(kT_s)$, $w_{m,k} \blacktriangleright\ w_m(kT_s)$, $w_m(t)$ is the stationary additive Gaussian noise at $m^{\text{th}}$ receive antenna which is independent of the other antennas and $\varepsilon$ is the carrier frequency offset induced through the term $e^{j2\pi k\varepsilon}$. The carrier frequency offset is the multiplication of frequency error by sampling time that is $\varepsilon = \Delta f \times T_s$ where $\Delta f$ is the frequency error and $T_s$ is the sampling time. The fading multi-path channel $h_{nm}(l)$ is considered to be quasi static. Supposing the channel length equal to $L$, we have $N_t \times N_r$ paths, each of which can be modeled by an equivalent FIR complex filter of order $L$-1. These taps are assumed to be independent zero mean complex Gaussian random variables with variance $1/2P(l)$ per dimension.

Let the coded QPSK symbols of $n^{\text{th}}$ transmit antenna be $\mathbf{a}_n = \begin{bmatrix} a_{n,0} & a_{n,1} & \dots & a_{n,N_c-1} \end{bmatrix}$ where $N_c$ is the length of OFDM symbol. To convert coded symbols to an OFDM symbol we use the inverse Fourier transformation. Suppose that IFFT of $\begin{bmatrix} a_{n,0} & a_{n,1} & \dots & a_{n,N_c-1} \end{bmatrix}$ is $\begin{bmatrix} c_{n,0} & c_{n,1} & \dots & c_{n,N_c-1} \end{bmatrix}$. That is:

$$\begin{bmatrix} c_{n,0} & c_{n,1} & \dots & c_{n,N_c-1} \end{bmatrix} = \mathbf{F}^{-1} \begin{bmatrix} a_{n,0} & a_{n,1} & \dots & a_{n,N_c-1} \end{bmatrix} \tag{184}$$

**F** denotes the normalized DFT matrix with $(k,l)$ element equal to $1/\sqrt{N_c}\, e^{-j2\pi(k-1)(l-1)/N_c}$.

By using the same notation as the previous chapter we can rewrite (183) as:

$$\mathbf{r}_{m,k} = \sum_{n=1}^{N_t} \mathbf{E}_k \mathbf{C}_{n,k} \mathbf{h}_{nm} + \mathbf{w}_{m,k} \tag{185}$$

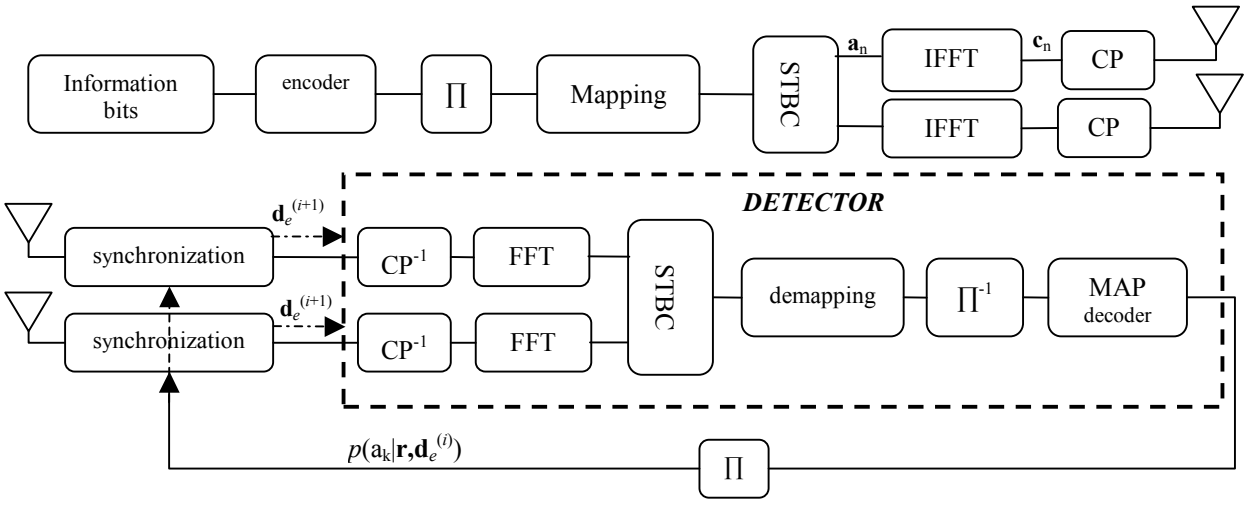

Figure 42: MIMO-OFDM iterative synchronization

Each STBC code word consists of ($P \times N_t$) STBC symbols, which are transmitted from $N_t$ transmitter antennas and across $P$ consecutive OFDM slots at a particular OFDM subcarrier (Figure 43). The STBC code words at different OFDM subcarriers are independently encoded, therefore, during OFDM slots, altogether STBC $N_c$ code words (or $N_c P N_t$ ) STBC code symbols) are transmitted. The first OFDM word is a preamble that is sent for initial synchronization.

According to Tarokh [20], the STBC is defined by a $P \times N_t$ code matrix $\mathbf{G}$, where $N_t$ denotes the number of transmitter and $P$ denotes the number of time slots for transmitting a STBC code word. Each row of $\mathbf{G}$ is a permuted and transformed (i.e., negated and/or conjugated) form of the $N_t$ dimensional vector of complex data symbols $\boldsymbol{x}$. As a simple example, we consider a 2×2 STBC (i.e. $N_t$=2, $P$=2). Its code matrix $\mathbf{G_1}$ is defined by

$$\mathbf{G}_1 = \begin{bmatrix} x_1 & x_2 \\ -x_2^* & x_1^* \end{bmatrix} \tag{186}$$

The input to this STBC is the data vector $\mathbf{x}=[x_1, x_2]^T$. During the first time slot, the two symbols in the first row $[x_1, x_2]$ of $\mathbf{G}_1$ are transmitted simultaneously from the two transmitter antennas; during the second time slot, the symbols in the second row $[-x_2^*, x_1^*]$ of $\mathbf{G}_1$ are transmitted.

In our STBC-OFDM system, we apply the above STBC encoder to data symbols transmitted at different subcarriers independently. For example, by using the STBC defined by $\mathbf{G}_1$, at the $k^{th}$ subcarrier, during the first OFDM slot, two symbols $c_{1,k}$ $c_{2,k}$ are transmitted simultaneously from two transmitter antennas; during the next OFDM symbols, $-c_{2,k}^*$ $c_{1,k}^*$ symbols are transmitted.

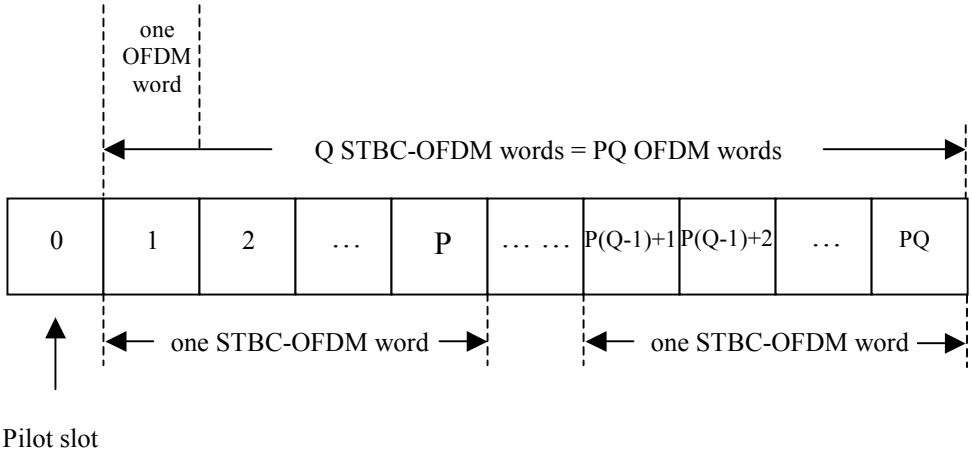


Figure 43: Data burst structure.

# 6.3 EM algorithm

Expectation-Maximization (EM) algorithm is an algorithm to estimate a parameter $\mathbf{d}_e$ based on an observation **r** in the case where the observation **r** depends also on another unknown parameter (which is called the nuisance parameter) and estimation could have been easily computed if we had access to this parameter. MAP and ML estimation are two possible techniques to compute estimates of a parameter. MAP estimation exploits a-priori information on the parameter. ML estimation is suited to problems where a-priori information is missing, or when the parameter is non-random (i.e. unknown but deterministic). Straightforward application of ML or MAP procedure is not possible in most estimation problems due to their complexity. The EM algorithm is a technique that solves the MAP or ML problem in an iterative way [124]-[128]. The EM algorithm at each iteration breaks down in two steps: the Expectation step (E-step) and the Maximization step (M-step). The algorithms starts with an initial estimate $\mathbf{d}_e^{(0)}$ of $\mathbf{d}_e$, E- and M-steps are performed successively and after each M-step a new estimate of $\mathbf{d}_e$ is produced. Therefore, a sequence of estimates $\mathbf{d}_e$: $[\mathbf{d}_e^{(0)}, \mathbf{d}_e^{(1)}, \mathbf{d}_e^{(2)}, \ldots]$ is obtained. When $\mathbf{d}_e$ is continuous, the EM algorithm will converge to a value $\mathbf{d}_e^{(\infty)}$.

**Complete data**

We refer to $\mathbf{d}_c$ as the *complete data*. In order to qualify as valid complete data, we have the following sufficient condition on $\mathbf{d}_c$:

$$p(\mathbf{d}_c,\mathbf{r}|\,\mathbf{d}_e) = p(\mathbf{r}|\,\mathbf{d}_c)\,p(\mathbf{d}_c\,|\,\mathbf{d}_e) \tag{187}$$

so that the observation can depend on $\mathbf{d}_e$ only through $\mathbf{d}_c$. In other way, we can rewrite (187) as:

$$p(\mathbf{d}_e\,|\mathbf{d}_c,\,\mathbf{r}) = p(\mathbf{d}_e\,|\mathbf{d}_c) \tag{188}$$

**Initial estimate:** An initial estimate $\mathbf{d}_e^{(0)}$ of $\mathbf{d}_e$ is required to start the EM algorithm.

**E-step:** At iteration $i \geq 0$, the E-step is given by:

$$\begin{aligned} Q\left(\mathbf{d}_e\left|\mathbf{d}_e^{(i)}\right.\right) &= E_{\mathbf{d}_c}\left\{\log p(\mathbf{d}_c,\mathbf{d}_e)\left|\mathbf{r},\mathbf{d}_e^{(i)}\right.\right\} \\ &= \log p(\mathbf{d}_e) + E_{\mathbf{d}_c}\left\{\log p(\mathbf{d}_c\left|\mathbf{d}_e\right.)\left|\mathbf{r},\mathbf{d}_e^{(i)}\right.\right\} \\ &= \log p(\mathbf{d}_e) + \int \log p(\mathbf{d}_c\left|\mathbf{d}_e\right.)\, p(\mathbf{d}_c\left|\mathbf{r},\mathbf{d}_e^{(i)}\right.)\, d\mathbf{d}_c \end{aligned} \tag{189}$$

**M-step**: Following the E-step, the M-step is given by:

$$\mathbf{d}_e^{(i+1)} = \arg\max_{\mathbf{d}_e} Q\left(\mathbf{d}_e\left|\mathbf{d}_e^{(i)}\right.\right) \tag{190}$$

**Missing data:** In many technical papers, $\mathbf{d}_c = [\mathbf{r},\, \mathbf{d}_m]$, where $\mathbf{d}_m$ is referred to as the *missing* (or unobserved) data. In that case, $\mathbf{d}_c$ is always a valid complete data. Often, $\mathbf{d}_m$ will be independent of $\mathbf{d}_e$, so that the E-step becomes:

$$\begin{aligned} Q\left(\mathbf{d}_e\left|\mathbf{d}_e^{(i)}\right.\right) &= \log p(\mathbf{d}_e) + \int \log p(\mathbf{r},\mathbf{d}_m\left|\mathbf{d}_e\right.)\, p(\mathbf{d}_m\left|\mathbf{r},\mathbf{d}_e^{(i)}\right.)\, d\mathbf{d}_m \\ &\propto \log p(\mathbf{d}_e) + \int \log p(\mathbf{r}\left|\mathbf{d}_m,\mathbf{d}_e\right.)\, p(\mathbf{d}_m\left|\mathbf{r},\mathbf{d}_e^{(i)}\right.)\, d\mathbf{d}_m \end{aligned} \tag{191}$$

**Discrete parameters:** When $\mathbf{d}_e$ is – or contains – a discrete parameter (finite value) there might be the convergence problems. For example, symbol timing involves estimating a discrete parameter. To avoid the convergence problems associated with (finite valued) discrete parameters one can use the following approach [131]:

Let $\mathbf{b} = \{\mathbf{b}_c, \mathbf{b}_d\}$, where $\mathbf{b}_c$ and $\mathbf{b}_d$ denote the discrete and continuous components of $\mathbf{b}$, respectively. Assume that $\mathbf{b}_d$ can only take on values in a finite set $\mathbf{B}_d$. We keep $\mathbf{b}_d$ fixed to some value $\tilde{\mathbf{b}}_d \in \mathbf{B}_d$ and iteratively update only $\mathbf{b}_c$:

$$\mathbf{b}_c^{(i+1)} = \arg\max_{\mathbf{b}_c} \left\{ Q\left(\tilde{\mathbf{b}}_d, \mathbf{b}_c \middle| \tilde{\mathbf{b}}_d, \mathbf{b}_c^{(i)}\right)\right\} \tag{192}$$

Let us denote by $\hat{\mathbf{b}}_c(\tilde{\mathbf{b}}_d)$ the estimate obtained after convergence of the above equation. The final estimate of $\mathbf{b}_d$ then becomes:

$$\hat{\mathbf{b}}_d = \arg\max_{\tilde{\mathbf{b}}_d \in B_d} \left\{ Q\left(\tilde{\mathbf{b}}_d, \hat{\mathbf{b}}_c(\tilde{\mathbf{b}}_d)\right)\right\} \tag{193}$$

While the final estimate of $\mathbf{b}_c$ is given by $\hat{\mathbf{b}}_c(\hat{\mathbf{b}}_d)$.

## 6.4 EM synchronizer

We assume that the initial synchronization has already been done (e.g. using data aided methods). It means that, the coarse frame synchronization has succeeded to detect a packet arrival but the precise beginning is not well determined and despite the fact that carrier frequency offset has been already estimated and compensated, there still exists a residual carrier frequency offset.

We take the observation vector $\mathbf{r}_{obs}$ with length $N_c$. Assume $k_0$ as time offset of observation vector with respect to beginning of the training sequence. The objective of our algorithm is to estimate $\mathbf{h}$, $k_0$ and CFO from the observation vector $\mathbf{r}_{obs}$. Since the transmitted signal is not known, a-priori information provided by MAP decoder is used to estimate the unknown parameters. Estimation will be performed in an iterative manner by using EM algorithm. It is worth mentioning the fact that there is a small residual correlation between the MAP encoder (extern code) and STBC code (inner code) which is ignored because of the interleaver (Figure 42).

We now make use of the EM algorithm to estimate $\mathbf{d}_e=[k_0, \mathbf{h}, \varepsilon]$. The coded data symbols can be considered as the nuisance parameters. Defining the complete data $\mathbf{d}_c=[\mathbf{r}_{obs}, \mathbf{c}]$, the E-step in EM algorithm can be written as:

$$Q\left(\mathbf{d}_e \middle| \mathbf{d}_e^{(i)}\right) = E_{\mathbf{c}}\left\{\log p(\mathbf{r}|\mathbf{d}_e) \middle| \mathbf{r}, \mathbf{d}_e^{(i)}\right\} \tag{194}$$

and

$$p(\mathbf{r}_{obs}|\varepsilon, k, \mathbf{h}) = \frac{1}{(\pi\sigma_w^{\,2})^N} \exp\left\{-\frac{\left\|\mathbf{r}_{obs} - (I_{N_r} \otimes \mathbf{E}_0\mathbf{C}_k)\mathbf{h}\right\|^2}{\sigma_w^{\,2}}\right\} \tag{195}$$

So we have:

$$\begin{aligned} Q\left(\mathbf{d}_e \middle| \mathbf{d}_e^{(i)}\right) &\propto -E_{\mathbf{c}}\left\{\left\|\mathbf{r}_{obs} - (I_{N_r} \otimes \mathbf{E}_0\mathbf{C}_k)\mathbf{h}\right\|^2 \middle| \mathbf{r}, \mathbf{d}_e^{(i)}\right\} \\ &\propto -E_{\mathbf{c}}\left\{\left\|(I_{N_r} \otimes \mathbf{E}_0\mathbf{C}_k)\mathbf{h}\right\|^2 + 2\mathrm{Re}\left(\mathbf{r}_{obs}^{\,H}(I_{N_r} \otimes \mathbf{E}_0\mathbf{C}_k)\mathbf{h}\right) \middle| \mathbf{r}, \mathbf{d}_e^{(i)}\right\} \\ &\propto -\mathbf{h}^H(I_{N_r} \otimes E_{\mathbf{C}_k^H\mathbf{C}_k})\mathbf{h} + 2\Re\mathrm{e}\left(\mathbf{r}_{obs}^{\,H}(I_{N_r} \otimes \mathbf{E}_0 E_{\mathbf{C}_k})\mathbf{h}\right) \end{aligned} \tag{196}$$

where $E_{\mathbf{C}_k^H\mathbf{C}_k} = E\left\{\mathbf{C}_k^{\,H}\mathbf{C}_k \middle| \mathbf{r}, \mathbf{d}_e^{(i)}\right\}$ and $E_{\mathbf{C}} = E\left\{\mathbf{C}_k \middle| \mathbf{r}, \mathbf{d}_e^{(i)}\right\}$.

To calculate the expectation of matrix **C** we use a-posteriori probabilities provided by MAP decoder. Interleaved a-posteriori outputs of the MAP decoder are equal to $p(a_k|\mathbf{r},\mathbf{d}_e^{(i)})$ and by (184) we know that $\left[c_{n,0} \quad c_{n,1} \quad \ldots \quad c_{n,N_c-1}\right] = \mathbf{F}^{-1}\left[a_{n,0} \quad a_{n,1} \quad \ldots \quad a_{n,N_c-1}\right]$. We denote the vector $\left[a_{n,0} \quad \ldots \quad a_{n,N_c-1}\right]$ by $\vec{\mathbf{a}}_n$.

Since $E\{f(\mathbf{x})\} = \sum_{\mathbf{x}} f(\mathbf{x})p(\mathbf{x})$, the expectation of matrix **C** would be calculated as:

$$E\left\{c_{n,k} \middle| \mathbf{r}, \mathbf{d}_e^{(i)}\right\} = E\left\{\mathbf{f}_k^{-1}\vec{\mathbf{a}}_n \middle| \mathbf{r}, \mathbf{d}_e^{(i)}\right\} = \sum_{\vec{\mathbf{a}}_n} \mathbf{f}_k^{-1}\vec{\mathbf{a}}_n \; p(\vec{\mathbf{a}}_n) = \mathbf{f}_k^{-1}.\left(E\left\{\vec{\mathbf{a}}_n \middle| \mathbf{r}, \mathbf{d}_e^{(i)}\right\}\right) \tag{197}$$

where $\mathbf{f}_p^{\,-1} = \frac{1}{\sqrt{N_c}}[e^{2\pi jpk/N_c}]^T \quad k = 0,\ldots,N_c-1$ is a row of IDFT matrix. The precise value of $E_{\mathbf{C}_k^H\mathbf{C}_k}$ is calculated in the appendix. However $E_{\mathbf{C}_k^H\mathbf{C}_k}$ is approximately equal to $\mathbf{I}_{LNt\times LNt}$. In order to maximize $Q$ we set the partial derivative of $Q$ (eq. (196)**)** with respect to **h** to zero, we obtain the estimate for $\mathbf{h}_0$ as:

$$\mathbf{h}^{(i+1)} = (I_{N_r} \otimes E_{\mathbf{C}_k^H\mathbf{C}_k})^{-1}(I_{N_r} \otimes \mathbf{E}_0 E_{\mathbf{C}_k})^H \mathbf{r}_{obs} \tag{198}$$

Intuitively speaking, the decoder does not work well in the first iterations and so $E_{\mathbf{C}_k}$ would be very small, whereas $E_{\mathbf{C}_k^H\mathbf{C}_k}$ is always equal to $\mathbf{I}_{LNt\times LNt}$ and thus the estimate of **h** would be close

to zero; hence we may infer that our EM channel estimator is biased. Wautelet cites the same phenomenon in [129]. By calculating the expectation of EM channel estimator one can analytically verify that the EM channel estimator is a biased estimator. That is, the expected value of channel taps by using (185) can be calculated as:

$$\begin{aligned} E(\hat{\mathbf{h}}^{(i+1)}) &= (I_{N_r} \otimes E_{\mathbf{C}_k^H \mathbf{C}_k})^{-1} (I_{N_r} \otimes \mathbf{E}_0 E_{\mathbf{C}_k})^H E(\mathbf{r}_{obs}) \\ &= (I_{N_r} \otimes E_{\mathbf{C}_k^H \mathbf{C}_k})^{-1} (I_{N_r} \otimes \mathbf{E}_0 E_{\mathbf{C}_k})^H (I_{N_r} \otimes \mathbf{E}_0 E_{\mathbf{C}_k}) \mathbf{h}_0 \\ &= (I_{N_r} \otimes E_{\mathbf{C}_k^H \mathbf{C}_k})^{-1} (I_{N_r} \otimes E_{\mathbf{C}_k})^H (I_{N_r} \otimes E_{\mathbf{C}_k}) \mathbf{h}_0 \end{aligned} \tag{199}$$

It can be seen from (199) that there is a bias factor equal to $(I_{N_r} \otimes E_{\mathbf{C}_k^H \mathbf{C}_k})^{-1} (I_{N_r} \otimes E_{\mathbf{C}_k})^H (I_{N_r} \otimes E_{\mathbf{C}_k})$. To solve this problem we divide the EM channel estimate by the biased factor and so the channel estimate will be:

$$\mathbf{h}^{(i+1)} = (I_{N_r} \otimes E_{\mathbf{C}_k}{}^H E_{\mathbf{C}_k})^{-1} (I_{N_r} \otimes \mathbf{E}_0 E_{\mathbf{C}_k})^H \mathbf{r}_{obs} \tag{200}$$

Substituting equation (200) into equation (196), the delay [$k_0$ , $\varepsilon$] can be estimated by maximizing the following likelihood function:

$$\begin{aligned} Q(k, \varepsilon \left| k^{(i)}, \varepsilon^{(i)} \right.) &\propto \mathbf{r}_{m,obs}^H (I_{N_r} \otimes \mathbf{E}_0 E_{\mathbf{C}_k} (E_{\mathbf{C}_k}{}^H E_{\mathbf{C}_k})^{-1} E_{\mathbf{C}_k}{}^H) \mathbf{r}_{obs} \\ &= \sum_{m=1}^{Nr} \mathbf{r}_{m,obs}^H \mathbf{E}_0 E_{\mathbf{C}_k} (E_{\mathbf{C}_k}{}^H E_{\mathbf{C}_k})^{-1} E_{\mathbf{C}_k}{}^H \mathbf{E}_0^H \mathbf{r}_{m,obs} \end{aligned} \tag{201}$$

Therefore:

$$\left[ k_0^{(i+1)}, \varepsilon^{(i+1)} \right] = \arg\max_{k,\varepsilon} \left\{ Q(k, \varepsilon \left| k^{(i)}, \varepsilon^{(i)} \right.) \right\} \tag{202}$$

From (200) and (202), it appears that the estimation of channel is decoupled from the estimate of time and frequency, that is, once one calculates the estimate of carrier frequency and timing offset one can estimate the channel coefficients. In contrast, timing offset and frequency seems not to be decoupled. However by using the method suggested in the previous chapter, we try to decouple the estimation of frequency offset from timing offset.

To estimate $\varepsilon$, let suppose that $k = \tilde{k} \in [-N_{CP} \quad 0]$. Now we can simplify $Q$ further as the previous chapter:

$$Q(\tilde{k},\varepsilon\left|\tilde{k},\varepsilon^{(i)}\right.)\propto\sum_{m=1}^{N_r}\mathbf{r}_{obs}^{H}\mathbf{E}_{\mathbf{0}}\underbrace{E_{\mathbf{C}_k}(E_{\mathbf{C}_k}{}^{H}E_{\mathbf{C}_k})^{-1}E_{\mathbf{C}_k}{}^{H}}_{\Phi}\mathbf{E}_{\mathbf{0}}^{H}\mathbf{r}_{obs}=\sum_{m=1}^{Nr}\sum_{p=0}^{Nc-1}r_{m,p}^{*}\sum_{q=0}^{Nc-1}[\phi]_{p,q}r_{m,q}e^{-j2\pi\varepsilon(q-p)} \quad (203)$$

Considering Hermitain symmetry of $\Phi$ we have:

$$Q(\tilde{k},\varepsilon\left|\tilde{k},\varepsilon^{(i)}\right.)\propto\sum_{m=1}^{N_r}\left\{\sum_{p=0}^{Nc-1}[\phi]_{p,p}\left|r_{m,p}\right|^2+\underbrace{2\mathrm{Re}\left(\sum_{p=0}^{Nc-2}\sum_{q=p+1}^{Nc-1}[\phi]_{p,q}r_{m,p+k}^{*}r_{m,q}e^{-j2\pi\varepsilon(q-p)}\right)}_{Q'(\tilde{k},\varepsilon\left|\tilde{k},\varepsilon^{(i)}\right.)}\right\} \quad (204)$$

$\mathrm{Re}(.)$ is the real part of a quantity. Note that the first term in (204) is independent of $\varepsilon$ and has no role in frequency estimation and thus to estimate $\varepsilon^{(i+1)}$ we have to maximize the second term. i.e. $\varepsilon^{(i+1)}=\arg\max_{\varepsilon}\left\{Q'(\tilde{k},\varepsilon\left|\tilde{k},\varepsilon^{(i)}\right.)\right\}$. By denoting $s = q - p$, we can further simplify the second term:

$$Q'(\tilde{k},\varepsilon\left|\tilde{k},\varepsilon^{(i)}\right.)=\mathrm{Re}\left(\left\{\sum_{s=1}^{N-1}r_s'e^{-j2\pi\varepsilon s}\right\}\right) \quad (205)$$

where $r_s'=\sum_{m=1}^{N_r}\sum_{p=0}^{N-1-s}[\phi]_{p,p+s}r_{m,p}^{*}r_{m,p+s}$ .

Using equation (205), $\varepsilon^{(i+1)}$ can be estimated by the FFT algorithm. A better frequency resolution can be achieved by zero-padding to effectively increase the FFT length. Moreover, one can use the algorithm of Chirp z-transform (CZT), to increase sufficiently the FFT length without considerable increase in complexity. Although using FFT (and CZT) algorithm significantly reduces the complexity of the estimator, in the case of low carrier frequency offsets, we propose a more reduced complexity algorithm to estimate $\varepsilon^{(i+1)}$. Setting the derivative of the equation (205) with respect to ε equal to zero, we have:

$$\mathrm{Im}\left(\sum_{s=1}^{N_c-1}sr_s'e^{-j2\pi\varepsilon^{(i+1)}s}\right)=0 \quad (206)$$

Or equivalently:

$$\sum_{s=1}^{N_c-1}s\left|r_s'\right|\sin\left(\arg(r_s')-2\pi\varepsilon^{(i+1)}s\right)=0 \quad (207)$$

Let's suppose $\arg(r'_s) \approx j2\pi\varepsilon^{(i+1)}s$ for the moment. For small values of frequency error this approximation can be supposed valid. However, if one is not satisfied with the approximation, he can use one of the two methods presented in chapter 5.3.2). Using the approximation $\sin(x) \approx x$, we have:

$$\sum_{s=1}^{Nc-1} s\left|r'_s\right| \arg(r'_s) = 2\pi\varepsilon^{(i+1)} \sum_{s=1}^{Nc-1} s^2 \left|r'_s\right| \tag{208}$$

$\varepsilon^{(i+1)}$ can be thus calculated as follow:

$$\varepsilon^{(i+1)} = \frac{1}{2\pi} \frac{\sum_{s=1}^{Nc-1} s\left|r'_s\right| \arg(r'_s)}{\sum_{s=1}^{Nc-1} s^2 \left|r'_s\right|} \tag{209}$$

Presence of the arg(.) in (209) makes $\hat{\varepsilon}$ ambiguous when $\left|\arg(r'_s)\right|$ exceed π. Therefore, $\varepsilon$ must to be less than $1/2N_c$.

In brief, we keep $k$ fixed to $\tilde{k} \in \left[-N_{CP} \quad 0\right]$ and we estimate $\hat{\varepsilon}^{(i+1)}$ from (209) and update it iteratively. An initial estimate is required to find the residual carrier frequency offset and in our case where we had supposed that carrier frequency offset is very small, we can set the initial value to zero. Let us denote by $\hat{\varepsilon}(\tilde{k})$ the final estimate obtained for CFO after convergence, the symbol timing can be estimated as follows:

$$\hat{k} = \underset{\tilde{k} \in \left[-N_{cp} \quad 0\right]}{\arg\max} \left\{ Q\left(\tilde{k}, \hat{\varepsilon}(\tilde{k})\right) \right\} \tag{210}$$

Note that no initial estimate is required for $k$. The final estimate of CFO is given by $\hat{\varepsilon}(\hat{k})$.

As previously mentioned, EM algorithm is a biased channel estimator. To solve this problem we proposed an unbiased version of EM algorithm. But still there is another solution for the bias problem: In fact, if we use our EM algorithm in a Decision-Directed (DD) mode we can avoid the bias problem. In the Decision-Directed (DD) mode the hard decisions on the symbols are used instead of their expected values. In other words, the equations remain unchanged, except that $E_{\mathbf{C}_k}$ will be replaced by $\breve{\mathbf{C}}_k$ which is filled with $\breve{c}_{n,p}$ according to the pattern presented in (144) and (139). $\breve{c}_{n,p}$ is the $p^{\text{th}}$ element of the inverse Fourier transformation of $\breve{\mathbf{a}}_n$ where $\breve{\mathbf{a}}_n = \left[\breve{a}_{n,0}, ..., \breve{a}_{n,N_c-1}\right]$ and $\breve{a}_{n,q}$ is the hard decision made by the MAP decoder on the $p^{\text{th}}$

symbol of the $n^{th}$ transmit antenna. Consequently, the EM estimator in DD mode is unbiased and there is no need for a bias factor.

## 6.5 Simulation results

In the simulation we used a 2×2 MIMO OFDM system with 256 subcarriers. The length of cyclic prefix is 32. As channel encoder, we used a (5,7) convolutional code. The log-MAP algorithm is used for decoding. The size of each STBC-OFDM word is equal to 2 OFDM words and each OFDM word is composed of 256 QPSK symbols. For sake of EM convergence and supposing that an initial coarse frequency synchronization has already been done, we set the maximum value of ε equal to 0.1 percent. Exponentially decaying channel power delay profile of length 8 is used, decaying 3 dB per tap in positive time direction. The channel is fixed during transmission of one packet and independent of those of other packets. It is assumed that the average total TX power $P$ is distributed among the TX antennas such that $\sigma_u^2 = P/Nt$. The SNR per receive antenna is $P/\sigma_w^2 = N_t\sigma_u^2/\sigma_w^2$ . For the initial synchronization we used a training sequence with length of 64 QPSK symbols.

Figure 44 represents the Bit-Error-Rate (BER) versus the $E_b/N_0$. The dashed-line curve corresponds to the case where synchronization is perfect. Figure 44 shows also BER for both case of unbiased code aided synchronization (soft synchronization) and Decision Directed (DD) synchronization. As previously mentioned, in the DD estimation we use the same algorithm but instead of $E_{\mathbf{C}_k}$ the hard decisions made by the decoder are used. The performance of the DD synchronizer and the unbiased code-aided synchronizer is very close to each other (about 0.5 dB far from perfect synchronization). However, in both cases DD synchronizer and unbiased code aided synchronizer a matrix inverse must be calculated in each iteration. In practice, we know that both $E_{\mathbf{C}_k}{}^H E_{\mathbf{C}_k}$ in case of unbiased code-aided synchronizer and $\breve{\mathbf{C}}_K^H \breve{\mathbf{C}}_K$ in case of using DD synchronizer are close to the unity matrix and so one can approximate them by the unity matrix. This leads us to two further algorithms which can be called the approximation of code aided and decision directed algorithms. In these algorithms a matrix inversion will be avoided in each iteration and thus a great amount of complexity will be reduced. We note that the

approximation of unbiased code-aided algorithm is the same as the approximation of biased code aided algorithm. Figure 44 shows that the performance of the approximation of DD and code-aided (CA) synchronization is very close to the performance of unbiased CA and DD algorithm. The first curve in the Figure 44 corresponds to the first iteration in the EM algorithm and shows the case where we have an uncompensated residual synchronization error. As it can be seen, even a very small amount of frequency offset can drastically degrade the overall performance of the MIMO-OFDM system.

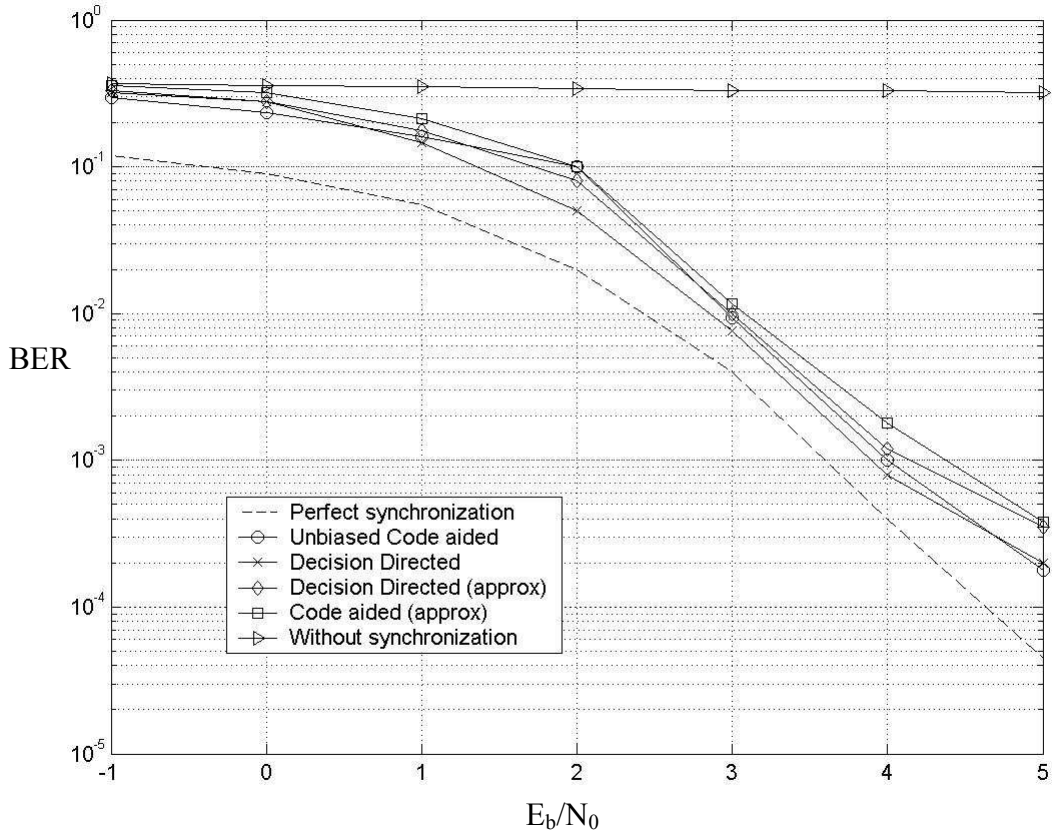


Figure 44: Comparison of iterative synchronization performance. Perfect synchronization; unbiased code-aided synchronization; Decision Directed synchronization; the approximation of Decision Directed synchronizer; the approximation of code-aided synchronization, the first iteration in the synchronization that corresponds to the performance of system without synchronization

Figure 45 represents the minimum square error of channel estimation. The unbiased EM channel estimator has a poor performance in low SNRs but it outperforms other methods in high SNRs. DD channel estimation has a good performance for all SNRs. Using the approximation of the DD method, the performance of channel estimation is degraded slightly. Moreover, the performance of biased estimator and approximate code-aided estimator resemble to the performance of approximation of DD estimator.

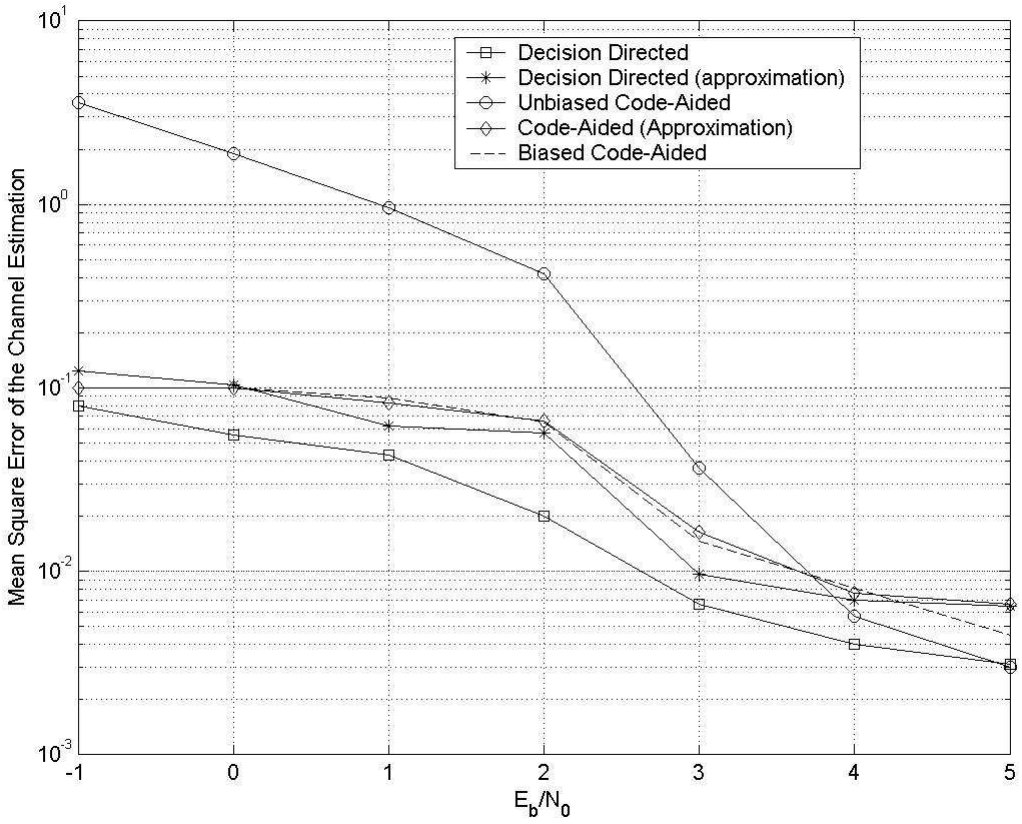


Figure 45: Comparison of iterative channel estimation with different methods: Decision Directed channel estimation; the approximation of Decision Directed channel estimator; unbiased code-aided channel estimator; the approximation of code-aided channel estimator; unbiased code-aided channel estimator.

.

Figure 46 represents the normalized mean square error of carrier frequency offset (CFO). In this figure the results of both code-aided and decision-directed method are depicted. The performance of frequency estimation is approaching the Cramer Rao lower bound in high SNRs. The results of the approximations of code-aided and decision-directed methods are very close to code-aided and decision-directed methods and thus the corresponding curves are not drawn.

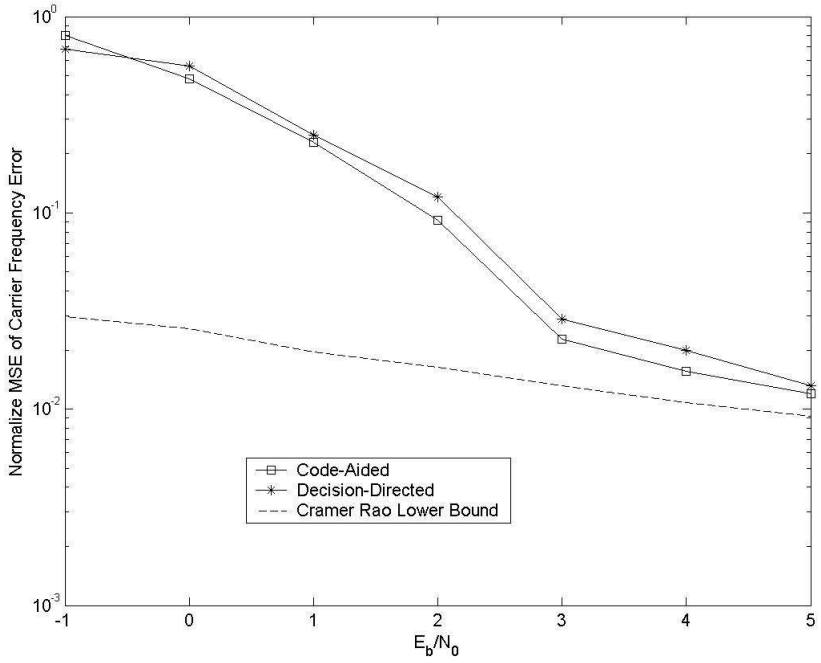


Figure 46: Comparison of the normalized mean square error of carrier frequency offset for both methods: Decision-Directed and Code-Aided. The Cramer-Rao lower bound is also shown

Figure 47 depicts the performance of the MIMO-OFDM system per iteration. For this figure (and for the Figure 48 and Figure 49 as well) we had an unbiased code-aided synchronizer which works in $E_b/N_0$=5dB. The performance of the system improves exponentially. In the first iteration the performance is quite poor and within few iterations it improves considerably. Figure 48 shows the normalized minimum square error of carrier frequency offset estimation. Our estimation improves again exponentially. Figure 49 represents the improvement of channel

estimation. Unlike two previous figures, we do not have the best estimation in the first iteration. That is due to the important effect of the carrier frequency offset on channel estimation. We have set a rough initial value for the carrier frequency error and the performance of channel estimation improves as soon as an improvement is made on the CFO estimate. It can be seen from these figures that even by few iterations carrier frequency offset can be compensated considerably and thus a great deal of improvement in performance is achieved. We can conclude that code-aided synchronization can be a crucial task in the data transmission mode.

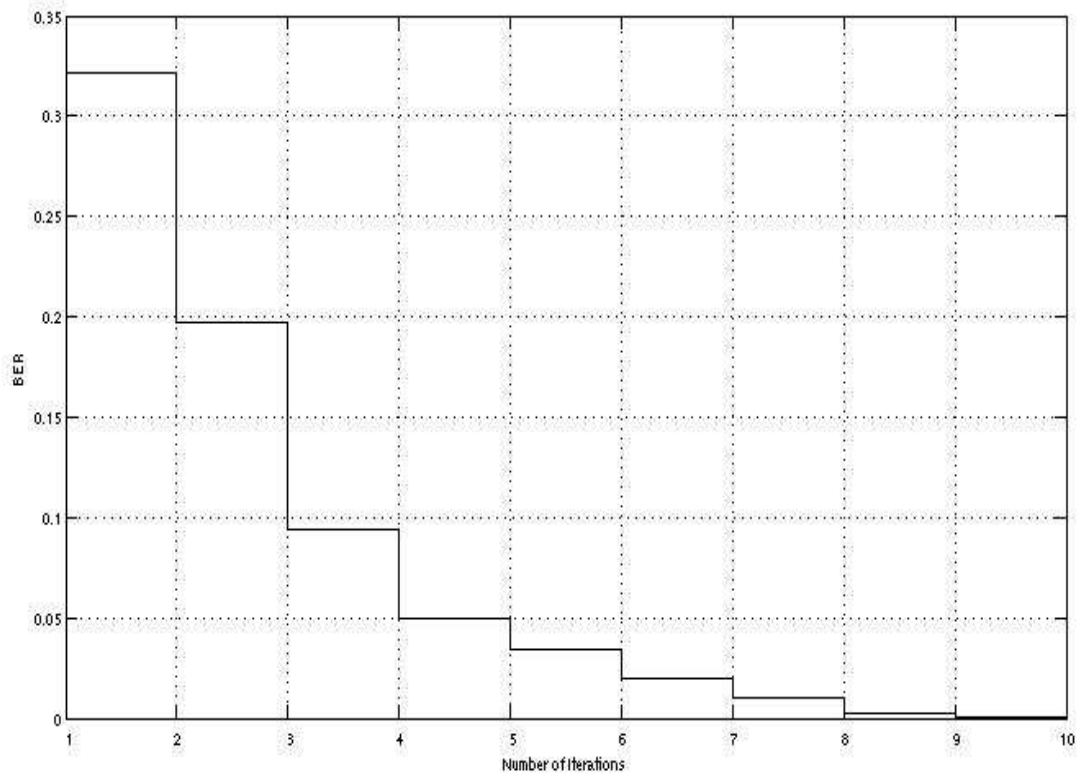


Figure 47: Exponentially improvement of the performance per iteration.

An unbiased code-aided synchronizer is used at $E_b/N_0$=5 dB

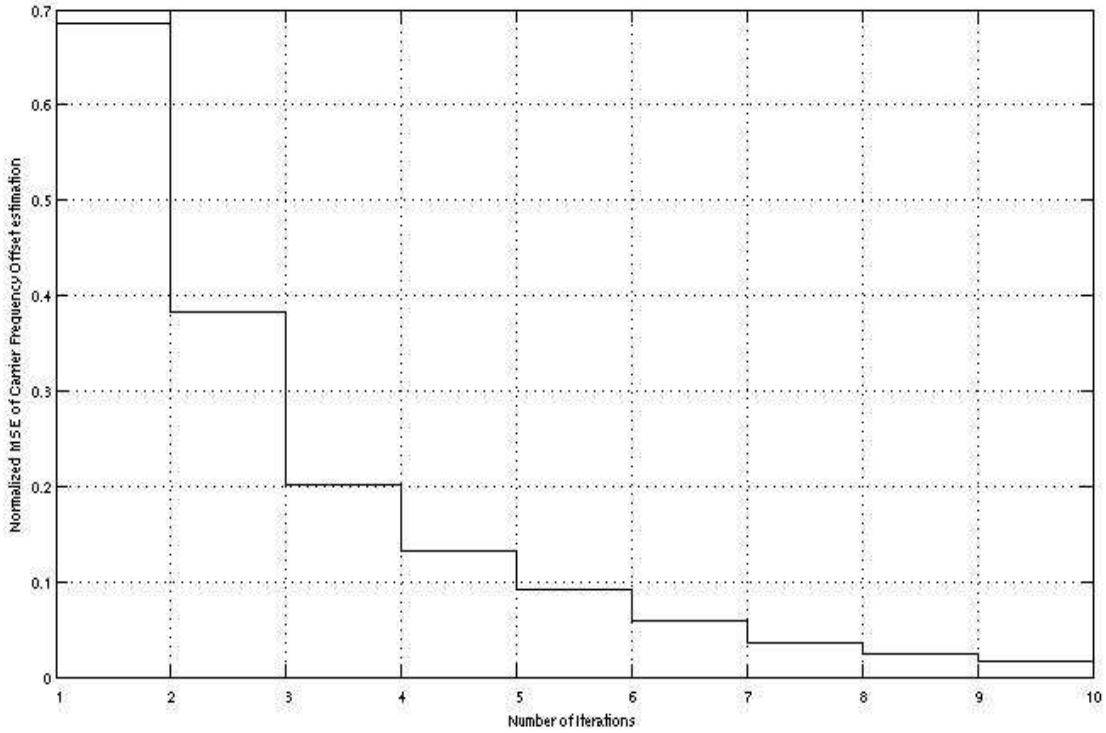


Figure 48: Exponentially improvement of the normalized minimum square error of carrier frequency offset per iteration. An unbiased code-aided synchronizer is used at $E_b/N_0$=5 dB

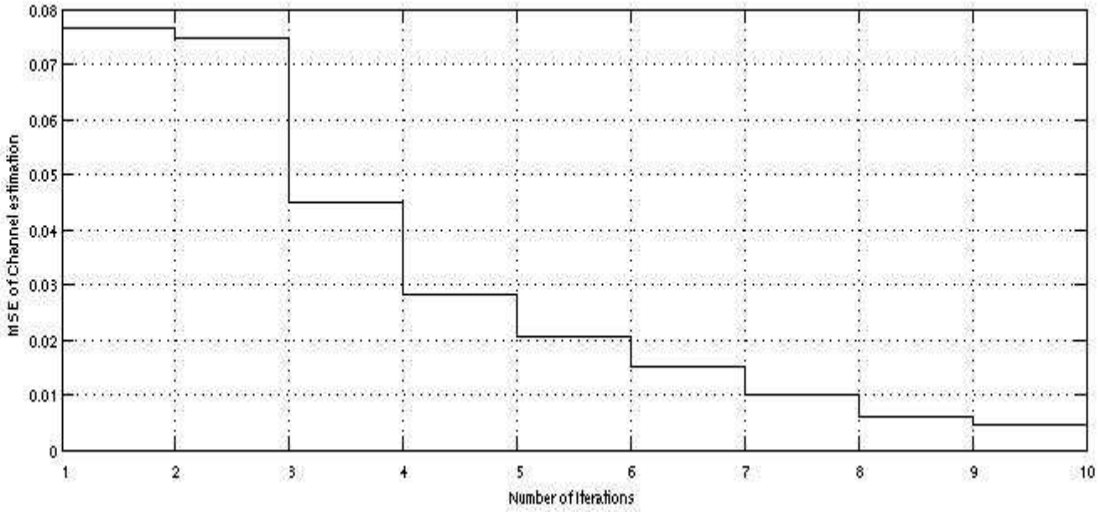


Figure 49: Exponentially improvement of the minimum square error of channel estimation per iteration. An unbiased code-aided synchronizer is used at $E_b/N_0$=5 dB

Figure 50 shows the effect of the carrier frequency offset in the EM estimation of the channel coefficients. In the figure the performance of the data-aided method in the genie estimator, where there is no carrier frequency error and the information of training sequence is fully exploited, is shown; this curve is approximately equivalent with the Cramer-Rao lower bound for the channel estimation. Moreover, in this figure there are the curves for which we have used decision directed EM algorithm to estimate the channel coefficients. For the *initial* estimation, the existence of carrier frequency offset has little effect on the estimation of channel coefficients. On the other hand, the existence of the carrier frequency offset has a relatively considerable impact on the final channel estimation. However this effect will be attenuated in the high SNRs, that is in $E_b/N_0$=5 MSE of channel estimation is in about 0.5 dB from Cramer Rao lower bound. Another interesting point in the figure is that the final estimation of channel in EM method is worse than the initial estimation for the $E_b/N_0$'s lower than 2dB. However, since in the low SNRs part of the carrier frequency offset is compensated the overall performance of the system in terms of BER is improved (Figure 44).

To evaluate the performance of time-synchronization algorithm, we use the same metric as (180) Figure 51 shows the probability of timing failure. The first and the last curves correspond to the optimised Data-Aided algorithm presented in the previous chapter for SISO and MIMO cases. The performances of CA and DD methods, which are close to each other, are between these two curves and in high SNRs the performance of CA and DD symbol timing approaches the optimised data-aided algorithm.

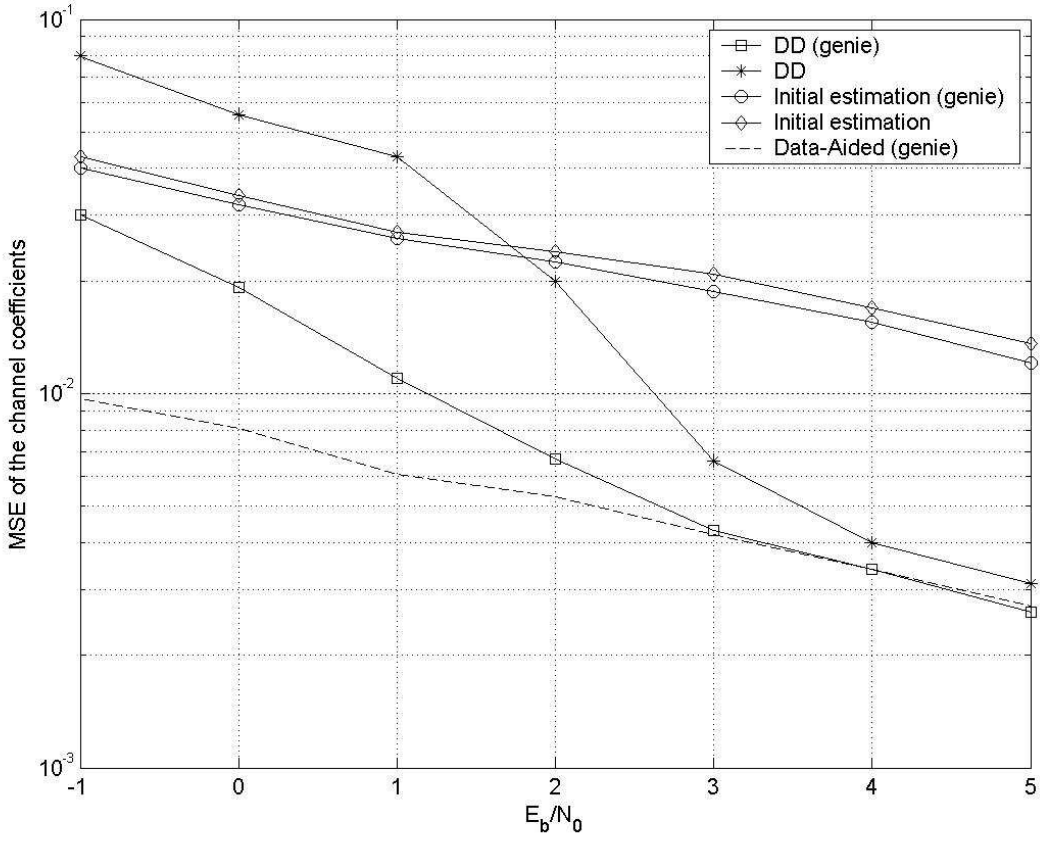


Figure 50: Comparison of channel estimation in different situations: Decision Directed channel estimation in the case where we do not have any carrier frequency error (genie); Decision Directed channel estimation; The results of the first iteration in the EM algorithm in the case where we do not have carrier frequency offset; The results of the first iteration in the EM algorithm; Data aided estimation of a genie estimator.

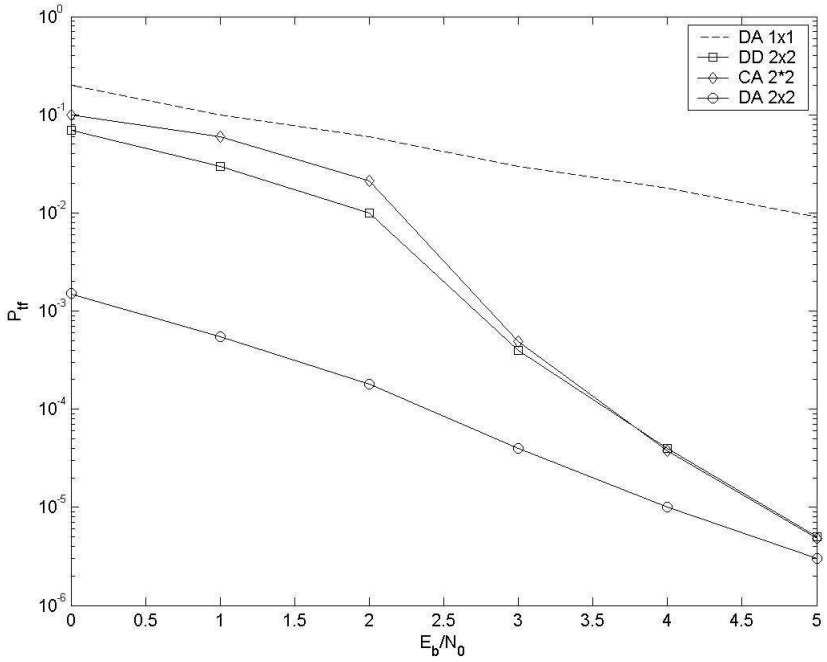


Figure 51: Timing failure probabilities to miss a ±5 interval

## Appendix 1: Computation of $E_{\mathbf{C}_k^H \mathbf{C}_k}$

Concerning the calculation of $E_{\mathbf{C}_k^H \mathbf{C}_k}$, we know from (144) that:

$$\mathbf{C_k} = \left[ \mathbf{C}_{1,k} \quad \ldots \mathbf{C}_{N_r,k} \right]_{N_c \times N_t L}$$

Therefore:

$$\mathbf{C}_\mathbf{k}^H \mathbf{C}_\mathbf{k} = \begin{bmatrix} \mathbf{C}_{1,k}{}^H \mathbf{C}_{1,k} & \cdots & \mathbf{C}_{1,k}{}^H \mathbf{C}_{N_t,k} \\ \vdots & \mathbf{C}_{m,k}{}^H \mathbf{C}_{n,k} & \vdots \\ \mathbf{C}_{N_t,k}{}^H \mathbf{C}_{1,k} & \cdots & \mathbf{C}_{N_t,k}{}^H \mathbf{C}_{N_t,k} \end{bmatrix}_{LN_t \times LN_t}$$

Now, we can calculate $E\left\{ \mathbf{C}_{n,k}^H \mathbf{C}_{m,k} \middle| \mathbf{r}, \mathbf{d}_e^{(i)} \right\}$ as follows:

$$\left[ E\left\{ \mathbf{C}_{n,k}^H \mathbf{C}_{m,k} \middle| \mathbf{r}, \mathbf{d}_e^{(s)} \right\} \right]_{i,j} = \sum_{\vec{\mathbf{c}}_m, \vec{\mathbf{c}}_n} \left[ \mathbf{C}_{m,k}{}^H \mathbf{C}_{n,k} \right]_{i,j} p(\vec{\mathbf{c}}_m, \vec{\mathbf{c}}_n \left| \mathbf{r}, \mathbf{d}_e^{(s)} \right.)$$

$$= \sum_{\vec{\mathbf{c}}_m, \vec{\mathbf{c}}_n} \sum_{k=2}^{N_c+1} c_{m,k-i}{}^* c_{n,k-j} p(\vec{\mathbf{c}}_m, \vec{\mathbf{c}}_n \left| \mathbf{r}, \mathbf{d}_e^{(s)} \right.)$$

$$= \sum_{k=2}^{N_c+1} \sum_{\vec{\mathbf{c}}_m, \vec{\mathbf{c}}_n} c_{m,k-i}{}^* c_{n,k-j} p(\vec{\mathbf{c}}_m, \vec{\mathbf{c}}_n \left| \mathbf{r}, \mathbf{d}_e^{(s)} \right.)$$

$$= \sum_{k=2}^{N_c+1} y_{k-i,k-j}^{mn}$$

where $\vec{\mathbf{c}}_n = \left[ c_{n,0} \quad \ldots \quad c_{n,N_c-1} \right]$ and $y_{p,q}^{mn} = \sum_{\vec{\mathbf{c}}_m, \vec{\mathbf{c}}_n} c_{m,p}{}^* c_{n,q} p(\vec{\mathbf{c}}_m, \vec{\mathbf{c}}_n \left| \mathbf{r}, \mathbf{d}_e^{(s)} \right.)$

Suppose $\mathbf{f}_p = \frac{1}{\sqrt{N_c}} [e^{2\pi jpk/N_c}]^T \quad k = 0, \ldots, N_c - 1$ is a row of IDFT matrix. Therefore, $c_{m,p} = \vec{\mathbf{a}}_m \mathbf{f}_p^H$ where $\vec{\mathbf{a}}_n = \left[ a_{n,0} \quad \ldots \quad a_{n,N_c-1} \right]$. $y_{p,q}^{mn}$ can be calculated as below:

$$y_{p,q}^{mn} = \sum_{\vec{\mathbf{c}}_m, \vec{\mathbf{c}}_n} c_{m,p}{}^* c_{n,q} p(\vec{\mathbf{c}}_m, \vec{\mathbf{c}}_n) = \sum_{\vec{\mathbf{a}}_m, \vec{\mathbf{a}}_n} \mathbf{f}_p \vec{\mathbf{a}}_m{}^H \vec{\mathbf{a}}_n \mathbf{f}_q^H p(\vec{\mathbf{a}}_m, \vec{\mathbf{a}}_n \left| \mathbf{r}, \mathbf{d}_e^{(s)} \right.)$$

$$= \sum_{\vec{\mathbf{a}}_m, \vec{\mathbf{a}}_n} \mathbf{f}_p \left[ a_{m,i}{}^* a_{n,j} \right]_{N_c \times N_c} \mathbf{f}_q^H p(\vec{\mathbf{a}}_m, \vec{\mathbf{a}}_n \left| \mathbf{r}, \mathbf{d}_e^{(s)} \right.)$$

$$= \mathbf{f}_p \left[ \mathrm{E}(a_{m,i}{}^* a_{n,j} \left| \mathbf{r}, \mathbf{d}_e^{(s)} \right.) \right]_{N_c \times N_c} \mathbf{f}_q^H$$

$$= \mathbf{f} \mathbf{M} \mathbf{f}_q^H$$

and therefore:

$$\mathbf{y} = \mathbf{F}.\mathbf{M}.\mathbf{F}^H$$

where $\mathbf{F}$ is $N_c \times N_c$ IDFT matrix equal to $\left[ \mathbf{F} \right]_{m,n} = \frac{1}{\sqrt{N_c}} e^{2\pi jmn/N_c}$.

$a_{m,i}$ and $a_{n,j}$ are independent if $m \neq n$ and $i \neq j$ and so:

$\mathsf{E}\left(a_{m,i}{}^{*}a_{n,j}\left|\mathbf{r},\mathbf{d}_{e}{}^{(s)}\right.\right)=\mathsf{E}\left(a_{m,i}\left|\mathbf{r},\mathbf{d}_{e}{}^{(s)}\right.\right)^{*}\mathsf{E}\left(a_{n,j}\left|\mathbf{r},\mathbf{d}_{e}{}^{(s)}\right.\right)$ (if $m\neq n$ and $i\neq j$)

And if $m=n$ and $i=j$, $\mathsf{E}\left(a_{m,i}{}^{*}a_{n,j}\right)=1$ in case of using QPSK modulation.

Hence to calculate $E_{\mathbf{C}_k^H\mathbf{C}}$ we have to calculate $E\left\{\mathbf{C}_{n,k}^H\mathbf{C}_{m,k}\left|_{\mathbf{r},\mathbf{d}_e{}^{(i)}}\right.\right\}$ and we showed that the $i^{\text{th}}$, $j^{\text{th}}$ element of $E\left\{\mathbf{C}_{n,k}^H\mathbf{C}_{m,k}\left|_{\mathbf{r},\mathbf{d}_e{}^{(i)}}\right.\right\}$ is equal to $\sum_{k=2}^{N_c+1}y_{k-i,k-j}^{mn}$ where $\mathbf{y}=\mathbf{F}.\mathbf{M}.\mathbf{F}^H$. The off-diagonal elements of matrix **M** are equal to $\mathsf{E}\left(\vec{\mathbf{a}}_m\left|\mathbf{r},\mathbf{d}_e{}^{(s)}\right.\right)^H\mathsf{E}\left(\vec{\mathbf{a}}_n\left|\mathbf{r},\mathbf{d}_e{}^{(s)}\right.\right)$ and the diagonal elements of **M** are equal to one.

However, $E_{\mathbf{C}_k^H\mathbf{C}_k}$ is, in practice, approximately equal to identity matrix.

Perspective Chapter 7

# MIMO-OFDMA SYNCHRONIZATION

## 7.1 Introduction

OFDM systems have given birth to OFDMA (Orthogonal Frequency Division Multiple Access) systems. In the case of OFDMA, the set of available carriers is divided into groups called subchannels and each user is assigned with one or multiple subchannels.

Of course, as the classical OFDM transmission scheme, OFDMA is very sensitive to Carrier Frequency Offset (CFO) between the transmitter and the receiver. The presence of CFO entails loss of orthogonality among carriers, leading to severe BER performance

degradation. In addition, coherent detection which constitutes the most performing demodulation principle, implies an accurate Channel Impulse Response (CIR) estimation algorithm at the receiver. The combination of CFO compensation algorithm and CIR guessing algorithm leads to particularly complex algorithms in uplink OFDMA due to the number of unknowns.

The CFO estimation problem for uplink OFDMA has been already addressed in several papers [133]-[135]. However, except the recent papers of Morelli [136]-[138], none considers the extreme case in which any available carrier frequency can be allocated to each user. In fact two methods are classically proposed for carrier assignment: the sub-band carrier assignment and the interleaved carrier assignment. In each case, frequency carriers are always assigned by groups according to particular spectrum sharing rules.

Morelli [136] proposes to investigate the case where a new user enters the network and, assuming that the remaining users are already synchronized, he shows how to synchronize the new user. In this chapter we investigate a more complex case because we propose a ML algorithm which enables to jointly estimate CFO and channel for all new users simultaneously and regardless of the carrier assignment scheme. We propose to study the case where all users enter simultaneously the system. In this case, we assume that they send out pilot symbols on the carriers only assigned to them. The proposed synchronization procedure is based on the EM algorithm and it can be easily interfaced with MAP-EM demodulators. Using the EM principle for OFDMA synchronization in a MIMO system is a new approach and, within the EM loop, we propose MIMO OFDM original synchronization algorithms which yield to closed form solutions for the estimation of each user's CFO and channel parameters. Particularly, we propose a new powerful joint CFO-channel estimation algorithm within the EM loop which enables to converge within only a few iterations. The main drawback of the proposed algorithms is that new users entering the network have to send training data sequentially on their transmit antennas i.e. at a given time instant, only one transmit antenna per user is switched on. In fact, using an EM-based approach enables to decompose a computationally intensive multi-dimensional optimization problem into separate smaller ML optimization problems. Very recently, Morelli & al in [138] followed a similar approach with the alternating-projection algorithm. As in our case, this enables the authors to consider whatever kind of carrier assignment schemes. However, similar to our above mentioned constraint on the way training data are sent into the network, the carrier

synchronization algorithm in this paper proceeds sequentially instead of jointly. As for Morelli's paper, simulation results show the ability of our proposed algorithm to reach the Cramer Rao lower bound. However, the big difference between the two proposed papers, is that we consider here the case of MIMO OFDMA systems whereas [138] deals with the case of Single Input Single Output (SISO) systems and, due to this difference, a fair comparison between the two systems is difficult. We think that the algorithm presented in [138] can be extended to the case of MIMO systems but, due to the use of performing algorithms within the EM loop, we believe that our algorithms will converge faster than an extended version of [138].

# 7.2 System Description

We consider a MIMO OFDMA system with $N_c$ carriers and $N_u$ users using LDPC channel encoders. Each user is equipped with $N_t$ transmit antennas and the Base Station has $N_r$ received antennas. We denote $\Omega$ the set of all possible transmitted symbols. The $N_t.\ N_c.\log_2\Omega$ information bits of user $k$ are first encoded by a rate $R=1/N$ LDPC encoder into $(N.N_t.N_c.\log_2(\Omega))$ coded bits and then the binary LDPC coded bits are modulated into $(N.N_t.N_c)$ MPSK symbols. These $(N.N_t.N_c)$ symbols are split into $N$ streams of $(N_t.N_c)$ symbols.

Then, the $(N_t.N_c)$ symbols $x^k_{1,0}(n),...,x^k_{1,N_c-1}(n),x^k_{2,0}(n),...,x^k_{2,N_c-1}(n),...,x^k_{N_t,0}(n),...,x^k_{N_t,N_c-1}(n)$ enter $N_t$ IFFT modules of size $N_c$. The number of IFFT modules is equal to the number of transmit antennas. For a given output block $\boldsymbol{s}^k_p(n)_{N_c\times1}$, $p=1,2,...,N_t$ of an IFFT module ($p^{\text{th}}$ block for example corresponding to $\boldsymbol{x}^k_p(n)=[x^k_{p,0}(n),x^k_{p,1}(n),...,x^k_{p,N_c-1}(n)]^T_{N_c\times1}$), there are some zero components where the corresponding carriers are not used ($s^k_{p,j}(n)=0$ for carriers $j$ = $n_1$,$n_2$,…).

The corresponding time-domain output of the IFFT module can be written as:

$$\boldsymbol{s}^k_p(n)_{N_c\times1}=\boldsymbol{W}^H_{N_c\times N_c}.\boldsymbol{x}^k_p(n)_{N_c\times1} \tag{211}$$

with $\boldsymbol{W}(n,m)=\frac{1}{\sqrt{N_c}}.\exp(-j.2.\pi.(n-1).(m-1)/N_c)$. A cyclic prefix (CP) of length $N_{CP}$ is then inserted in $\boldsymbol{s}^k_p(n)$ to form $\boldsymbol{u}^k_p(n)$ of length $N_s=N_c+N_{CP}$ to combat the dispersive nature of the transmission channel. This operation can be described as: $\boldsymbol{u}^k_p(n)_{N_s\times1}=\boldsymbol{\Theta}^k_{p\,N_s\times N_c}.\boldsymbol{s}^k_p(n)_{N_c\times1}$ where $\boldsymbol{\Theta}^k_p$ denotes the matrix of CP added symbols. The channel between transmit antenna $p$ and receive antenna $q$ for user $k$ is a MIMO frequency selective channel and it is denoted as:

$$\boldsymbol{h}^k_{pq}=[h^k_{pq}(1),...,h^k_{pq}(L^k_{p,q})]^T \tag{212}$$

This expression takes the shaping filters' impulse response into account. Using (212) and the definition of $\boldsymbol{u}^k_p(n)$ and assuming the presence of CFO and timing errors in reception we can write the output of the BS receive filter at time $m$ for receive antenna $q$ as:

$$r_q(m)=\sum_{k=1}^{N_u}\left\{e^{j.2\pi\varepsilon_k.m}\sum_{p=1}^{N_t}\sum_{l=1}^{L^k_{p,q}}h^k_{pq}(l).u^k_p(m-l-\mu^k_{p,q})\right\}+w_q(m) \tag{213}$$

$\varepsilon_k = \Delta f_k T_s$ is the normalized angular frequency offset with respect to the carrier spacing for the $k^{\text{th}}$ user. $\mu_{p,q}^k$ corresponds to the $k^{\text{th}}$ user's integer-valued timing error encountered over the $p^{\text{th}}$ transmit $q^{\text{th}}$ receive antenna link. The fractional timing error is indistinguishable from the phase of the channel impulse response and is incorporated in $\boldsymbol{h}_{pq}^k$. $w_q(m)$ is a zero-mean white Gaussian noise sample. Note that for normalization purpose, the transmitted power on each antenna is equal to $1/N_t$. Figure 52 illustrates the synchronization problem with the definition of $\mu_{p,q}^k$ in the case $q = 1$. At the BS station, the serial-to-parallel (S/P) conversion transforms $r_q(m)$ into $\boldsymbol{r}_q(n)_{Q\times 1}$. We then form the vector $\boldsymbol{y}_q(n)$ by discarding the CP from $\boldsymbol{r}_q(n)$. To simplify the complex estimation problem of the relative delays $\mu_{p,q}^k$, the MIMO channel impulse response $\boldsymbol{h}_{p,q}^k$ of each user and the offset frequency $\varepsilon_k$ we first show that the estimation of the relative delays can be embedded into the channel impulse response estimation problem. To do this we define at first $L_{\max} = \max_{p,q,k}\left\{\mu_{p,q}^k + L_{p,q}^k\right\}$. If we assume that $N_{CP} \geq L_{\max}$, each received block $\boldsymbol{r}_q(n)$ $q = 1,\ldots,N_r$ contains only intersymbol interference from its immediate previous block and, after the cyclic prefix removal, the blocks $\boldsymbol{y}_q(n) = [y_q(n.N_s), y_q(n.N_s+1), ...., y_q(n.N_s+N_c-1)]^T$, contain no intersymbol interference. This condition generally holds for OFDMA systems. After CP removal, the received signal $\boldsymbol{y}_q(n)$ can be written in matrix form as:

$$\boldsymbol{y}_q(n) = \sum_{k=1}^{N_u} e^{j.\tilde{\varepsilon}_k} \boldsymbol{E}(\varepsilon_k) \sum_{p=1}^{N_t} \boldsymbol{A}_p^k(\mu_{p,q}^k)\boldsymbol{h}_{p,q}^k + \mathbf{w}_q(n) \tag{214}$$

We have $\tilde{\varepsilon}_k = \varepsilon_k(nN_s + N_{CP})$ which is the phase term associated with block $n$ and $\boldsymbol{E}(\varepsilon_k) = diag(1, e^{j.2\pi\varepsilon_k}, \cdots, e^{j2\pi(K-1)\varepsilon_k})_{N_c\times N_c}$. The definition of the circulant matrix $\boldsymbol{A}_p^k(\mu_{p,q}^k)$ is given just below.

$$\boldsymbol{A}_p^k(\mu_{p,q}^k) = \begin{bmatrix} u_p^k(n-\mu_{p,q}^k) & u_p^k(n-1-\mu_{p,q}^k) & \cdots & u_p^k(n+1-L_{p,q}^k-\mu_{p,q}^k) \\ u_p^k(n+1-\mu_{p,q}^k) & u_p^k(n+\mu_{p,q}^k) & \cdots & u_p^k(n+2-L_{p,q}^k-\mu_{p,q}^k) \\ \vdots & \vdots & \ddots & \vdots \\ u_p^k(n+N_c-1-\mu_{p,q}^k) & u_p^k(n+N_c-2-\mu_{p,q}^k) & \cdots & u_p^k(n+N_c-L_{p,q}^k-\mu_{p,q}^k) \end{bmatrix}_{N_c\times L_{p,q}^k} \tag{215}$$

Using the definition of $\boldsymbol{A}_i^k(\mu_i^k)$ we have the following relationship:

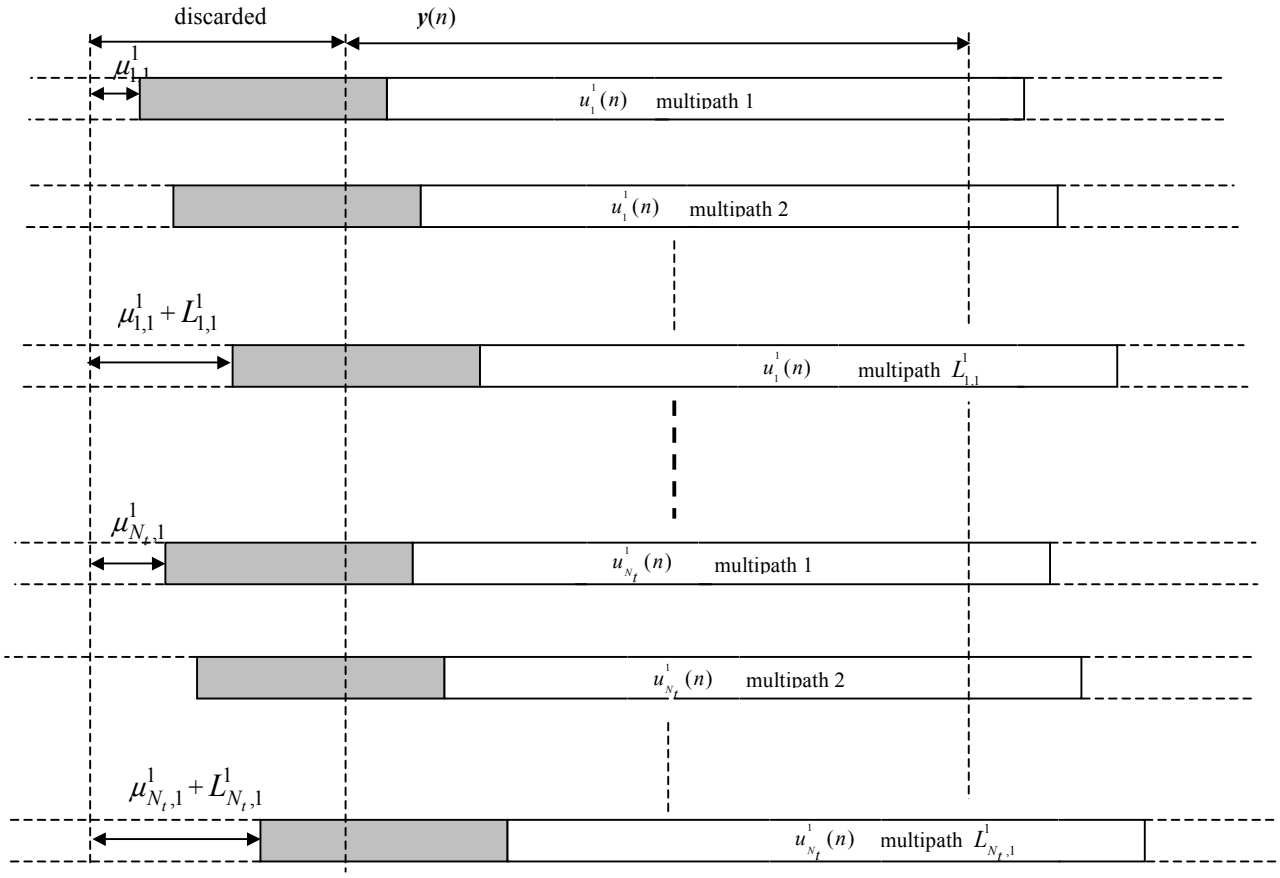


Figure 52: Illustration of multipath and timing errors for user 1, receive antenna 1.

$$\begin{aligned}
\boldsymbol{A}_p^k(\mu_{p,q}^k).\boldsymbol{h}_{p,q}^k &= \begin{bmatrix} u_p^k(n-\mu_{p,q}^k) & u_p^k(n-1-\mu_{p,q}^k) & \cdots & u_p^k(n+1-L_{p,q}^k-\mu_{p,q}^k) \\ u_p^k(n+1-\mu_{p,q}^k) & u_p^k(n+\mu_{p,q}^k) & \cdots & u_p^k(n+2-L_{p,q}^k-\mu_{p,q}^k) \\ \vdots & \vdots & \ddots & \vdots \\ u_p^k(n+N_c-1-\mu_{p,q}^k) & u_p^k(n+N_c-2-\mu_{p,q}^k) & \cdots & u_p^k(n+N_c-L_{p,q}^k-\mu_{p,q}^k) \end{bmatrix}_{N_c\times L_{p,q}^k} \boldsymbol{h}_{p,q\,L_{p,q}^k\times 1}^k \\
&= \begin{bmatrix} u_p^k(n) & u_p^k(n-1) & \cdots & u_p^k(n-N_{CP}+1) \\ u_p^k(n+1) & u_p^k(n) & \cdots & u_p^k(n-N_{CP}+2) \\ \vdots & \vdots & \ddots & \vdots \\ u_p^k(n+N_c-1) & u_p^k(n+N_c-2) & \cdots & u_p^k(n+N_c-N_g) \end{bmatrix}_{N_c\times N_{CP}} \cdot \begin{bmatrix} \mathbf{0}_{\mu_{p,q}^k\times 1} \\ \boldsymbol{h}_{p,q\,L_{p,q}^k\times 1}^k \\ \mathbf{0}_{(N_g-\mu_{p,q}^k-L_{p,q}^k)\times 1} \end{bmatrix}_{N_{CP}\times 1} \\
&= \boldsymbol{A}_p^k(n).\boldsymbol{h}_{p,q,\mu_{p,q}^k}^k
\end{aligned} \tag{216}$$

Equation (216) clearly shows that the estimation of the delays $\mu_{p,q}^k$ can be embedded into the estimation of the channel impulse response. Hence, the CIR $\boldsymbol{h}_{p,q}^k$ becomes $\boldsymbol{h}_{p,q,\mu_{p,q}^k}^k$. In fact, the position of the first non-zero element in $\boldsymbol{h}_{p,q,\mu_{p,q}^k}^k$ indicates the timing error associated with user *k*

and the parameter $L_{p,q}^k$ can be found by counting the number of non-zero elements in $\boldsymbol{h}_{p,q,\mu_{p,q}^k}^k$. We can rewrite (216) in the following form:

$$\begin{aligned}\boldsymbol{y}_q(n) &= \sum_{k=1}^{N_u} e^{j.\tilde{\varepsilon}_k} \boldsymbol{E}(\varepsilon_k) \sum_{p=1}^{N_t} \boldsymbol{A}_p^k(n) \boldsymbol{h}_{p,q,\mu_{p,q}^k}^k + \mathbf{w}_q(n) \\ &= \sum_{k=1}^{N_u} \sum_{p=1}^{N_t} \boldsymbol{G}_p^k(\varepsilon_k) \boldsymbol{h}_{p,q,\mu_{p,q}^k}^k + \mathbf{w}_q(n)\end{aligned} \tag{217}$$

with $\boldsymbol{G}_p^k(\varepsilon_k) = e^{j.\tilde{\varepsilon}_k} \boldsymbol{E}(\varepsilon_k).\boldsymbol{A}_p^k(n)$. For the next section and the EM algorithm derivation, we assume the case when all users are new users who want to enter the system. It constitutes the most difficult situation due to the high number of unknowns. To solve this critical issue, we suppose that all users send out pilot symbols on the carriers assigned only to them in the *n*th block simultaneously. The pilot symbols are organized in the same way as those given in [140], that is, the first OFDM word in the packet contains pilot symbol and the others carry out information symbols.

# 7.3 EM synchronization algorithms

For the following steps we will consider the most difficult case when all users are new users. Since $\mathbf{w}(n)$ is assumed to be a zero-mean Gaussian noise, the least-squares solution is also the maximum likelihood (ML) solution. The maximum likelihood estimates of $\varepsilon_k$ and $\boldsymbol{h}_{i,\mu_i^k}^k$ are given by minimizing the following quadratic cost function:

$$\min_{\varepsilon,\boldsymbol{h}} \left\{ \sum_{q=1}^{N_r} \left| \boldsymbol{y}_q(n) - \sum_{k=1}^{N_u} \sum_{p=1}^{N_t} \boldsymbol{G}_p^k(\varepsilon_k) \boldsymbol{h}_{p,q,\mu_{p,q}^k}^k \right|^2 \right\} \tag{218}$$

with: $\boldsymbol{\varepsilon} = \left[ \varepsilon_1, \varepsilon_2, \cdots, \varepsilon_{N_u} \right]^T$ and

$$\boldsymbol{h} = [(\boldsymbol{h}_{1,1,\mu_{1,1}^1}^1)^T, (\boldsymbol{h}_{1,2,\mu_{1,2}^1}^1)^T, ..., (\boldsymbol{h}_{1,N_r,\mu_{1,N_r}^1}^1)^T, ..., (\boldsymbol{h}_{N_t,1,\mu_{N_t,1}^1}^1)^T, (\boldsymbol{h}_{N_t,2,\mu_{N_t,2}^1}^1)^T, ..., (\boldsymbol{h}_{N_t,N_r,\mu_{N_t,N_r}^1}^1)^T, ...$$
$$..., (\boldsymbol{h}_{1,1,\mu_{1,1}^{N_u}}^{N_u})^T, (\boldsymbol{h}_{1,2,\mu_{1,2}^{N_u}}^{N_u})^T, ..., (\boldsymbol{h}_{1,N_r,\mu_{1,N_r}^{N_u}}^{N_u})^T, ..., (\boldsymbol{h}_{N_t,1,\mu_{N_t,1}^{N_u}}^{N_u})^T, (\boldsymbol{h}_{N_t,2,\mu_{N_t,2}^{N_u}}^{N_u})^T, ..., (\boldsymbol{h}_{N_t,N_r,\mu_{N_t,N_r}^{N_u}}^{N_u})^T]^T$$

Instead of directly solving the multi-dimensional optimization problem we propose to use an EM (Expectation-Maximization) approach. Our approach here is based on Feder and E.Weinstein's approach in [126]. They use EM-algorithm for parameter estimation of superimposed signals. Xie and Georghiades in [130] used the same method to convert MIMO-OFDM channel estimation

into several SISO OFDM channel estimation. We use here the same method for OFDMA synchronization. It should be noted that Feder and E.Weinstein use EM algorithm in a special case and although their equations are driven from the classical EM equations they appear somewhat different:

## 7.3.1 Using EM algorithm

$\boldsymbol{y}$ is the observed data and as the complete data we choose $\boldsymbol{d}_q^k$ , $k = 1,2,\ldots,N_u$ which is given by:

$$\boldsymbol{d}_q^k = \sum_{p=1}^{N_t} \boldsymbol{G}_p^k(\varepsilon_k).\boldsymbol{h}_{p,q,\mu_{p,q}^k}^k + \mathrm{w}_q^k \tag{219}$$

where $\mathbf{w}_q = \sum_{k=1}^{N_u} \mathbf{w}_q^k$ . Thus, $\boldsymbol{y}_q = \sum_{k=1}^{N_u} \boldsymbol{d}_q^k$ , where $\boldsymbol{d}_q^k$ is the component in the received signal $\boldsymbol{y}_q$ contributed by the $k$th user. The EM algorithm starts with an arbitrary initial guess $\hat{\varepsilon}_k(0)$ $k$ = $1,\ldots,N_u$, and $\hat{\boldsymbol{h}}_{p,q,\mu_{p,q}^k}^k(0)$ , $k = 1,\ldots,N_u$, $p = 1,\ldots, N_t$, $q = 1,\ldots,N_r$. We suppose that at the $m^{\text{th}}$ stage of the algorithm, we have obtained the set of estimated values $[\hat{\boldsymbol{\varepsilon}}(m), \hat{\boldsymbol{h}}(m)]$:

$\hat{\boldsymbol{\varepsilon}}(m) = \left[\hat{\varepsilon}_1(m), \hat{\varepsilon}_2(m), \cdots, \hat{\varepsilon}_{N_u}(m)\right]^T$ and

$\hat{\boldsymbol{h}}(m) = [(\hat{\boldsymbol{h}}_{1,1,\mu_{1,1}^1}^1(m))^T, \ldots, (\hat{\boldsymbol{h}}_{1,N_r,\mu_{1,N_r}^1}^1(m))^T, \ldots, (\hat{\boldsymbol{h}}_{N_t,1,\mu_{N_t,1}^1}^1(m))^T, \ldots, (\hat{\boldsymbol{h}}_{N_t,N_r,\mu_{N_t,N_r}^1}^1(m))^T, \ldots$

$, \ldots, (\hat{\boldsymbol{h}}_{1,1,\mu_{1,1}^{N_u}}^{N_u}(m))^T, \ldots, (\hat{\boldsymbol{h}}_{1,N_r,\mu_{1,N_r}^{N_u}}^{N_u}(m))^T, \ldots, (\hat{\boldsymbol{h}}_{N_t,1,\mu_{N_t,1}^{N_u}}^{N_u}(m))^T, \ldots, (\hat{\boldsymbol{h}}_{N_t,N_r,\mu_{N_t,N_r}^{N_u}}^{N_u}(m))^T]^T$

For the $(m+1)^{\text{th}}$ iteration, the parameters are updated by the following procedure:

***E-step* (*Expectation Step*)**: As Feder and Weinstein showed in [126] the calculation of expected value of Q function will lead to calculate the following terms:

For $k = 1,\ldots,N_u$, $p = 1,\ldots,N_t$ and $q = 1,\ldots, N_r$ :

$$\hat{\boldsymbol{b}}_{p,q}^k(m) = \boldsymbol{G}_p^k(\hat{\varepsilon}_k(m)).\hat{\boldsymbol{h}}_{p,q,\mu_{p,q}^k}^k(m) \tag{220}$$

$$\hat{\boldsymbol{d}}_{p,q}^k(m) = \hat{\boldsymbol{b}}_{p,q}^k(m) + \beta_k\left[\boldsymbol{y}_{p,q} - \sum_{j=1}^{N_u} \hat{\boldsymbol{b}}_{p,q}^j(m)\right] \tag{221}$$

$$\hat{\boldsymbol{d}}_q^k(m) = \sum_{p=1}^{N_t} \hat{\boldsymbol{d}}_{p,q}^k(m) \tag{222}$$

where $\beta_k$'s are chosen such that $\sum_{k=1}^{N_u} \beta_k = 1$ and $\boldsymbol{y}_{p,q}$ represents the sum of the contributions from transmit antenna $p$ to the complete receive vector of antenna $q$: $\boldsymbol{y}_q = \sum_{\boldsymbol{p}} \boldsymbol{y}_{p,q} = \sum_{p=1}^{N_t} \sum_{k=1}^{N_u} \boldsymbol{y}_{p,q}^k$. In general, $\boldsymbol{y}_{p,q} = \sum_{k=1}^{N_u} \boldsymbol{y}_{p,q}^k$ can't be obtained directly from $\boldsymbol{y}_q$. To obtain this parameter vector, one has to assume that the different transmit antennas of each user send their training data sequentially in the network i.e. the other transmit antennas are switched off. This means that during the training period, we have to solve $N_u$ parallel SIMO identification problems. This implies a quasi-synchronous system in a micro-cell environment for example. In this case, each user would achieve timing acquisition through a downlink synchronization channel before initiating the uplink transmission. In the case of SIMO systems where each user is equipped with only one transmit antenna, the problem is directly simplified and the Expectation Step can be written with obvious notations:

$$\hat{\boldsymbol{b}}_q^k(m) = \boldsymbol{G}^k(\hat{\varepsilon}_k(m)).\hat{\boldsymbol{h}}_{q,\mu_q^k}^k(m) \tag{223}$$

$$\hat{\boldsymbol{d}}_q^k(m) = \hat{\boldsymbol{b}}_q^k(m) + \beta_k \left[ \boldsymbol{y}_q - \sum_{j=1}^{N_u} \hat{\boldsymbol{b}}_q^j(m) \right] \tag{224}$$

***M-step* (*Maximization Step*)**: Moreover, Feder and Weinstein showed in [126] that the maximization of Q function entails the following minimization:

For $k = 1,\ldots,N_u$:

$$[\hat{\varepsilon}_k(m+1), \hat{\boldsymbol{h}}^k(m+1)] = \arg\min_{\varepsilon_k, \boldsymbol{h}^k} \left\{ \sum_{q=1}^{N_r} \left| \hat{\boldsymbol{d}}_q^k(m) - \sum_{p=1}^{N_t} \boldsymbol{G}_p^k(\varepsilon_k).\boldsymbol{h}_{p,q,\mu_{p,q}^k}^k \right|^2 \right\} \tag{225}$$

with: $\hat{\boldsymbol{h}}^k(m+1) = [\hat{\boldsymbol{h}}_{1,1,\mu_{1,q}^k}^k(m+1),\ldots,\hat{\boldsymbol{h}}_{N_t,1,\mu_{N_t,q}^k}^k(m+1),\ldots,\hat{\boldsymbol{h}}_{1,N_r,\mu_{1,q}^k}^k(m+1),\ldots,\hat{\boldsymbol{h}}_{N_t,N_r,\mu_{N_t,q}^k}^k(m+1)]^T$. The problem with (225) is that we have to estimate two parameters with only one cost function. We propose at first to estimate the parameter $\hat{\varepsilon}_k(m+1)$. We have:

$$\sum_{p=1}^{N_t} \boldsymbol{G}_p^k(\varepsilon_k).\boldsymbol{h}_{p,q,\mu_{p,q}^k}^k = e^{j.\tilde{\varepsilon}_k} \boldsymbol{E}(\varepsilon_k).\sum_{p=1}^{N_t} \boldsymbol{A}_p^k(n).\boldsymbol{h}_{p,q,\mu_{p,q}^k}^k \tag{226}$$

We can use a matrix form to rewrite equation (226). We obtain:

$$\sum_{p=1}^{N_t} \boldsymbol{G}_p^k(\varepsilon_k).\boldsymbol{h}_{p,q,\mu_{p,q}^k}^k = \boldsymbol{V}_F(\varepsilon_k)_{K\times N_t.N_g}.\boldsymbol{h}_{q\,N_t.N_g\times 1}^k \tag{227}$$

with: $\boldsymbol{h}_q^k = \left(\boldsymbol{h}_{1,q,\mu_{1,q}^k}^k \;\; \boldsymbol{h}_{2,q,\mu_{2,q}^k}^k \;\; ... \;\; \boldsymbol{h}_{N_t,q,\mu_{N_t,q}^k}^k\right)^T_{N_t.N_{CP}\times 1}$, $\boldsymbol{V}_F(\varepsilon_k) = e^{j.\tilde{\varepsilon}_k}.\boldsymbol{E}(\varepsilon_k).\boldsymbol{A}_k(n)$

and $\boldsymbol{A}_k(n) = \left[\boldsymbol{A}_1^k(n),...,\boldsymbol{A}_{N_t}^k(n)\right]_{K\times N_t.N_g}$

The ML estimation of $\varepsilon_k$ leads to the following mathematical development:

$$\hat{\varepsilon}_k = \arg\max Q_k(\varepsilon_k) = \arg\max \sum_{q=1}^{N_r} \log p(\hat{\boldsymbol{d}}_q^k(m)\big|\boldsymbol{E}(\omega_k)) \tag{228}$$

Assuming that: $\hat{\boldsymbol{d}}_q^k(m) = \sum_{p=1}^{N_t} \boldsymbol{G}_p^k(\varepsilon_k).\boldsymbol{h}_{p,q,\mu_{p,q}^k}^k + \mathbf{w}_q^k(m)$ and following the same approach as those given in [140], we obtain:

$$\begin{aligned} Q_k(\varepsilon_k) &= -\sum_{q=1}^{N_r} (E_{\boldsymbol{h}_q^k\left|\hat{\boldsymbol{d}}_k^q(m),\varepsilon_k\right.}\left\{\left\| \hat{\boldsymbol{d}}_q^k(m) - \boldsymbol{V}_F(\varepsilon_k).\boldsymbol{h}_q^k \right\|^2\right\} + const) \\ &= -\sum_{q=1}^{N_r} (\left\| \hat{\boldsymbol{d}}_q^k(m) - \boldsymbol{V}_F(\varepsilon_k).\tilde{\boldsymbol{h}}_q^k \right\|^2 + \text{trace}(\boldsymbol{V}_F(\varepsilon_k).\hat{\boldsymbol{\Sigma}}_{\boldsymbol{h}_q^k}.\boldsymbol{V}_F^H(\varepsilon_k)) + const) \end{aligned} \tag{229}$$

with:

$$\hat{\boldsymbol{\Sigma}}_{\boldsymbol{h}_q^k} = \left(\boldsymbol{V}_F^H(\varepsilon_k).\boldsymbol{\Sigma}_{\mathbf{w}_q^k}^{-1}.\boldsymbol{V}_F(\varepsilon_k) + \boldsymbol{\Sigma}_{\boldsymbol{h}_q^k}^{-1}\right)^{-1} \tag{230}$$

and

$$\tilde{\boldsymbol{h}}_q^k = \hat{\boldsymbol{\Sigma}}_{\boldsymbol{h}_q^k}.\boldsymbol{V}_F^H(\varepsilon_k).\boldsymbol{\Sigma}_{\mathbf{w}_q^k}^{-1}.\hat{\boldsymbol{d}}_q^k(m) \tag{231}$$

Using (229), we propose two different ways to obtain the estimation of the frequency offset: they are named algorithm 1 and algorithm 2. The first one does not need estimation of users' channel parameters whereas the second one uses at each step the former estimation of the channel parameters to help enhance the estimation of the carrier frequency offset.

- **Algorithm 1 (FFT–based algorithm):** It is possible to further simplify the expressions given in (230),(231). We have:

$$\boldsymbol{\Sigma}_{\boldsymbol{h}_q^k} = E\left(\boldsymbol{h}_q^k.\left(\boldsymbol{h}_q^k\right)^H\right) = \text{diag}[(\sigma_{1,q}^k(1))^2,...,(\sigma_{1,q}^k(N_{CP}))^2,...,(\sigma_{N_t,q}^k(1))^2,...,(\sigma_{N_t,q}^k(N_{CP}))^2 \tag{232}$$

with: $(\sigma_{i,q}^k(j))^2 = E\left(\left|h_{i,q}^k(j)\right|^2\right)$. Concerning the computation of $\boldsymbol{\Sigma}_{\mathbf{w}_q^k}$, we have directly:

$$\boldsymbol{\Sigma}_{\mathbf{w}_q^k} = E\left(\mathbf{w}_q^k.\left(\mathbf{w}_q^k\right)^H\right) = \sigma_{\mathrm{w}_q^k}^2.\boldsymbol{I}_K \tag{233}$$

A great simplification is then made if we remark that the matrix product leads to diagonal matrices with value $N_c/N_t$ on the main diagonal. Using the approximation $\boldsymbol{V}_F^H(\varepsilon_k).\boldsymbol{V}_F(\varepsilon) \approx \frac{N_c}{N_t}.\boldsymbol{I}_{N_t.N_{CP}}$, we obtain first:

$$\hat{\boldsymbol{\Sigma}}_{\boldsymbol{h}_q^k} \approx \left(\frac{N_c}{N_t}.1/\sigma_{\mathbf{w}_q^k}^2.\boldsymbol{I}_{N_t.N_{CP}} + \boldsymbol{\Sigma}_{\boldsymbol{h}_q^k}^{-1}\right)^{-1} \approx \frac{N_t}{N_c}.\sigma_{\mathbf{w}_q^k}^2.\boldsymbol{I}_{N_t.N_{CP}} \quad \text{for classical SNR values} \tag{234}$$

and then, $\tilde{\boldsymbol{h}}_q^k \approx \left(N_t/N_c\right).\boldsymbol{V}_F^H(\varepsilon_k).\hat{\boldsymbol{d}}_q^k(m)$. Eventually, $Q_k(\varepsilon_k)$ is simplified as:

$$Q_k(\varepsilon_k) = -\sum_{q=1}^{N_r}\left(\left\|\hat{\boldsymbol{d}}_q^k(m) - \frac{N_t}{N_c}.\boldsymbol{V}_F(\varepsilon_k).\boldsymbol{V}_F^H(\varepsilon_k).\hat{\boldsymbol{d}}_q^k(m)\right\|^2 + \mathrm{trace}\left(\frac{N_t.\sigma_{\mathbf{w}_q^k}^2}{N_c}\boldsymbol{V}_F(\varepsilon_k).\boldsymbol{V}_F^H(\varepsilon_k)\right)\right) \tag{235}$$

The second term, $\mathrm{trace}\left(\frac{N_t.\sigma_{\mathbf{w}_q^k}^2}{N_c}\boldsymbol{V}_F(\varepsilon_k).\boldsymbol{V}_F^H(\varepsilon_k)\right)$, is independent of $\varepsilon_k$ and hence has no contribution to the detector. The ML algorithm for the determination of $\varepsilon_k$ is thus:

$$\hat{\varepsilon}_k(m+1) = \arg\min_{\varepsilon_k}\sum_{q=1}^{N_r}\left\|\left(\boldsymbol{I}_{N_t.N_{CP}} - \frac{N_t}{N_c}.\boldsymbol{V}_F(\varepsilon_k).\boldsymbol{V}_F^H(\varepsilon_k)\right).\hat{\boldsymbol{d}}_q^k(m)\right\|^2 \tag{236}$$

Expression (236) enables to obtain a rough guess of the frequency offset $\varepsilon_k$. However, it is possible to obtain a more accurate estimation by using a FFT-based search algorithm. Developing (236) and dropping terms irrelevant of $\varepsilon_k$, we obtain:

$$\begin{aligned}\hat{\varepsilon}_k(m+1) &= \arg\min_{\varepsilon_k}\sum_{q=1}^{N_r}\hat{\boldsymbol{d}}_q^k(m)^H.\boldsymbol{V}_F(\varepsilon_k).\boldsymbol{V}_F^H(\varepsilon_k).\hat{\boldsymbol{d}}_q^k(m)\\ &= \arg\min_{\varepsilon_k}\sum_{q=1}^{N_r}\left(\sum_{p=0}^{N_c-1}\sum_{r=0}^{N_c-1}\left[\hat{\boldsymbol{d}}_q^k(m)\right]_p^*\left[\boldsymbol{Q}_k\right]_{p,r}\left[\hat{\boldsymbol{d}}_q^k(m)\right]_r.e^{j.\varepsilon_k(p-r)}\right)\end{aligned} \tag{237}$$

with $\boldsymbol{Q}_k = \boldsymbol{A}_k(n).\boldsymbol{A}_k^H(n)$, (237) can be further written in the form:

$$\hat{\omega}_k(m+1) = \arg\min_{\varepsilon_k} \sum_{q=1}^{N_r} \{ \sum_{p=0}^{N_c-1} [\boldsymbol{Q}_k]_{p,p} . \left\| \left[ \hat{\boldsymbol{d}}_q^k(m) \right]_p \right\|^2 + ... \\ 2.\mathrm{Re}\left( \sum_{p=0}^{N_c-2} \sum_{r=p+1}^{N_c-1} [\boldsymbol{Q}_k]_{p,r} . \left[ \hat{\boldsymbol{d}}_q^k(m) \right]_p^H . \left[ \hat{\boldsymbol{d}}_q^k(m) \right]_r . e^{j.\varepsilon_k.(r-p)} \right) \} \tag{238}$$

The first term in (238) is constant. By denoting $n = r - p$, we can further simplify (238) to:

$$\hat{\varepsilon}_k(m+1) = \arg\ \min_{\varepsilon_k}\ 2.\mathrm{Re}\left( \sum_{q=1}^{N_r} \left\{ \sum_{n=1}^{N_c-1} r_q^k(n).e^{-j.n.\varepsilon_k} \right\} \right) \\ = \arg\ \min_{\varepsilon_k}\ 2.\mathrm{Re}\left( R^k(\varepsilon_k) \right) \tag{239}$$

where $R^k(\varepsilon_k) = \sum_{n=1}^{N_c-1} r^k(n).e^{-j.n.\varepsilon_k}$ and $r^k(n) = \sum_{q=1}^{N_r} \sum_{m=0}^{N_c-1-n} [\boldsymbol{Q}_k]_{m,m+n} \left[ \hat{\boldsymbol{d}}_q^k(m) \right]_m^* . \left[ \hat{\boldsymbol{d}}_q^k(m) \right]_{m+n}$.

$R^k(\varepsilon_k)$ can be easily computed using the FFT algorithm. When $\hat{\varepsilon}_k(m+1)$ has been obtained, it is straightforward, using (225) and (227), to estimate $\hat{\boldsymbol{h}}^k(m+1)$. We solve (225) by finding the LMS solution of each term in the sum, this yields to:

$$\hat{\boldsymbol{h}}_q^k(m+1) = \mathrm{pinv}(\boldsymbol{V}_F(\hat{\varepsilon}_k(m+1))).\hat{\boldsymbol{d}}_q^k(m) \\ = (\boldsymbol{V}_F^H(\hat{\varepsilon}_k(m+1)).\boldsymbol{V}_F(\hat{\varepsilon}_k(m+1)))^{-1}.\boldsymbol{V}_F^H(\hat{\varepsilon}_k(m+1)).\hat{\boldsymbol{d}}_q^k(m) \tag{240}$$

- **Algorithm 2 (joint CFO and channel estimation):** In the former algorithm the CFO was estimated independently of the channel parameters since $\boldsymbol{V}_F(\varepsilon_k)$ does not depend on them. In the new proposed algorithm we estimate the carrier frequency offset using channel parameters obtained at the former EM iteration steps. We suppose that we used algorithm 1 (FFT-based search method) for the first steps. Using the FFT based method enables to obtain a rough estimate of the channel parameters (237) and we found that reusing equation (229) by replacing $\tilde{\boldsymbol{h}}_q^k$ by the estimate $\hat{\boldsymbol{h}}_q^k(m)$ of $\boldsymbol{h}_q^k$ obtained at iteration step *m* yields to results very close to the theoretical Cramer-Rao bound. This original method proceeds as follows. We have now:

$$Q_k(\varepsilon_k) = -\sum_{q=1}^{N_r} (\left\| \hat{\boldsymbol{d}}_q^k(m) - \boldsymbol{V}_F(\varepsilon_k).\hat{\boldsymbol{h}}_q^k(m) \right\|^2 + const) \tag{241}$$

and developing $Q_k(\omega_k)$ with dropping the terms irrelevant of $\omega_k$, this gives the estimation rule:

$$\begin{aligned}\hat{\varepsilon}_k(m+1) &= \arg\max_{\varepsilon_k} \left\| \sum_{q=1}^{N_r} (\boldsymbol{V}_F(\varepsilon_k).\hat{\boldsymbol{h}}_q^k(m))^H .\hat{\boldsymbol{d}}_q^k(m) \right\|^2 \\ &= \arg\max_{\varepsilon_k} \left\| \sum_{q=1}^{N_r} e^{-j.\tilde{\varepsilon}_k} .\hat{\boldsymbol{h}}_q^k(m)^H .\boldsymbol{A}_k^H(n).\boldsymbol{E}(\varepsilon_k)^H .\hat{\boldsymbol{d}}_q^k(m) \right\|^2 \end{aligned} \quad (242)$$

Then, we use vector $\boldsymbol{\Lambda}_{k,q,n}$ of size $1\times N_c$ defined as: $\boldsymbol{\Lambda}_{k,q,n} = \hat{\boldsymbol{h}}_q^k(m)^H .\boldsymbol{A}_k^H(n)$ and we denote $\Lambda_{k,q,n}(p)$ as the $p$th element of this vector. (29) can be further developed as:

$$\hat{\varepsilon}_k(m+1) = \arg\max_{\varepsilon_k} \left\| e^{-j.\tilde{\varepsilon}_k} \sum_{q=1}^{N_r} \sum_{p=0}^{N_c-1} \Lambda_{k,q,n}(p).e^{-j.p.\varepsilon_k} .\left[\hat{\boldsymbol{d}}_q^k(m)\right]_p \right\|^2 \quad (243)$$

or: $\hat{\varepsilon}_k(m+1) = \arg\max_{\varepsilon_k} \left\| \sum_{p=0}^{N_c-1} u(p).e^{-j.p.\varepsilon_k} \right\|^2$ with $u(p) = e^{-j\tilde{\varepsilon}_k} \sum_{q=1}^{N_r} \Lambda_{k,q,n}(p).\left[\hat{\boldsymbol{d}}_q^k(m)\right]_p$ Denoting $\chi(p) = \sum_{s=0}^{N_c-1-p} u(s).u^*(p+s)$, the search procedure for $\varepsilon_k$ is then obtained as:

$$\begin{aligned}\hat{\varepsilon}_k(m+1) &= \arg\max \ \mathrm{Re}\left( \sum_{p=1}^{N_c-1} \chi(p).e^{-j2\pi p.\varepsilon_k} \right) \\ &= \arg\max(\Psi(\varepsilon_k)) \end{aligned} \quad (244)$$

with $\Psi(\omega_k) = \mathrm{Re}\left( \sum_{p=1}^{N_c-1} \chi(p).e^{-j2\pi p.\varepsilon_k} \right)$. Setting the derivative of $\Psi(\varepsilon_k)$ to zero, we obtain:

$$\frac{\partial \Psi(\varepsilon_k)}{\partial \varepsilon_k} = \mathrm{Im}\left( \sum_{p=1}^{K-1} p.\chi(p).e^{-j2\pi p\varepsilon_k} \right) = 0$$

or, equivalently: $\sum_{p=1}^{N_c-1} p.|\chi(p)|.\sin(\arg(\chi(p) - 2\pi p.\varepsilon_k)) = 0$. To obtain a close form solution, we need to use the approximation $\sin(x) \approx x$, this implies that the initial guessed value for $\varepsilon_k$ obtained by the FFT-based search algorithm is sufficiently close to the true value. In this case, we eventually obtain:

$$\hat{\varepsilon}_k(m+1) = \frac{1}{2\pi} \frac{\sum_{p=1}^{N_c-1} p.|\chi(p)|.\arg(\chi(p))}{\sum_{p=1}^{K-1} p^2.|\chi(p)|} \quad (245)$$

$\hat{\boldsymbol{h}}^k(m+1)$ is then obtained in the same way as given in equation (27). The advantage of the EM algorithm is that its complexity grows linearly with the number of users and the maximization step can be accomplished in parallel for all $N_u$ users. However, such algorithm exhibits some known drawbacks. The first one concerns its slow convergence rate since it is inversely related to

the Fisher information of the complete data space (113). The second one is the introduction of free variables $\beta_k$'s. Inappropriate values of $\beta_k$'s may not only lead to slower convergence rate but also may give birth to convergence to local stationary points. In fact, it seems that no accurate rules for the choice of $\beta_k$'s exist at the moment. For the presented simulation results we found that $\beta_k$ constitutes the best trade-off between accuracy and convergence rate.

### 7.3.2 Using SAGE algorithm

The space-alternating generalized expectation-maximization (SAGE) algorithm was proposed in [126][127] to overcome the drawbacks of the conventional EM algorithm. The advantage of this new algorithm is that instead of optimizing over all "complete" data $\boldsymbol{d}_q^k$'s simultaneously in each iteration, we consider one $\boldsymbol{d}_q^k$ per iteration, for $k = 1,..., N_u$ sequentially, and associate all noise with the current $\boldsymbol{d}_q^k$. The complete SAGE algorithm in the $m^{\text{th}}$ iteration step is given just below.

- ***E-step* (*Expectation Step*)**: For $k = 1,\ldots,N_u$, $p = 1,\ldots,N_t$ and $q = 1,..., N_r$, compute

$$\hat{\boldsymbol{b}}_{p,q}^k(m) = \boldsymbol{G}_p^k(\hat{\varepsilon}_k(m)).\boldsymbol{h}_{p,q,\mu_{p,q}^k}^k(m) \tag{246}$$

$$\hat{\boldsymbol{d}}_{p,q}^k(m) = \hat{\boldsymbol{b}}_{p,q}^k(m) + \beta_k\left[\boldsymbol{y}_{p,q} - \sum_{j=1}^{N_u}\hat{\boldsymbol{b}}_{p,q}^j(m)\right] \tag{247}$$

$$\hat{\boldsymbol{d}}_q^k(m) = \sum_{p=1}^{N_t}\hat{\boldsymbol{d}}_{p,q}^k(m) \tag{248}$$

- M-step (Maximization Step): Compute

$$[\hat{\varepsilon}_i(m+1), \hat{\boldsymbol{h}}^i(m+1)] = \arg\min_{\varepsilon_i, \boldsymbol{h}^i}\left\{\sum_{q=1}^{N_r}\left|\hat{\boldsymbol{d}}_q^i(m) - \sum_{p=1}^{N_t}\boldsymbol{G}_p^i(\varepsilon_k).\boldsymbol{h}_{p,q,\mu_{p,q}^i}^i\right|^2\right\} \tag{249}$$

and for $k \neq i$

$$[\hat{\varepsilon}_k(m+1), \hat{\boldsymbol{h}}^k(m+1)] = [\hat{\varepsilon}_k(m), \hat{\boldsymbol{h}}^k(m)] \tag{250}$$

Equation (249) can be optimized using Algorithm 1 and Algorithm 2 already mentioned in section 7.3.1.

### 7.3.3 Initialization Strategies

EM-based algorithms do not always converge to the global maximum point particularly if there exists multiple stationary points. To avoid this, one can think of two initialization strategies.

1- Since, for classical OFDM systems, the absolute value of the carrier frequency offset (CFO) is bounded by half of the OFDM subcarrier spacing, we can assume that $\varepsilon_k$ is a zero-mean uniformly distributed random variable and so we take $\hat{\varepsilon}_k(0) = 0$. Then, $\hat{\boldsymbol{h}}_q^k(0)$ is obtained by substituting $\hat{\varepsilon}_k(0)$ in equation (27) with $q = 1,\ldots,N_r$.

2- We use the MLE estimation method proposed by Morelli in [136] for OFDMA systems. By regarding all other users' signals as noise, we make a rough guess on $\hat{\varepsilon}_k(0)$ and $\hat{\boldsymbol{h}}_q^k(0)$ for $k = 1,...,N_u$. However, we found in our simulations that this method yields to poor results and so we have only retain strategy 1.

### 7.3.4 Algorithms complexity

To evaluate the complexity of our algorithms, the number of complex multiplication needed to implement them could be calculated.

- For the first EM algorithm, we proceed first with a FFT based search method for CFO estimation. The needed number of multiplication is equal to $\vartheta\left((N_c^2 - N_c)N_r + 2R\log R\right)$, where $R$ is the length of FFT. The accuracy of CFO estimation in this algorithm is limited to FFT length and better frequency resolution can be achieved by zero-padding to effectively increase FFT length.

For the second algorithm within the EM loop, that is joint CFO and channel estimation, the number of complex multiplication needed at each iteration is equal to $\vartheta\left(N_r N_t LK + \frac{(N_c^2 + N_c)}{2}\right)$.

So we find :

- $\vartheta\left(N_u.((N_c^2 - N_c)N_r + 2R\log R)\right)$ multiplications per iterations for CFO estimation with FFT.
- $\vartheta\left((N_r N_t LN_c + \frac{(N_c^2 + N_c)}{2}).N_u\right)$ multiplications per iterations for CFO estimation with the joint CFO-channel estimation.

To calculate channel coefficients we need to calculate pseudoinverse of a matrix $N_c \times N_t N_g$ and to calculate pseudoinverse of a matrix we use the SVD decomposition of that matrix. The complexity of calculating SVD of such a matrix in the worst case is equal to $\vartheta\left(N_u.\min\left(N_c \times N_t{}^2 N_{CP}{}^2, N_c{}^2 \times N_t N_g\right)\right)$. Therefore, the complexity of calculating $\boldsymbol{h}$ is equal to : $\vartheta\left(N_u.(\min\left(N_c N_t{}^2 N_{CP}{}^2, K^2 N_t N_g\right) + K N_t N_{CP})\right)$.

- For the SAGE algorithm, the big difference is that we have to update only one user's parameters at each iteration. So, the results obtained for EM still holds omitting $N_u$.

As it can be seen, our algorithm requires a heavy computational burden. However, we achieve an accurate synchronizer for an OFDMA system which employs general carrier assignment scheme.

# 7.4 Simulation results

The OFDMA system we used in our simulation is a simplified version of IEEE 802.11a. We consider an OFDMA system with QPSK modulation and $N_c$ = 128 carriers, $N_{CP}$ 18, $N_u$ = 4 users. Each user has 32 carriers randomly assigned and $N_t$ = 2 transmit antennas. In the simulation runs we will deal with the case where the BS station is equipped with $N_r = 2$ receive antenna. We consider here MIMO independent frequency selective channels for each link between one transmit and one receive antenna at the BS. They are modelled as typical GSM urban quasi-static channels and we use here the models provided in [122][123]. With such kind of channel, it is easy to show that $h_{p,q}^k(l)$ only takes significant values for $l \le 7$, which means $L_{p,q}^k = 8, \forall k, \forall p, \forall q$ for our signal model (see equation (215)). For the delays $\mu_{p,q}^k$ we use uniform random laws with maximum value $\mu_{p,q\max}^k = N_g - L_{p,q\max}^k = 18 - 8 = 10$. For the LDPC channel encoder, we use a (3,6) regular codes with parity check matrix of 512 × 1024, i.e. with coding rate 0.5. The packet size is equal to 512 QPSK symbols and the first OFDM word contains pilot symbols. Random simulated channels are quasi-static ones i.e. they remain constant over the time duration of a packet. For the generation of CFO we use a random uniform law with maximum value $2.\pi/128 \approx 0.05$, hence a random value $\varepsilon_k$ is generated from the interval [-0.31,0.31].

The structure of the receiver for each user is similar to those presented in [140] except that we incorporate the joint CFO-channel algorithm presented in previous section at the front end of the base-station receiver. Hence, we obtain two iterative structures: the first one is our EM or SAGE based joint CFO-channel estimation algorithm of section 7.3 and the second one is the joint MAP-EM LDPC decoder of B.Lu and Wang. Our joint EM based CFO-channel estimation algorithm iterates ten times: the first five steps are done with the FFT based search algorithm (Algorithm 1 of section 7.3.1) with a FFT size of 512 and the remaining ones use the joint CFO-channel estimation procedure (Algorithm 2 of section 7.3.1). After this first stage we enter the MAP-EM LDPC decoder circuit with the resulting CFO and channel estimates. We use five inner iterations for the EM MAP demodulator and thirty inner iterations within the LDPC decoder [140] Five complete iterations in the MAP-EM LDPC decoder structure are made before we compute the BER. The performance of our synchronization algorithms is first evaluated with the mean square error (MSE) of the estimated parameters. Without loosing any generality, we will consider user 1 as the user of interest.. Then, in order to quantify the efficiency of the proposed algorithms, we can use the mean CRB's of $\Delta f_1$ and $\boldsymbol{h}^1$. In the following simulation result we consider a situation in which $N_u = 4$ users want to enter the system simultaneously.

At first we consider the case where the mean power of each user is the same, equal to $1/N_t$. The MSE is averaged for each SNR over 1000 packet Monte-Carlo runs. For the initialization strategy of the EM algorithms, we use the first one described in section 7.3.3 . The simulation results are presented on Figure 53 for $N_r$ = 2 receive antennas. We add simulation results for comparison purpose from the simple algorithm of Morelli presented in [136]. EM and SAGE algorithms exhibit the same performances and they work very close to the mean Cramer Rao Bound (mean CRB) $\overline{CRB(\varepsilon_1)}$: the difference between SAGE or EM and the mean CRB is less than 0.5 dB. Morelli's algorithm [136] leads to poorer performance and exhibits a MMSE floor at high SNR's due to the fact that it was originally devoted to mono-user systems.

We consider a near-far effect on Figure 53 with $N_r$ = 2 receive antennas when user 1's average signal power is 3 dB lower than the other users. It is clear in this case that only the EM based methods can lead to satisfactory results. Morelli's algorithm can't accommodate the strong interference power from the other users and exhibits no improvement by increasing the SNR. EM based algorithms perform within 0.7 dB from the mean CRB: $\overline{CRB(\varepsilon_1)}$.

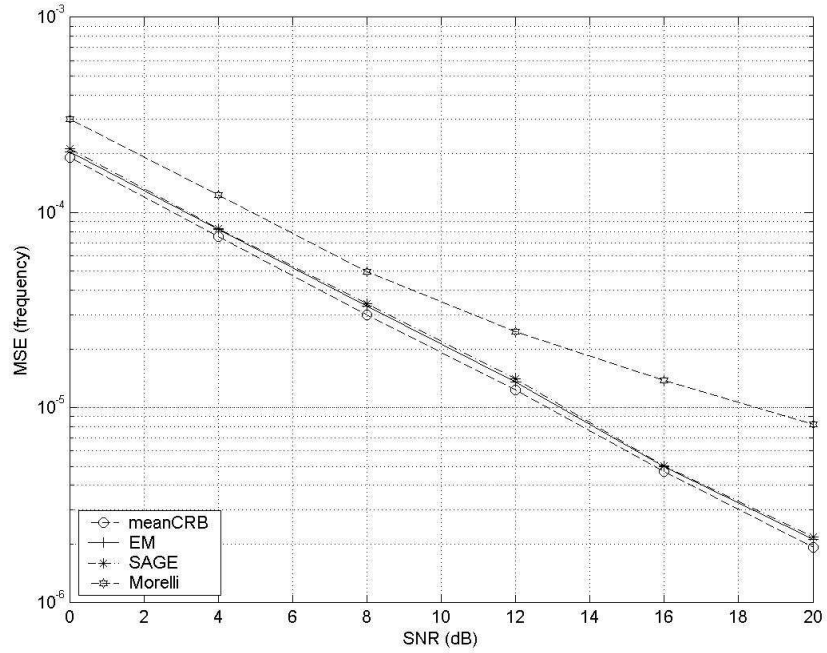


Figure 53:CFO estimation performance with equal power assignment

$N_r = 2$, two receive antennas

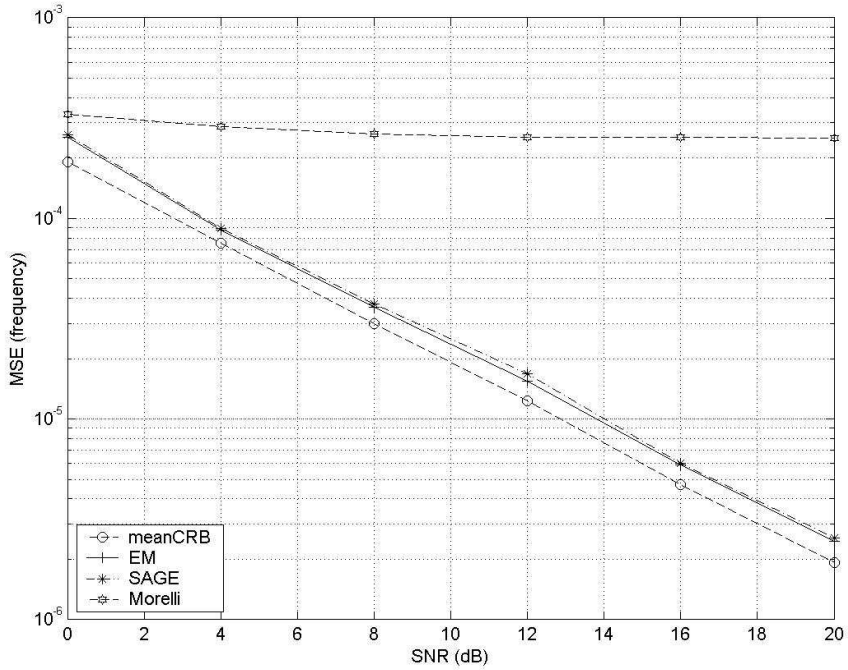


Figure 54: CFO estimation performance in a near-far situation

$N_r = 2$, two receive antennas

To illustrate the performances of the proposed EM algorithms in the case of channel estimation, we give the MSE performances in the context of the near-far effect. The results are plotted on Figure 55. One can remark that the two proposed algorithms perform extremely close to each other. They work within 1 dB from the mean CRB $\overline{CRB(\boldsymbol{h}^1)}$.

Figure 56 examines the convergence behaviour of the proposed iterative algorithms in the near-far context. We define a convergence threshold for the square error of the CFO equal to $10^{-3}$. That means all the packets having a CFO square error superior to $10^{-3}$ are considered rejected. We compute the ratio of the number of correct received packets over the total number of transmitted packets, this gives the success rate illustrated on Figure 56. One can remark on Figure 56 a better success rate for EM algorithm compared to SAGE. That means that despite a slower convergence, EM is less likely to encounter data blocks that cause divergence. However, the difference between the two curves is always inferior to ten percent.

The similar behaviour of EM and SAGE is confirmed by the study of the BER on Figure 57. The performances exhibited by the EM algorithm are slightly better than those of the SAGE algorithm (the difference is 0.3 dB at BER = $10^{-4}$). On the other hand, the simple Morelli's algorithm [136] exhibits an error floor in this case. When compared to EM-SAGE algorithms the loss is approximately equal to 2 dB at BER = $10^{-3}$.

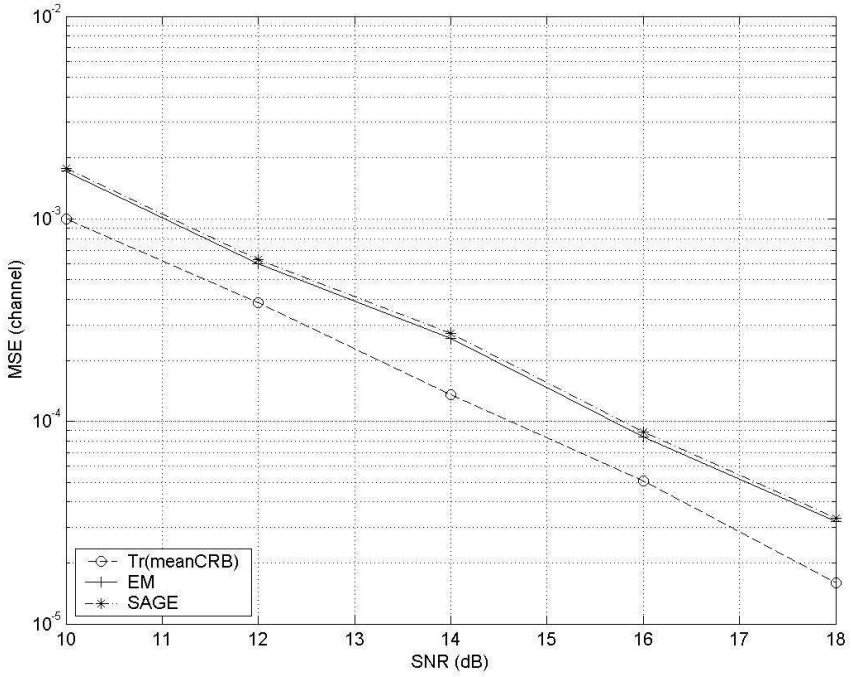


Figure 55: Channel estimation performance in a near-far situation

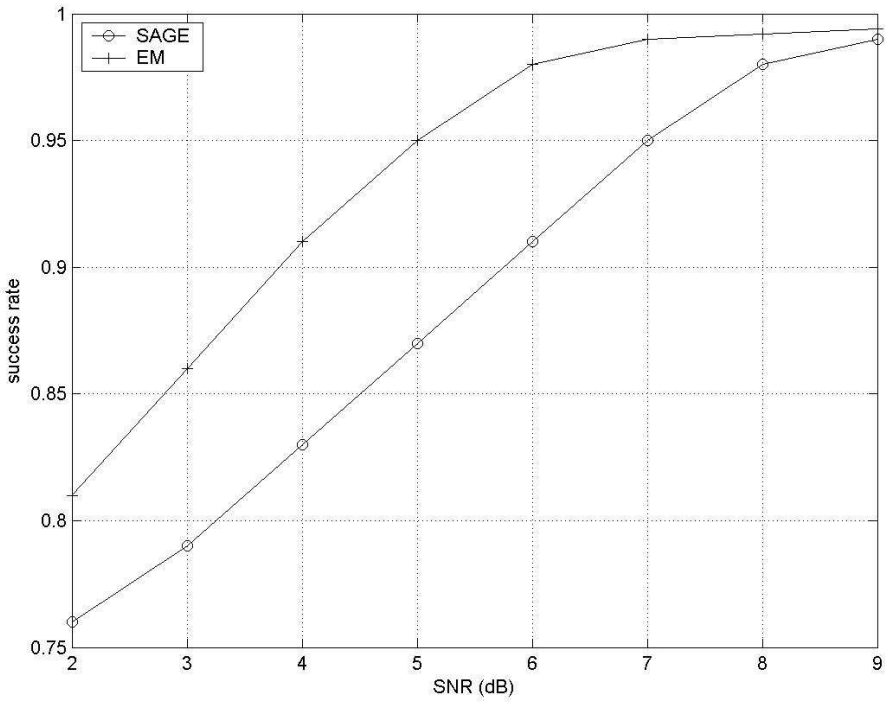


Figure 56: Success rate for the transmitted packets for EM and SAGE algorithms near-far situation (Decion Directed mode), $N_r = 2$, two receive antennas

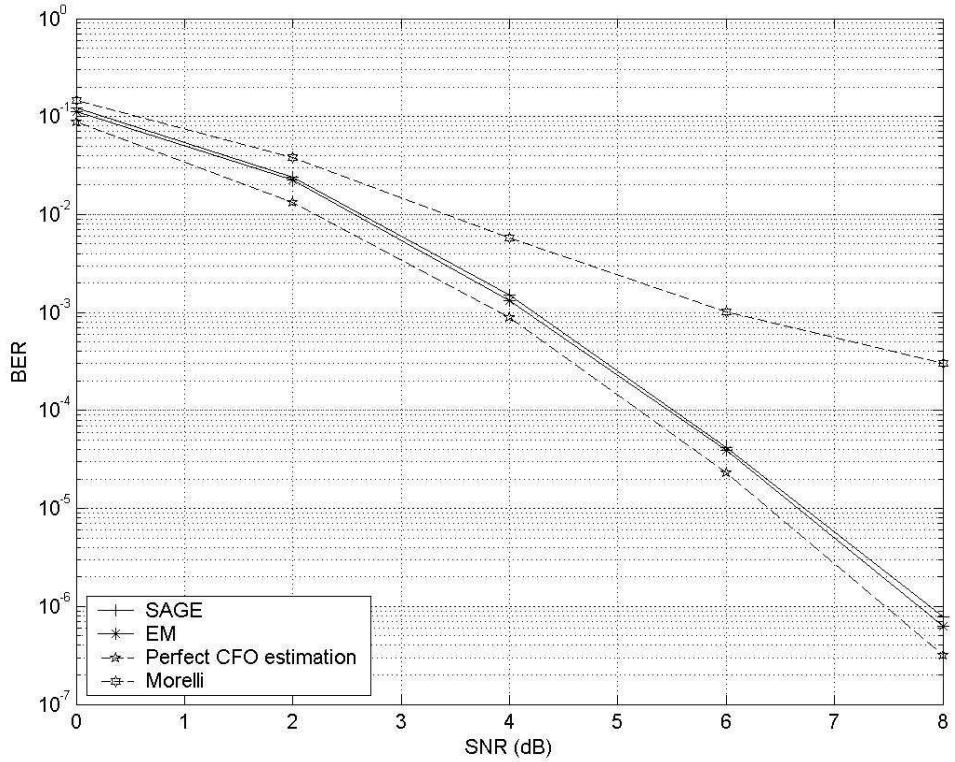


Figure 57:BER performance comparison between synchronisation algorithms

near-far situation (DD mode), $N_r = 2$, two receive antennas

# Chapter 8

# CONCLUSION

The problem of synchronization in MIMO-OFDM systems is addressed in this dissertation. Apart from introduction and conclusion, this manuscript is composed of six main chapters.

At first, in chapter 2 the problem of channel estimation was discussed. Since the channel estimation is not in the center of attention of this dissertation, this chapter is, in fact, a literature survey of works dealing with channel estimation methods, especially data aided channel estimation methods. Therefore, some good data aided channel estimation methods were presented and within the data aided methods two different cases were distinguished, one was the case where we use a training sequence in the beginning of the data packet and the other was the case where we insert known pilots inside the data packet. Chapter 2 finishes with a quick look at blind and iterative channel estimation methods.

Subsequently, chapter 3 deals with the problem of the frequency synchronization in MIMO-OFDM systems. The aim of this chapter is to estimate the carrier frequency offset induced either by a small difference in frequency of transmitter and receiver oscillators or by the Doppler effect. In this chapter three of the best existing methods for frequency synchronization, which were called FFT method, hopping pilots method and autocorrelation method, were presented. Along with these three methods another new method (so called iterative method) developed by us was

presented. This method is, in fact, a part of the joint algorithm presented in chapter 5. Hence chapter 3 only attempts to outline the section of frequency estimation of the joint algorithm. Then, we compared these four methods and concluded that the autocorrelation method and iterative method outperform two other algorithms. Iterative method shows a slightly better performance with higher complexity. It has to be mentioned that since the iterative method is a part of joint method presented in chapter 5 the complete simulation of this method as well as other methods along with the simulation is presented in chapter 5.

Chapter 4 addresses the problem of time synchronization. Since MIMO-OFDM time synchronization approaches are influenced by non MIMO−OFDM time synchronization methods, Chapter 4 starts with a review of non MIMO−OFDM time synchronization methods designed for either MIMO non-OFDM or non-MIMO OFDM systems. Then, the problem of time synchronization in MIMO-OFDM systems was discussed. The problem of time synchronization is divided into frame synchronization and symbol timing and each issue was separately addressed. Concerning frame synchronization, the conventional method of autocorrelation was presented first and then we proposed some modifications to make this algorithm more precise. The results are presented under the title of advanced autocorrelation method. The advanced autocorrelation method was originally designed for flat fading channels however it presents good results for frequency selective channels. Concerning symbol timing, some of well known methods were presented. In addition, a new symbol timing method developed by us was discussed.

Chapter 5 is about a new joint time-frequency synchronization and channel estimation method for MIMO-OFDM systems. In this chapter we developed an ML criterion for MIMO-OFDM synchronization. Maximizing this criterion leads to find the exact packet arrival time as well as carrier frequency offset and channel coefficients. Since, the estimation of frequency offset by numerical search is almost impossible we have proposed a two-stage algorithm to find analytically frequency offset. At the first stage we do frequency synchronization by using an FFT algorithm. This rough estimation permits us to compensate the main part of frequency error and so at this level time synchronization can be done. Then, at the second stage, we propose an iterative method to compensate the residual frequency error. Chapter 5 finishes with presenting the result of this algorithm and comparing it with other algorithms. It is concluded at the end of

this chapter that the best synchronizer would be either the joint algorithm presented in this chapter or a mixed synchronizer that use autocorrealtion method for frequency synchronization and frame synchronization and reduced complexity method (presented in chapter 4) for symbol timing. The complexity of the latter is less while the performance of the former is slightly better.

Chapter 6 attempts to present the concept of code-aided algorithm. Actually, by using code-aided algorithm data portion of the frame can be used for synchronization and that leads to improve the spectral efficacy as well as the possibility of finding the possible changes in the estimation parameters. In this chapter we use soft information provided from the decoding process, in order to fully exploit the code properties during estimation. EM algorithm as a tool toward code aided algorithm is described in this chapter and then we have explained how one can use EM algorithm for MIMO-OFDM synchronization. The results are depicted at the end of the chapter.

MIMO OFDM systems have given birth to OFDMA systems. In the case of OFDMA, the set of available carriers is divided into groups called subchannels and each user is assigned with one or multiple subchannels. Hence, we dedicated the perspective chapter to some suggestion about MIMO OFMA synchronization. Actually we proposed a synchronization procedure based on the EM algorithm. Here, we used EM-based approach to decompose a computationally intensive multi-dimensional optimization problem into separate smaller ML optimization problems. The result of this chapter shows acceptable performance. However, we have supposed that the different transmit antennas of each user send their training data sequentially in the network i.e. the other transmit antennas are switched off. This implies a quasi-synchronous system in a micro-cell environment for example. Also, the correction of one's frequency and timing cannot be accomplished at the Base Station. Hence, this problem can be still considered as an open problem which demands further works.

# REFERENCES

***Author's publications***


[1] Amir Saemi, Jean-Pierre Cances, Vahid Meghdadi, "Synchronization Algorithms for MIMO OFDMA Systems", Accepted by *IEEE transaction on Wireless Communications* March 2007.

[2] Amir Saemi, Vahid Meghdadi, Jean-Pierre Cances, Mohammad Reza Zahabi, "Joint ML time-frequency synchronization and channel estimation algorithm for MIMO-FDM systems", invited by *IET Proceeding of Circuit, Devices and Systems*.

[3] Amir Saemi, vahid Meghdadi, Jean-Pierre Cances, and Mohammad Reza Zahabi, "Iterative time-frequency synchronizer for MIMO-OFDM systems", submitted in *EURASIP Journal on Wireless Communications and Networking* (EURASIP JWCN).

[4] Amir Saemi, Vahid Meghdadi, Jean-Pierre Cances, M.Reza Zahabi, Jean-Michel Dumas, "ML Time-Frequency synchronization for MIMO-OFDM Systems in Unknown Frequency Selective fading channels.", *PIMRC* 2006.

[5] Amir Saemi, Vahid Meghdadi, Jean-Pierre Cances, Mohammad Reza Zahabi, Jean-Michel Dumas,"ML Symbol Synchronization For General MIMO-OFDM Systems In Unknown Frequency-Selective Fading Channels", *EUSIPCO* 2006.

[6] Amir Saemi, Vahid Meghdadi, Jean-Pierre Cances, M.Reza Zahabi, Jean-Michel Dumas, "ML Time Synchronization Algorithm Joint with Channel Estimation for MIMO-OFDM Systems in Frequency Selective Fading Channel" *CSNDSP* 2006, ISBN: 960-89282-0-6.

[7] V. Meghdadi, A. Saemi, J.P. Cances, M.J. Syed, G. Ferre; J.M. Dumas, "Improving frequency synchronization by means of new correlation criteria for fine time synchronization in MIMO system**",** *ELMAR, 2005. 47th International Symposium*, pp 283-286.

[8] A. Saemi, V. Meghdadi, JP. Cances, et al. "Fine Timing and Frequency Synchronization for MIMO System," *Proc. of IST Mobile and Wireless Communications Summit*, 2005.

[9] Amir Saemi, Vahid Meghdadi, Jean-Pierre Cances, "MIMO-OFDM iterative time-frequency synchronization", will be published in *Proc PIMRC* 2007, Sept. 2007.

[10] A. Saemi, G. Ferré, J. P. Cances and V. Meghdadi, "Synchronization algorithms for MIMO-OFDMA systems", will be published in *Proc PIMRC* 2007, Sept. 2007.

[11] M.R. Zahabi, V. Meghdadi, J.P. Cances, A. Saemi, J.M. Dumas, B. Barelaud, "Analog CMOS Kernel for ML Decoding of Convolutional Codes" *Proc. ICEE* 2006.

[12] M. R. Zahabi, V. Meghdadi, J.P. Cances and A. Saemi, "A Mixed-signal matched-filter design and simulation", will be published in *Proc of the International Conference on Digital Signal Processing (DSP)*, July 2007.

[13] M. R. Zahabi, V. Meghdadi, J.P. Cances and A. Saemi, "Mixed-signal realization of matched-filters for high rate communication systems", will be published in *Proc. PIMRC* 2007, Sept. 2007.

[14] Ferre, G.; Cances, J.P.; Meghdadi, V.; Dumas, J.M.; Saemi, A.; **"**Layered Space-Time Coding for 3n Transmit Antenna Communication Systems", *Proc. PIMRC* 2006.

***Other references***

[15] Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: High-Speed Physical Layer in the 5 GHz Band, *IEEE Standard 802.11a*-1999.

[16] Local and Metropolitan Area Networks—Part 16, Air Interface for Fixed Broadband Wireless Access Systems, *IEEE Standard IEEE 802.16a*.

[17] L. J. Cimini, Jr., "Analysis and simulation of a digital mobile channel using orthogonal frequency division multiplexing," *IEEE Trans. Commun*., vol. COM-33, pp. 665–675, July 1985.

[18] S. Alamouti, "A simple transmit diversity technique for wireless communications," *IEEE J. Select. Areas Commun.*, vol. 16, pp. 1451–1458, Oct. 1998.

[19] V. Tarokh, H. Jafarkhani, and A. R. Calderbank, "Space–time block codes from orthogonal designs," *IEEE Trans. Inform. Theory,* vol. 45, pp. 1456–1467, July 1999.

[20] V. Tarokh, N. Seshadri, and A. R. Calderbank, “Space–time codes for high data rate wireless communication: Performance criterion and code construction,” *IEEE Trans. Inform. Theory*, vol. 44, pp. 744–765, Mar. 1998.

[21] P. W. Wolniansky, G. J. Foschini, G. D. Golden, and R. A. Valenzuela, “V-Blast: An architecture for realizing very high data rates over the rich-scattering channel,” in *Proc. Int. Symp. Signals, Systems and Electronics* (ISSE 1998), pp. 295–300.

[22] Fredrik Tufvesson, Mike Faulkner and Ove Edfors, "Time and frequency synchronization for OFDM using PN-sequence preambles", *Proceedings of IEEE Vehicular Technology Conference*, Amsterdam, The Netherlands, September 19-22, 1999, pp. 2203-2207

[23] M. Schmidl and D. C. Cox, “Robust frequency and timing synchronization for OFDM,” *IEEE Trans. Commun.*, vol. 45, pp. 1613-1621, Dec. 1997.

[24] P.Moose, “A technique for orthogonal frequency division multiplexing frequency offset correction,” *IEEE Trans. Commun.*, vol. 42, pp. 2908-2914, Oct. 1994.

[25] S. H. Müller-Weinfurtner, “On the optimality of metrics for coarse frame synchronization in OFDM: A comparison,” *in Proc. PIMRC* , Sept. 1998, pp. 533-537.

[26] J.J. van de Beek, M. Sandell, and P.O. Borjesson,”Timing and frequency synchronization in OFDM system using the cyclic prefix”, *In Proceedings of International Symmposium on Synchronization*, pp. 16-19, Essen,Germany, December 1995.

[27] E.G. Larsson, G. Liu, J. Li, and G. Giannakis, “Joint symbol timing and channel estimation for OFDM based WLANs," *IEEE Communications Letters*, Vol. 5, No. 8, pp. 325-327, August 2001.

[28] Yik-Chung Wu, Kun-Wah Yip, Tung-Sang Ng and E. Serpedin, “ML Symbol Synchronization for OFDM-based WLANs in Unknown Frequency-Selective Fading Channels,” Proceedings of the 38th *Asilomar Conference on Signals, Systems and Computers*, pp. 339-344, Nov. 7-10, 2004.

[29] H. Minn, M. Zeng, and V. K. Bhargava, “On timing offset estimation for OFDM systems,” *IEEE Commun. Lett.*, vol. 4, pp. 242-244, July 2000.

[30] N. Lashkarian and S. Kiaei, “Class of cyclic-based estimators for frequency-offset estimation of OFDM systems,” IEEE *Trans. Commun*., vol. 48, pp. 2139-2149, Dec. 2000.

[31] A. J. Coulson, “Maximum likelihood synchronization for OFDM using a pilot symbol: Algorithm,” *IEEE J. Select. Areas Commun.*, vol. 19, pp. 2486-2494, Dec. 2001.

[32] A. J. Coulson, “Maximum likelihood synchronization for OFDM using a pilot symbol: Analysis,” *IEEE J. Select. Areas Commun.*, vol. 19, pp. 2495-2503, Dec. 2001.

[33] Speth, M. Classen, F. Meyr, H. “Frame synchronization of OFDM systems in frequency selective fadingchannels”, *Vehicular Technology Conference*, 1997 IEEE 47th,Vol 3, pp.1807-1811, May 1997

[34] Speth M, Fechtel S A, et al. “Optimum receiver design for OFDM-based broadband transmission .part II. A case study.” *IEEE Trans. Communication*, vol. 49(4): pp 571-578, Apr. 2001

[35] T. Keller and L. Hanzo, “Orthogonal frequency division multiplex synchronization techniques for wireless local area networks,” in *Proc. PIMRC* Taipei, Taiwan R.O.C., 1996, pp. 963-967.

[36] Pierre R. Chevillat, Dietrich Maiwald, and Gottfried Ungerboeck, “Rapid Training of a Voiceband Data–Modem Receiver Employing an Equalizer with Fractional–T Spaced Coefficients,” *IEEE Trans. on Commun*., vol. 35, no. 9, pp. 869–876, 1987

[37] K. Shi and E. Serpedin, “Robust Coarse Frame and Carrier Synchronization for OFDM Systems: A New Metric and Performance Evaluation" *IEEE Trans. on Wireless Communications*, vol. 3, no. 3, pp. 1271-1284, July 2004.

[38] R. van Nee and R. Prasard, *OFDM for Wireless Multimedia Communications*, Norwell, MA: Artech House, 2000.

[39] Nandula, S. Giridhar, K, “Robust timing synchronization for OFDM based wireless LAN system”, *TENCON* 2003. *Conference on Convergent Technologies for Asia-Pacific Region*, Vol 4,pp. 1558- 1561, Oct. 2003

[40] Wang, K., Faulkner, M., Singh, J. and Tolochko, I., “Timing Synchronization for 802.11a WLANS under Multipath Channels”, *Australian Telecommunications Networks & Applications Conference*, Melbourne, Australia, 8 - 10 December, 2003.

[41] Kun-Wah Yip, Yik-Chung Wu and Tung-Sang Ng, “Timing-synchronization analysis for IEEE 802.11a wireless LANs in frequency-nonselective Rician fading environments,” *IEEE Transactions on Wireless Communications*, vol. 3, no. 2, pp.387-394, Mar. 2004.

[42] Yik-Chung Wu, Kun-Wah Yip, Tung-Sang Ng and E. Serpedin, “Maximum-Likelihood Symbol Synchronization for IEEE 802.11a WLANs on Unknown Frequency-Selective Fading Channels," *IEEE Trans. on Wireless Communications*, (submitted July 2003, revised Jan. 2004, June 2004, accepted Aug. 2004).

[43] M. Speth, D. Daecke and H. Meyr, “Minimum overhead burst synchronization for OFDM based broadband transmission,” Proc. *Globecom* 98, pp. 3227-3232, 1998

[44] H. Sampath, S. Talwar, J. Tellado, V. Erceg, and A. Paulraj, “A fourth-generation MIMO-OFDM: Broadband wireless system: Design, performance, and field trial results,” *IEEE Commun. Mag.*, vol. 40, no. 9, pp. 143-149, Sept. 2002.

[45] A. N. Mody and G. L. Stüber, “Synchronization for MIMO OFDM systems,” *in Proc. IEEE Global Commun*. Conf., vol. 1, Nov. 2001, pp. 509-513.

[46] Suehiro N. and Hatori M., “Modulatable Orthogonal Sequences and their Application to SSMA Systems”, *Trans. on Inform. Theory*, 34, 93–100, 1998.

[47] A. van Zelst and T. C. W. Schenk, “Implementation of a MIMO OFDM- based Wireless LAN System,” *IEEE Trans. on Signal Processing*, vol. 52, no. 2, pp. 483–494, Feb. 2004.

[48] N.Duc Long and H. Park ,” Joint Fine Time Synchronization and Channel Estimation for MIM0-OFDM WLAN”, *IEEE Intelligent Signal Processing and Communication Systems Conference* (ISPACS 2004).

[49] En Zhou, Xing Zhang, Hui Zhao and Wenbo Wang, “Synchronization algorithms for MIMO OFDM systems”, *Wireless Communications and Networking Conference, IEEE***,** Vol. 1, pp. 18- 22, March 2005.

[50] Ayman F. Naguib, Vahid Tarokh, Nambirajan Seshadri, and A. Robert Calderbank, “A space-time coding modem for high-data-rate wireless communications,” *IEEE J. Selected Areas Commun*., vol. 16, no. 8, pp. 1459–1478, Oct. 1998.

[51] Yik-Chung Wu, S. C. Chan and E. Serpedin, “Symbol-Timing Estimation in Space-Time Coding Systems based on Orthogonal Training Sequences," *IEEE Trans. on Wireless Communications*, vol. 4, no. 2, pp. 603-613, Mar. 2005.

[52] M. Oerder and H. Meyr, “Digital filter and square timing recovery”, *IEEE Trans. on Comm*., vol.36, pp. 605-612, May 1988

[53] Yik-Chung Wu and Shing-Chow Chan, “On the Symbol Timing Recovery in Space-Time Coding Systems,” Proceedings of the *IEEE Wireless Communications and Networking Conference* (WCNC) 2003, pp.420-424, Mar. 2003.

[54] Yik-Chung Wu and E. Serpedin, “Symbol Timing Estimation in MIMO Correlated Flat-Fading Channels,”" *Wireless Communications and Mobile Computing (WCMC) Journal* , Special Issue on "Multiple-Input Multiple-Output (MIMO) Communications", Wiley, vol. 4, Issue 7, pp. 773-790, Nov. 2004.

[55] Schumacher L, Kermoal JP, Frederiksen F, Pedersen KI, Algrans A, Mogensen PE. MIMO channel characterisation. Deliverable D2 V1.1 of IST-1999-11729 METRA Project, pp.1–57, February 2001. Available online: http://www.ist-metra.org

[56] Yik-Chung Wu and E. Serpedin, “Training Sequences Design for Symbol Timing Estimation in MIMO Correlated Fading Channels,” Proceedings of the *IEEE Globecom* 2004, vol. 1, pp. 81-85, Nov. 2004.

[57] Yik-Chung Wu and E. Serpedin, “Low-complexity feedforward symbol timing estimator using Conditional Maximum Likelihood principle,” *IEEE Communications Letters*, vol. 8, no. 3, pp.168-170, Mar. 2004

[58] Y. Liu, T. F. Wong, and A. Pandharipande, "Timing Estimation in Multiple-Antenna Systems over Rayleigh Flat-Fading Channels," *IEEE Transactions on Signal Processing*, June 2004. Accepted for publication

[59] S.N. Diggavi, N. Al-Dhahir et al “Differential Space-Time Transmission for Frequency Selective Channel” *Proc of 36th Conf. on Info. Sciences and Systems*;, pp. 859-862, Princeton Univ., Mar 20-22, 2002.

[60] Sumei Sun Wiemer, I. Ho, C.K. Tjhung, T.T., "Training sequence assisted channel estimation for MIMO OFDM", *Wireless Communications and Networking*, 2003. WCNC 2003, IEEE Vol. 1, pp. 38- 43, March 2003

[61] Changho Suh, Chan-Soo Hwang, Hokyu Choi, "Preamble design for channel estimation in MIMO-OFDM systems", GLOBECOM 2003 - *IEEE Global Telecommunications Conference*, vol. 22, no. 1, Dec 2003 pp. 317-321

[62] Y. (Geoffrey) Li, ``Simplified channel estimation for OFDM systems with multiple transmit antennas," *IEEE Transactions on Wireless Communications*, vol. 1, pp. 67-75, January 2002.

[63] Hassibi, B.; Hochwald, B.M.; "Hassibi "How much training is needed in multiple-antenna wireless links?" *Information Theory, IEEE Transactions* on Volume 49, Issue 4, Page(s):951 – 963, April 2003

[64] Xiaoli Ma; Liuqing Yang; Giannakis, G.B.; "Optimal training for MIMO frequency-selective fading channels", *Wireless Communications, IEEE Transactions* on Volume 4, Issue 2, pp. 453 – 466, March 2005

[65] X. Deng and A.M. Haimovich, "On Pilot Symbol Aided Channel Estimation for Time Varying Rayleigh Fading Channels," in Proc. *the 38th Annual Conference on Information Sciences and Systems (CISS'04),* Princeton, New Jersey, Mar. 2004.

[66] A.petropulu, R.Zhang and R. Lin "Blind OFDM channel estimation through simple linear precoding" *IEEE Transaction on Wireless Communication*, Vol.3, No.2, pp.647-655, Mar 2004

[67] Jinho Choi "Equalization and semi-blind channel estimation for space-time block coded signals over a frequency-selective fading channel", *Signal Processing, IEEE Transactions on*, Volume 52, Issue 3,Page(s):774 - 785, March 2004

[68] C. Budianu and L. Tong "Channel Estimation for Space-Time Orthogonal Block Codes," in *IEEE Transactions on Signal Processing,* vol. 50 , no. 10 , pp. 2515 - 2528, Oct. 2002.

[69] Xiaohong Meng and J.K. Tugnait, ``MIMO channel estimation using superimposed training," in *Proc. 2004 IEEE International Conf. on Communications*, Paris, France, June 20-24, 2004.

[70] M. Loncar, R. Müller, J. Wehinger, C. Mecklenbräuker, T. Abe, ”Iterative Channel Estimation and Data Detection in Frequency Selective Fading MIMO Channels “,in *European Transactions on Telecommunications* (ETT), 09-2004, pp. 459-470

[71] A. Grant, “Joint decoding and channel estimation for space-time codes,”. *IEEE Vehicular Technology Conference*, pp. 416-420, 2000.

[72] X. Deng, A.M. Haimovich, and J. Garcia-Frias, “Decision Directed Iterative Channel Estimation for MIMO Systems” in Proc. *IEEE International Conference on Commun. 2003* ICC’03, vol:4, pp. 2326-2329.

[73] Zhu Haidong, Farhang-Boroujeny B, Schlegel C. “Pilot embedding for joint channel estimation and data detection in MIMO communication systems.” *IEEE Communications Letters*, 7(1): 30-32, 2003

[74] C. Fragouli, N. Al-Dhahir, and W. Turin “Training-Based Channel Estimation for Multiple Antenna Broadband Transmissions” , *IEEE Transactions on Wireless Communications*, March 2003

[75] H. Bölcskei, R. W. Heath Jr., and A. J. Paulraj, “Blind channel identification and equalization in OFDM-based multi-antenna systems,” *IEEE Trans. Signal Process.*, vol. 50, no. 1, pp. 96–109, Jan. 2002.

[76] Z. Liu, G. B. Giannakis, S. Barbarossa, and A. Scaglione, “Transmit-antennae space-time block coding for generalized OFDM in the presence of unknown multipath,” *IEEE J. Select. Areas Commun.*, vol. 19, no. 7, pp. 1352–1364, Jul. 2001.

[77] S. Zhou, B. Muquet, and G. B. Giannakis, “Subspace-based (semi-) blind channel estimation for block precoded space-time OFDM,” *IEEE Trans. Signal Process.*, vol. 50, pp. 1215–1228, May 2002.

[78] S. Ohno and G. B. Giannakis, “Optimal training and redundant precoding for block transmissions with application to wireless OFDM,” *IEEE Trans. Commun.*, vol. 50, no. 12, pp. 2113–2123, Dec. 2002.

[79] IEEE 802.11a Standard, ISO/IEC 8802-11:1999/Amd 1:2000(E).

[80] B. M. Popovi´c, “Synthesis of power efficient multitone signals with flat amplitude spectrum,” *IEEE Trans. on Commun.,* vol.39, pp.1031-1033, 1991

[81] Y. (G.) Li, N. Seshadri, and S. Ariyavisitakul, "Channel estimation for OFDM systems with transmitter diversity in mobile wireless channels," *IEEE J. Select. Areas Commun.*, vol. 17, pp. 461–471, Mar. 1999.

[82] P. Spasojevic and C.Georghiades, "Complementary sequence for ISI channel estimation," *IEEE Trans. Inform. Theory*, vol. 47, pp. 1145-1152, Mar. 2001.

[83] G. Caire and U. Mitra, "Training sequence design for adaptive equalization of multi-user systems," in *Proc. 32nd Asilomar Conf.*, vol. 2, 1998, pp. 1479-1483.

[84] W. H.Mow, *Sequence Design for Spread Spectrum* Hong Kong: The Chinese University Press: Chinese Univ. Hong Kong, 1995.

[85] D. C. Chu, "Polyphase codes with good correlation properties," *IEEE Trans. Inform. Theory*, vol. IT–18, pp. 531-532, July 1972.

[86] W. H. Mow, "A new unified construction of perfect root-of-unity sequences," in *Proc. Spread-Spectrum Techniques and Applications* , vol. 3, 1996, pp. 955-959.

[87] S. N. Crozier, D. D.Falconer, and S. A.Mahmoud, "Least sum of squared errors (LSSE) channel estimation," in *Proc. IEEE*, pt. F, vol. 138, no. 4, Aug. 1902, pp. 371-378.

[88] C. Fragouli, N. Al-Dhahir, and W. Turin, "Finite-alphabet constant- amplitude training sequence for multiple-antenna broadband transmissions," in *Proc. IEEE Int. Conf. Commun.*, vol. 1, New York,NY, Apr. 28–May 1 2002, pp. 6–10.

[89] C. Fragouli, N. Al-Dhahir, "Reduced-complexity training schemes for multiple-antenna broadband transmissions," in *Proc. Wireless Communications Networking Conf.*, vol. 1, Mar. 17–21, 2002, pp. 78–83.

[90] I. Barhumi, G. Leus, and M. Moonen, "Optimal training sequences for channel estimation in MIMO OFDM systems in mobile wireless channels," in *Proc. Int. Zurich Seminar on Access, Transmission, Networking of Broadband Communications*. Zurich, Switzerland, Feb. 19–21, 2002, pp. 44-1–44-6.

[91] T.-L. Tung, K. Yao, and R. E. Hudson, "Channel estimation and adaptive power allocation for performance and capacity improvement of multiple-antenna OFDM systems," in *Proc. 3rd IEEE Workshop Signal Processing Advances in Wireless Communications*, Taoyuan, Taiwan, R.O.C., Mar. 20–23, 2001, pp. 82–85.

[92] Z. Wang, X. Ma, and G. B. Giannakis, "Optimality of single-carrier zero-padded block transmissions," in *Proc. Wireless Communications Networking Conf.*, vol. 2, Orlando, FL, Mar. 17–21, 2002, pp. 660–664.

[93] B. Muquet, M. de Courville, and P. Duhamel, "Subspace-based blind and semi-blind channel estimation for OFDM systems," *IEEE Trans. Signal Processing*, vol. 50, pp. 1699-1712, July 2002.

[94] E. Moulines, P. Duhamel, J.-F. Cardoso, and S. Mayrague, "Subspace methods for the blind identification of multichannel FIR filters," *IEEE Trans. Signal Processing*, vol. 43, pp. 516–525, Feb. 1995.

[95] A. L. Swindlehurst and G. Leus, "Blind and semi-blind equalization for generalized space-time block codes," *IEEE Trans. Signal Processing*, vol. 50, pp. 2489–2498, Oct. 2002.

[96] V. Buchoux, O. Cappe, E. Moulines, and A. Gorokhov, "On the performance of semi-blind subspace-based channel estimation," *IEEE Trans.Signal Processing*, vol. 48, pp. 1750–1759, June 2000.

[97] Li Q, Georghiades N, Wang X. An iterative receiver for turbo-coded pilot-assisted modulation in fading channels. *IEEE Communications Letters* 2001; 5(4):145–147.

[98] Valenti M, Woerner B. Iterative channel estimation and decoding of pilot symbol assisted turbo codes over flat-fading channels. *IEEE Journal on Selected Areas in Communications* 2001; 19(9):1697–1705.

[99] B. Farhang-Boroujeny, "Pilot-based channel identification: A proposal for semi-blind identification of communication channels," *Electronic Letter.* vol. 31, pp. 1044–1046, June 1995.

[100] F. Chin, B. Farhang-Boroujeny, and C. K. Ho, "Added pilot semi-blind channel estimation scheme for OFDM in fading channels," in *Proc. IEEE Globecom'01*, vol. 5, San Antonio, TX, Nov. 25–29, 2001, pp. 3075–3079.

[101] Speth, M., Fechtel, S. A., Fock, G., Meyr, H.: *"*Optimum Receiver Design for Wireless Broad-Band Systems Using OFDM - Part I" *, IEEE Transactions on Communications, Vol. 47, No. 11, November 1999*

[102] S. A. Fechtel, “OFDM carrier and sampling frequency synchronization and its performance on stationary and mobile channels,” *IEEE Trans. Consum. Electron.*, vol. 46, pp. 438–441, Aug. 2000.

[103] K. K. Dong, H. D. Sang, C. B. Hong, C. J. Hyung, and K. B. Ki, “A new joint algorithm of symbol timing recovery and sampling clock adjustment for OFDM systems,” *IEEE Trans. Consum. Electron*., vol. 44, pp. 1142–1149, Aug. 1998.

[104] M. J. F.-G. Garcia and J. M. Paez-Borrallo, “Tracking of time misalignments for OFDM systems in multipath fading channels,” *IEEE Trans. Consum.* Electron., vol. 48, pp. 982–989, Nov. 2002.

[105] M. Luise and R. Reggiannini, “Carrier frequency recovery in all-digital modems for burst-mode transmissions,” *IEEE Trans. Commun.*, vol. 43, pp. 1169–1178, Feb.-Apr. 1995.

[106] K. Shi, E. Serpedin et P. Ciblat, “Decision-Directed Fine synchronization in OFDM systems”, *IEEE Transactions on Communications*, vol. 53, no. 3, pp. 408-412, Mars 2005.

[107] Oberli, C.; Daneshrad, B., “Maximum likelihood tracking algorithms for MIMO-OFDM” *Communications, 2004 IEEE International Conference on*, Volume 4, 20-24, Page(s):2468 - 2472 June 2004

[108] G. Santella, “Frequency and symbol synchronization system of OFDM signals: Architecture and simulation results”, *IEEE Trans. on Vehicular Technology,* vol. 49, no. 1, pp. 254-275, Jan 2000.

[109] H. Liu and U. Tureli, “ A high efficiency carrier estimator for OFDM communications”, *IEEE Communication Letters*, vol. 2, no. 4, pp. 104-106, April 1998.

[110] J.-J. van de Beek, M. Sandell, and P. O. Börjesson, “ML estimation of time and frequency offset in OFDM systems,” *IEEE Trans. Signal Process*., vol. 45, pp. 1800–1805, Jul. 1997.

[111] T.C.W. Schenk and A. van Zelst, "Frequency Synchronization for MIMO OFDM Wireless LAN Systems", Proc. *IEEE Vehicular Technology Conference* Fall 2003 (VTC Fall 2003), Orlando (FL), 6-9 October 2003, paper 05D-03.

[112] R.L. Frank and S.A. Zadoff, “Phase Shift Pulse Codes With Good Periodic Correlation Properties”, *IRE Trans. on Information Theory*, vol. IT-8, pp. 381-382. 1962.

[113] D. C. Rife and R. R. Boorstyn, "Single-Tone Parameter Estimation From Discrete-Time Observations", *IEEE Trans. Inform. Thoery*, vol. IT-20, pp. 591-598, Sept. 1974.

[114] X. Ma, M.-K. Oh, G. B. Giannakis, and D.-J. Park, "Hopping Pilots for Estimation of Frequency-Offsets and Multi-Antenna Channels in MIMO-OFDM," *IEEE Transactions on Communications*, vol. 53, no. 1, pp. 162-172, January 2005.

[115] S. Barbarossa, M. Pompili, and G. B. Giannakis, "Channel-Independent Synchronization of Orthogonal Frequency Division Multiple Access Systems," *IEEE Journal on Selected Areas in Communications*, vol. 20, no. 2, pp. 474-486, February 2002.

[116] H. Bölcskei, "Blind estimation of symbol timing and carrier frequency offset in wireless OFDM systems," *IEEE Trans. Commun*., vol. 49, pp.988–999, Jun. 2001.

[117] Y. Yao, and G. B. Giannakis, "Blind Carrier-Frequency Offset Estimation of SISO, MIMO, and Multi-User OFDM,"' *IEEE Transactions on Communications*, vol. 53, no. 1, pp. 173-183, January 2005.

[118] Y. Sun, Z. Xiong and X. Wang, "EM-based iterative receiver design with carrier frequency offset estimation for OFDM-coded MIMO systems," *IEEE Trans. Comm*., to appear in Apr. 2005

[119] B. Lu, X. D. Wang, and Y. (Geoffrey) Li, ``Iterative receivers for space-time block coded OFDM systems in dispersive fading channels," *IEEE Transactions on Wireless Communications*, vol. 1, pp. 213-225, April 2002.

[120] Henk Wymeersch, Frederik Simoens and Marc Moeneclaey "Code-aided joint channel estimation and frame synchronization for MIMO systems", *Workshop on Signal Processing Advances in Wireless Communications (SPAWC'04), Lisbon, Portugal, July 2004.*

[121] Tim C.W. Schenk, Maurice M. de Laat, Peter F.M. Smulders and Erik R. Fledderus, "Symbol timing for multiple antenna OFDM systems", in: *Vehicular Technology Conference*, 2006. VTC 2006-Spring.

[122] M. Morelli and U. Mengali, "Carrier-frequency estimation for transmissions over selective channels," *IEEE trans. Commun.,* vol. 48, pp.1580-1589, Sept 2000

[123] P. Stoica, O. Besson, "Training sequence design for frequency offset and frequency-selective channel estimation" *IEEE trans. Commun.* Vol 51, pp:1910 – 1917, 2003

[124] J. A. Fessler and A. O. Hero, “Space-alternating generalized expectation- maximization algorithm,” *IEEE Trans. Signal Processing*, vol. 42, pp. 2664–2677, Oct. 1994.

[125] T. Moon, “The expectation-maximization algorithm*”, IEEE Signal Processing Mag*., vol. 13, n°6, pp. 47-60, 1996.

[126] M. Feder and E. Weinstein, “Parameter estimation of superimposed signals using the EM algorithm”*, IEEE Trans. ASSP*, vol. 36, n°4, April 1998.

[127] C. Georghiades and J. C. Han, “Sequence estimation in the presence of random parameters via the EM algorithm”, *IEEE Trans. Commun*., vol. 45, n°3, pp. 300-308, 1997.

[128] A.P. Dempster, N.M. Laird and D.B. Rubin, “Maximum likelihood from incomplete data via the EM algorithm”. *Journal of the Royal Statistical Society*, 39(1):pp. 1-38,1977, Series B.

[129] X. Wautelet, C. Herzet, A. Dejonghe, L. Vandendrope, “MMSE-base and EM Iterative Channel Estimation Methods”, *Symposium IEEE Benelux Chapter on Communications and Vehicular Technology*, 2003

[130] Y. Xie and C.N. Georghiades, "Two EM-type channel estimation algorithms for OFDM with transmitter diversity," *IEEE Trans. Commun*., vol.51, no.1, pp.106–115, Jan. 2003.

[131] Henk Wymeersch, , “Software Radio Algorithms for Coded Transmission”, *Faculty of Engineering, Ghent University, September2005.* Advisors**:** Prof. M. Moeneclaey and Prof. H. Steendam

[132] Henk Wymeersch, Frederik Simoens and Marc Moeneclaey “Code-aided joint channel estimation and frame synchronization for MIMO systems”, *Workshop on Signal Processing Advances in Wireless Communications (SPAWC'04), , July 2004*

[133] J. J. Van de Beek, P.O. Borjesson, M. L. Boucheret, D. Landstrom, J. M. Arenas, O. Odling, M. Wahlqvist and S. K. Wilson, “A time and frequency synchronization scheme for multiuser OFDM,” *IEEE JSAC*, vol. 17, pp. 1900-1914, Nov. 99.

[134] S. Barbarossa, M. Pompili and G. B. Giannakis, “Channel-independent synchronization of orthogonal frequency division multiple access systems,” *IEEE JSAC*, vol. 20, pp. 474-486, Feb. 2002.

[135] Z. Cao, U. Turelli and Y. D. Yao, “Efficient structure-based carrier frequency offset estimation for interleaved ofdma uplink,” *ICC Proc*., May 2003.

[136] M. Morelli, “Timing and frequency synchronization for the uplink of an OFDMA system,” *IEEE Trans. Commun*. vol. 52, pp. 296-306, Feb. 2004.

[137] M. Morelli and U. Mengali, “Carrier-frequency estimation for transmissions over selective channels”, *IEEE Trans. Commun.*, vol. 48, pp. 1580-1589, Sept. 2000

[138] M.O Pun, M. Morelli, C. C. Jay Kuo, “Maximum-Likelihood Synchronization and Channel Estimation for OFDMA Uplink Transmissions”, *IEEE Trans. Commun*., vol. 54, pp. 726-736, Apr. 2006.

[139] I. Ziskind and M. Wax, “Maximum likelihood localization of multiple sources by alternating projection”, *IEEE trans. Acoust. Speech, Signal Process*., vol. 36, n°10, pp. 1553-1560, Oct. 1998.

[140] B. Lu, X. Wang, and K. R. Narayanan, “LDPC-based space-time coded OFDM systems over correlated fading channels,” *IEEE Trans. Commun*., vol. 50, pp. 74-88, Jan. 2002.

[141] T. Roman et al, "Joint time-domain tracking of channel and frequency offsets for MIMO-OFDM systems", *Wireless Personal Communications*, vol31, pp181-200, 2004.

[142] U. Tureli and P. J. Honan, “Modified high-efficiency carrier estimator for OFDM communications with antenna diversity,” in Proc. 35th *Asilomar Conf*. Signals, Syst., Comput., vol. 2, Pacific Grove, CA, Nov. 2001, pp.1470–1474

[143] H. Meyr, M. Oerder and A. Polydoros, “On sampling rate, analog pre.ltering, and suf.cient statistics for digital receivers,” *IEEE Trans. Commun*., vol. 42, pp. 3208-3213, Dec. 1994.

[144] S. Kay, *Fundamentals of Statstical Signal processing: Estimation theory*. Englewood Cliffs, NJ: Prentice-Hall, 1993.

[145] P.Stoica and R. Moses, *Introduction to Spectral analysis*. Upper Saddle River, NJ: Prentice-Hall, 1997.

## ABSTRACT

Theoretical studies of communication links employing multiple transmit and receive antennas, also known as multiple-input multiple-output (MIMO), have shown great potential for providing highly spectrally efficient wireless transmissions. The early investigations focused almost entirely on flat fading channels. To consider frequency selective channel one efficient method in high rate wireless systems is Orthogonal Frequency-Division Multiplexing (OFDM). MIMO-OFDM combines OFDM and MIMO techniques thereby achieving spectral efficiency and increased throughput.

However, because of using Discrete Fourier Transform (DFT), an OFDM system is very sensitive to carrier frequency offset (CFO), which introduces inter-carrier interference. Accurate frequency synchronization is thus essential for reliable reception of the transmitted data. On the other hand, incorrect positioning of the DFT window within an OFDM word reintroduces ISI during data demodulation, causing serious performance degradation. This dissertation deals with MIMO-OFDM synchronization. To this aim, several synchronization algorithms are studied in this manuscript and a new Maximum-Likelihood (ML) joint time-frequency MIMO-OFDM synchronization algorithm together with channel estimation is presented. Moreover, an iterative Expectation-Maximization (EM) time-frequency synchronization algorithm is introduced and at the end the problem of synchronization in MIMO-OFDMA systems is considered.



## KEYWORDS

SISO, MIMO, OFDM, OFDMA, synchronization, symbol timing, frame synchronization, carrier frequency offset, CFO, channel estimation, ML, EM



## RESUME

La technique multi-antenne dans les systèmes de communication numérique (MIMO) sans fil augmente considérablement la capacité du canal de propagation. Les premières études se sont concentrées sur les canaux non-sélectifs en fréquence. Afin de combattre l'effet multi trajet des canaux radio-mobile, la modulation OFDM a été proposée depuis quelques années. La combinaison d'OFDM avec MIMO ouvre la porte vers des communications hauts débits.

Cependant, un système OFDM est très sensible à une erreur de fréquence porteuse qui détruit l'orthogonalité entre les porteuses. Cet effet va dégrader radicalement la performance du système. Les recherches rapportées dans ce mémoire aborde le problème de la synchronisation fréquentielle et temporelle ainsi que l'estimation du canal MIMO des systèmes MIMO-OFDM. Après avoir dressé l'état de l'art de la problématique correspondante, de nouveaux algorithmes basés sur l'algorithme de maximisation de vraisemblance et l'algorithme d'espérance et maximisation (EM), ont été proposés, puis simulés. Les résultats obtenus ont été comparés avec ceux de la littérature en terme de performance et de complexité. A la fin de ce mémoire le problème de la synchronisation dans les systèmes MIMO-OFDMA est étudié et un algorithme original est proposé.



## MOT CLES

SISO, MIMO, OFDM, OFDMA, synchronisation, timing de symbole, synchronisation de trame, CFO, estimation de canal, maximisation de vraisemblance, ML, EM