國 立 交 通 大 學

電子工程學系 電子研究所碩士班

碩 士 論 文

多重輸入輸出正交分頻多工系統

之同步設計研究

Designs of Synchronization Techniques for MIMO

OFDM Systems

研 究 生：劉佳旻

指導教授：陳紹基 博士

中 華 民 國 九 十 四 年 七 月

多重輸入輸出正交分頻多工系統

之同步設計研究

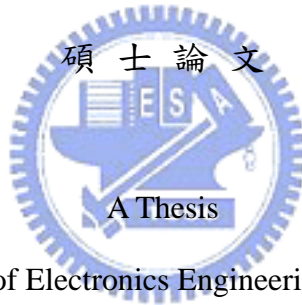# Designs of Synchronization Techniques for MIMO

OFDM Systems

研 究 生：劉佳旻　　　　　　Student：Chia-Min Liu

指導教授：陳紹基 博士　　　　Advisor：Sau-Gee Chen

國 立 交 通 大 學

電子工程學系　電子研究所碩士班

碩 士 論 文

A Thesis

Submitted to Department of Electronics Engineering & Institute of Electronics

College of Electrical Engineering and Computer Science

National Chiao Tung University

in Partial Fulfillment of the Requirements

for the Degree of

Master

in

Electronics Engineering

July 2005

Hsinchu, Taiwan, Republic of China

中華民國九十四年七月

多重輸入輸出正交分頻多工系統
之同步設計研究

學生：劉佳旻　　　　　　指導教授：陳紹基 博士

國立交通大學

電子工程學系　電子研究所碩士班

摘　　要

　　由於能有效處理頻率選擇性衰減的問題，正交分頻多工技術成為無線通訊技術中被廣泛使用的技術。高速資料傳送及高品質的通訊服務已成為無線通訊發展的主要挑戰之一，而多傳輸天線及多接收天線系統設計是有效克服這些問題的方法。然而，正交分頻多工技術最主要的缺點在於對頻率及時間偏移的敏感。在本篇論文裡，我們設計一套全面的同步系統，適用於多天線正交分頻多工的架構。此外，使用近似於 IEEE 802.1n 的設定來進行模擬。這個同步流程的設計包括封包偵測，頻率偏移的同步技術，符元時間偏移估測及時脈同步設計。在封包偵測的部份，我們把自相關運算的輸出值和訊號強度門檻做比較來偵測出封包的開端[44]。在符元時間偏移估測部分，我們先使用兩個接續窗口方法進行粗略的符元時間偏移估測，再用所提出的簡化直接時間維度精確符元時間同步(STDFS)方法去微調得到更精確的符元時間偏移估測。由於 IEEE 802.11n 先導訊號的特色，我們提出平滑方法利用其特色藉由平均自相關運算輸出值去加強頻率偏移的估

測。時脈偏移估測部分，我們藉由只運算最為分離的引導訊號去降低運算複雜度。至於時脈偏移補償部分，經由模擬結果，我們建議在 IEEE 802.11n 系統使用 Lagrange 內插器較為適合。

Designs of Synchronization Techniques
for MIMO OFDM Systems

Student: Chia-Min Liu   Advisor: Dr. Sau-Gee Chen

Department of Electronics Engineering &

Institute of Electronics

National Chiao Tung University

ABSTRACT

Since the ability to deal with the frequency-selective fading channel, orthogonal frequency division multiplexing (OFDM) is a popular method for wireless communication. The development of wireless communication for high date rate and high-quality service is becoming one of the major challenging targets in wireless communication. Multiple-input multiple-output (MIMO) OFDM technique is an efficient solution for these targets. However, a major drawback of OFDM is the high sensitivity to frequency and timing offset. We propose an overall synchronization scheme suitable for multiple-input multiple-output (MIMO) orthogonal frequency division multiplexing (OFDM) systems in this thesis. Moreover, we conduct the simulations in IEEE 802.11n-like setting. The synchronization schemes considered include frame timing detection, carrier frequency offset synchronization, symbol timing estimation and sampling clock offset synchronization. In the frame detection, we detect the beginning of a frame by comparing the auto-correlation outputs to the signal

power threshold [44]. In the symbol timing synchronization, we perform the coarse symbol timing estimation first with the double-sliding-window method [45], then adjust the symbol timing by the proposed simplified direct time-domain fine symbol timing synchronization (STDFS) method to get a more accurate symbol timing estimation than the current techniques. Owing to the features of the IEEE 802.11n preambles, we propose a smoothed method which utilizes the features to enhance the carrier frequency offset estimation by averaging the available auto-correlation outputs. In the sampling clock offset estimation, we reduce the computation complexity by only computing the most apart pilots. As for the sampling clock compensation, we find that Lagrange interpolator is suitable for IEEE 802.11n systems from simulation results.

誌　謝

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Due to the ability to deal with frequency selective fading and inter symbol interference, orthogonal frequency division multiplexing (OFDM) [1] is a popular method for wireless communication. OFDM may be combined with MIMO to increase the diversity gain and enhance the system capacity on frequency selective channels, resulting in MIMO-OFDM system designs [2]. In the proposals for IEEE Std. 802.11n [3][4], MIMO-OFDM is the major solution for high data rate and high quality service. However, the bit error rate performance of OFDM system is sensitive to timing and frequency error between transmitters and receivers. Precise synchronization is necessary in order to achieve the potential performance.

In this thesis, we investigate and design overall synchronization procedures for

MIMO OFDM systems, including the frame detection, the symbol timing synchronization, the carrier frequency offset synchronization, the sampling clock offset synchronization, and simulate the overall synchronization schemes by the Matlab programs in IEEE 802.11n-like setting.

1.1 Overview of IEEE 802.11n System

In response to growing market demand for higher-performance wireless local area networks (WLANs), the Institute of Electrical and Electronics Engineers - Standards Association (IEEE-SA) approved the creation of the IEEE 802.11 Task Group N (802.11 TGn) during the second half of 2003. The scope of TGn's objective is to define modifications to the Physical Layer and Medium Access Control Layer (PHY/MAC) that deliver a minimum of 100 megabit-per-second (Mbps) throughput.

WWiSE [3] and TgnSync [4] proposals are two major proposals for TGn Standards Association. They mainly follow IEEE802.11a [5][6] in their proposals and both claim backward compatibility with existing IEEE WLAN legacy solutions (802.11a/b/g). They both take advantage of multiple-input and multiple-output (MIMO) which is able to deliver 100 megabit-per-second (Mbps) throughput.

1.2 Motivation

Similar to single-input single-output (SISO) OFDM system, MIMO-OFDM system requires synchronization in both the time and frequency. Timing offset error

refs to the incorrect timing of OFDM symbols introducing inter-symbol–interference (ISI). Carrier frequency offset error is the misalignment of subcarrier frequency due to the fluctuations in receiver RF oscillators or channel's Doppler frequency. The carrier frequency offset will destroy the subcarrier orthogonal characteristic and introduce inter-carrier-interference (ICI). Sampling frequency offset is due to the misalignment of sampling clocks between the transmitter and the receiver. In digital communication systems, the receiver's clock has to be synchronized with the transmitter's. However, when the transmit frame becomes very long or the number of subcarriers in the OFDM system is very large, the clock frequency offset mismatch has to be taken into account.

For the reasons mentioned above, we propose an overall synchronization scheme including timing synchronization, carrier frequency synchronization, and sampling clock synchronization for MIMO-OFDM system using training symbols (data-aided) in this thesis.

1.3   Organization of This Thesis

The thesis is organized as follows. In chapter 2, we describe the MIMO-OFDM system model, explain the MIMO-OFDM concept and discuss the synchronization issues and the effects of synchronization errors. In chapter 3, we introduce the background and the physical layer concept of IEEE 802.11n. The proposed synchronization scheme is described in chapter 4. The synchronization scheme is simulated by the Matlab program and the results are shown in chapter 5. Finally, chapter 6 gives the conclusions and future work.

Chapter 2

Fundamentals of MIMO-OFDM Systems

2.1 MIMO-OFDM Basics [7]

The basic principle of OFDM is to divide the high-rate data stream into many low rate streams that each is transmitted simultaneously over its own subcarrier orthogonal to all the others. Due to narrowband property, they experience mostly flat fading, which makes channel equalization very simple. In order to eliminate intersymbol interference (ISI) and intercarrier interference (ICI) as much as possible, it is a good idea to add a trailing portion of each symbol to the head of itself, which is called cyclic prefix extension as shown in Figure 2.1. The guard interval is chosen larger than the expected

delay spread, such that multi-path components from one symbol will not interfere with its succeeding symbol. After a signal passes through the time-dispersive channel, orthogonality of its subcarrier components can be maintained by the introduction of cyclic prefix.
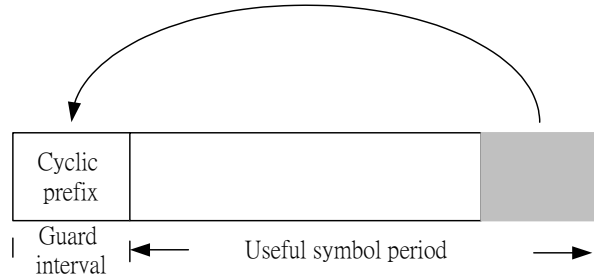
Figure 2.1 Guard interval and cyclic prefix structure of an OFDM symbol.

For the reasons mentioned above, OFDM is a leading modulation technique for wireless communications. Combining it with MIMO transmission systems increases the achievable throughput over wireless media significantly.

2.1.1 Baseband MIMO-OFDM Model

A MIMO OFDM system with N_T transmitter and N_R receiver antennas is considered as an $N_T \times N_R$ MIMO-OFDM system. Figure 2.2 shows the baseband structure of an $N_T \times N_R$ MIMO-OFDM system. The OFDM symbol transmitted by the pth transmit antenna is given by

$$s^p(n) = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} S^p(k) e^{j\frac{2\pi nk}{N}}, \quad 0 \le n \le N \tag{2.1}$$

where $S^p(k)$ denotes the transmitted data from the pth transmitter antenna, $1 \le p \le N_T$,

and N denotes the number of subcarriers. The received signal on the qth receiver antenna is

$$r^q(n) = \sum_{p=1}^{N_T} h^{p,q}(n) * s^p(n) + v^q(n) \qquad (2.2)$$

where $h^{p,q}(n)$ denotes the multi-path Rayleigh fading channel between the pth transmitter and the qth receiver antenna, and $v^q(n)$ denotes the complex additive white Gaussian noise with variance N_0 at the qth receiver antenna. The received signal is demodulated with FFT as

$$\begin{aligned} R^q(k) &= \sum_{n=0}^{N-1} r^q(n) e^{-j\frac{2\pi nk}{N}} \\ &= \sum_{p=1}^{N_T} H^{p,q}(k) S^p(k) + V^q(k), 0 \le k \le N \end{aligned} \qquad (2.3)$$
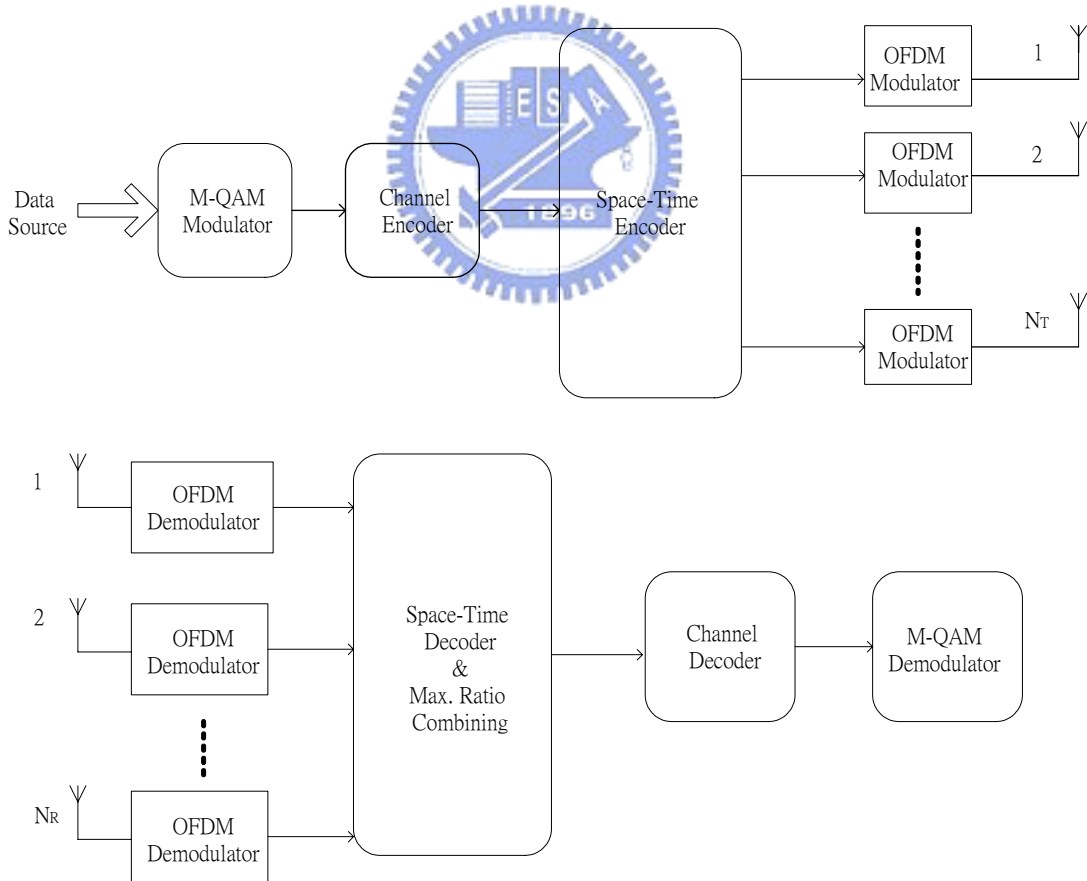
Figure 2.2 Block diagram of an $N_T \times N_R$ MIMO-OFDM system.

2.1.2 Channel Estimation of MIMO OFDM Systems [38]

Transmitter diversity is an effective technique for combating fading in mobile wireless communications, especially when receiver diversity is expensive or impractical. For OFDM systems with transmitter diversity using space-time coding, two or more different signals are transmitted from different antennas simultaneously. The received signal is the superposition of these signals, usually with equal power.

The transmitter antennas simultaneously transmit OFDM signals $S_p(k)$. Hence, discrete Fourier transform (DFT) of the received signal at each receiver antenna is the superposition of distorted transmitted signals, which can be expressed as

$$R(k) = \sum_{p=1}^{N_T} H_p(k) S_p(k) + V(k) \tag{2.4}$$

The frequency response at the kth tone corresponding to the pth transmitter antenna can be expressed as

$$H_p(k) = \sum_{l=0}^{K_0-1} h_p(l) W_N^{kl}, \qquad W_N = \exp(-j2\pi / N) \tag{2.5}$$

where K_o is the number of taps. Note that K_0 depends on the delay profiles and dispersion of the wireless channels.

According to [38], we can use the known long training sequences to find the temporal estimation of the channel parameters by minimizing the following MSE cost function.

$$C\left(\left\{\hat{h}_p(l); p = 1, 2, ..., N_T\right\}\right)$$
$$= \sum_{k=0}^{N-1} \left| R(k) - \sum_{p=1}^{N_T} \sum_{l=0}^{K_0-1} \hat{h}_p(l) W_K^{kl} T_p(k) \right|^2 \tag{2.6}$$

Further details can be found in [38, 39]. We will use the channel parameter estimation approaches to decode the space-time block coded data.

2.1.3 Space-Time Block Coding [40, 41]

Alamouti proposed a code scheme with full diversity that is originally designed for systems with two transmitter antennas and flat-fading channel [40]. Two consecutive symbols are processed in the transmitter in such a way that signals on the antennas are orthogonal. Therefore, they can be combined easily in the receiver. The scheme is further extended to four antennas [41]. Alamouti's scheme is discussed as follows.

Table 2.1 The symbol placement of Alamouti's scheme.

	Antenna 0	Antenna 1
Time t	S_0	S_1
Time t+T	$-S_1^*$	S_0^*

According to signal format listed in Table 2.1, the channel is assumed flat and constant across two consecutive symbols. That is,

$$h_0(t) = h_0(t+T) = h_0 = \alpha_0 e^{j\theta_0}$$
$$h_1(t) = h_1(t+T) = h_1 = \alpha_1 e^{j\theta_1}$$

(2.7)

where T is symbol duration. The received signal can be expressed as

$$r_0 = r(t) = h_0 S_o + h_1 S_1 + v_0$$
$$r_1 = r(t+T) = -h_0 S_1^* + h_1 S_0^* + v_1$$

(2.8)

8

where r_0 and r_1 are received signals at time t and $t+T$, respectively, while v_0 and v_1 are complex random variables representing receiver noise and interference, respectively. The combiner builds the following two combined signals.

$$\hat{S}_0 = h_0^* r_0 + h_1 r_1^*$$
$$\hat{S}_1 = h_1^* r_0 - h_0 r_1^*$$

(2.9)

Substituting (2.7) and (2.8) into (2.9), we get

$$\hat{S}_0 = (\alpha_0^2 + \alpha_1^2)S_0 + h_0^* v_0 + h_1 v_1^*$$
$$\hat{S}_1 = (\alpha_0^2 + \alpha_1^2)S_1 - h_0 v_1^* + h_1^* v_0$$

(2.10)

Unfortunately, the flat fading channel is usually hard to achieve. OFDM systems separate the total frequency-selective fading channel into lots of sub-channels, which are often flat fading. Therefore, OFDM systems can achieve the condition. Thus, for each sub-channels, the channel condition matches the constraint placed by Alamouti.

In both WWiSE and TGnSync IEEE 802.11n proposals, the number of data streams is supported up to four. In order to achieve the optional 4×3 MIMO OFDM configuration, we extend the scheme to four antennas proposed in [41].

When the number of data streams is three and four, we use the following space-time block codes, H_3 and H_4, to encode the data. The decoders for H_3 and H_4 are derived in the Appendix of [41].

$$H_3 = \begin{pmatrix} S_1 & S_2 & \dfrac{S_3}{\sqrt{2}} \\ -S_2^{\;*} & S_1^{\;*} & \dfrac{S_3}{\sqrt{2}} \\ \dfrac{S_3^{\;*}}{\sqrt{2}} & \dfrac{S_3^{\;*}}{\sqrt{2}} & \dfrac{\left(-S_1 - S_1^{\;*} + S_2 - S_2^{\;*}\right)}{2} \\ \dfrac{S_3^{\;*}}{\sqrt{2}} & -\dfrac{S_3^{\;*}}{\sqrt{2}} & \dfrac{\left(S_2 + S_2^{\;*} + S_1 - S_1^{\;*}\right)}{2} \end{pmatrix} \qquad (2.11)$$

$$H_4 = \begin{pmatrix} S_1 & S_2 & \dfrac{S_3}{\sqrt{2}} & \dfrac{S_3}{\sqrt{2}} \\ -S_2^{\;*} & S_1^{\;*} & \dfrac{S_3}{\sqrt{2}} & -\dfrac{S_3}{\sqrt{2}} \\ \dfrac{S_3^{\;*}}{\sqrt{2}} & \dfrac{S_3^{\;*}}{\sqrt{2}} & \dfrac{\left(-S_1 - S_1^{\;*} + S_2 - S_2^{\;*}\right)}{2} & \dfrac{\left(-S_2 - S_2^{\;*} + S_1 - S_1^{\;*}\right)}{2} \\ \dfrac{S_3^{\;*}}{\sqrt{2}} & -\dfrac{S_3^{\;*}}{\sqrt{2}} & \dfrac{\left(S_2 + S_2^{\;*} + S_1 - S_1^{\;*}\right)}{2} & -\dfrac{\left(S_1 + S_1^{\;*} + S_2 - S_2^{\;*}\right)}{2} \end{pmatrix} (2.12)$$

2.2 Synchronization Issues of MIMO OFDM

OFDM is an efficient method of fast data transmission over frequency selective fading channels. Due to the division of the data stream into many sub-streams transmitted at a lower data rate on different sub-carriers and the application of a guard period preceding the data pulse, ISI (inter-symbol interference) is almost avoided. In most of the OFDM systems, the guard period is filled with the samples from the end of the symbol making the requirements for timing recovery easier.

Although OFDM is a spectrally efficient method of digital modulation, it has some serious drawbacks. The major one is that it is highly sensitive to synchronization errors. In MIMO-OFDM systems, the effect of synchronization errors is also a serious problem. Therefore, MIMO-OFDM systems also require synchronization in both the

time and frequency domains.

Synchronization of an OFDM signal requires detecting the beginning of a frame, finding the symbol boundary, the carrier frequency offset, and the sampling clock offset. Here, we will discuss the effects of the carrier frequency offset, the symbol timing offset, and the sampling clock offset to an OFDM based system.

2.2.1 Effect of the Carrier Frequency Offset

Frequency offset is caused by inaccuracies and thermal changes of the oscillators used in the transmitter and receiver as well as by the terminal mobility (Doppler spread). In the IEEE 802.11n the frequency offset Δf in comparison to the subcarrier spacing of the OFDM signal is small.

Before discussing the effect of carrier frequency offset, we define the normalized frequency offset η as

$$\eta = \frac{\Delta f}{f_{spacing}} \tag{2.13}$$

$$f_{spacing} = \frac{1}{NT_s}$$

where $f_{spacing}$ is the subcarrier spacing of the OFDM system.

Generally, the fractional part of η causes attenuation and introduces ICI (Inter-Carrier-Interference), and the integral part of η introduces the effect that the received symbols would appear at incorrect output bins of the FFT demodulator. To prevent such cases, the frequency offset estimation and correction algorithm must be established.

The sampled signal with the sampling period T_s is [8]

$$r(n) = e^{j2\pi n \Delta f T_s} \frac{1}{N} \sum_{k=0}^{N-1} S(k) e^{j\frac{2\pi nk}{N}}$$

$$= \frac{1}{N} \sum_{k=0}^{N-1} S(k) e^{j\frac{2\pi n(k+\eta)}{N}}; \qquad n = 0,1,....,N-1 \qquad (2.14)$$

Thus, the received signal sample on the k-th subchannel is

$$\hat{S}(k) = \sum_{n=0}^{N-1} r(n) e^{-j\frac{2\pi nk'}{N}}$$

$$= \sum_{n=0}^{N-1} \frac{1}{N} \sum_{k=0}^{N-1} S(k) e^{j\frac{2\pi n(k+\eta)}{N}} e^{-j\frac{2\pi nk'}{N}} \qquad (2.15)$$

$$= S(k') \frac{\sin(\pi\eta)}{N \sin(\pi\eta/N)} e^{j\pi\eta(\frac{N-1}{N})} + I(k')$$

The first term shows the received $\hat{S}(k)$ amplitude degradation and phase shift due to the frequency offset. The second term $I(k')$ is the ICI caused by the frequency offset.

If degradation D in SNR caused by a frequency offset that is small relative to the subcarrier spacing, then it can be approximated as [9]

$$D \cong \frac{10}{3\ln 10} (\pi\Delta f T_u)^2 \frac{E_s}{N_0} \qquad (2.16)$$

where E_s is the averaged received energy, $N_0/2$ is the power spectral density of the AWGN, and T_u is the useful symbol time.

It can be clearly seen that the effects introduced by frequency offset Δf are twofold. First, the frequency offset causes attenuation and phase rotation of the desired data signal $S(k')$. The second form of the influence of the frequency offset is the intercarrier interference expressed by the second term. It cannot be easily compensated by the set of the complex equalizer. These two effects vindicate that the compensation of the frequency offset is an important task.

2.2.2 Effect of the Symbol Timing Offset

In symbol timing synchronization, inter channel interference (ICI) and inter symbol interference (ISI) occur when the FFT window wrongly covers an area that partially contains the symbol and cyclic prefix of the following symbol, as shown by case 3 in Figure 2.3.

Case 1 in Figure 2.3 shows the perfect FFT window position in AWGN channels [7]. In this case, the orthogonality of the sub-carriers is maintained, and there is no phase rotation in the frequency domain after the FFT block. However, it is not the case for a transmission in multi-path fading channels [10][11]. The symbol timing should be estimated in between the last delay path and the end of guard interval to avoid the ISI. Thus, the ISI is perfectly removed only when the first and last delay paths are accurately estimated.

When the cyclic prefix is longer than the channel delay spread, there is a certain range in the prefix itself that is not affected by the previous symbol. As long as the FFT window starts within this range (Case 2 in Figure 2.3), the symbol timing error just results in a circular shift in the time domain and therefore in a phase rotation in the frequency domain after the FFT block. In this case, the orthogonality of the sub-carriers is maintained.

In Case 3, the FFT will cover partially not only the target samples belonging to symbol n, but also part of the cyclic prefix of the next symbol $n+1$. Thus part of the information is lost and the prefix of symbol n+1 will cause irrecoverable ISI. The effect of this unwanted ISI is that it will destroy the orthogonality between the subcarriers of current symbol. As a result, one can see spreading effects of the transmitted constellation points which can be modeled as additional noise.
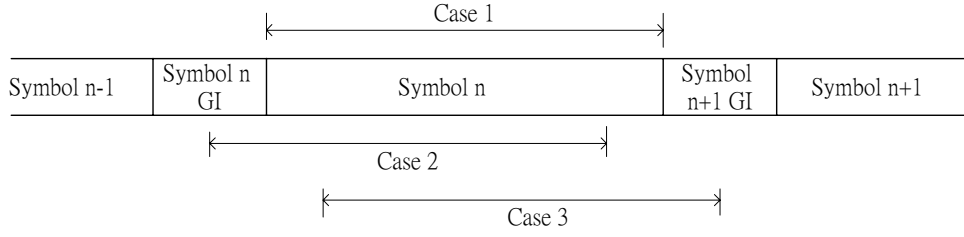
Figure 2.3 Scenario of boundary misplacement of a received OFDM symbol.

For the reasons mentioned above, the FFT window must be selected within the appropriate range. Thus, an MIMO-OFDM system needs to do timing synchronization and avoids the effect of timing errors.

2.2.3 Effect of Sampling Clock Offset

Sampling clock errors introduce the clock phase error and the clock frequency error. Clock phase error effects are similar to symbol timing errors and can be treated in the same way. The constant fractional error can be compensated by a simple rotation or interpolation techniques. However, the time varying offset results in phase changes and ICI, because the sub-carriers are not orthogonal any more.

If we define the sampling clock frequency offset as

$$\beta = \frac{T_s' - T_s}{T_s} \tag{2.17}$$

where T_s and $T_s^{'}$ are the transmitter sampling period and the receiver sampling period respectively. The received signal can be expressed as [12,13]

$$R_{n,k} = e^{j2\pi kn\beta \frac{N+N_g}{N}} H_{n,k} S_{n,k} \operatorname{sinc}(\pi k\beta) + V_{n,k} + N_\beta(n,k) \tag{2.18}$$

where $\operatorname{sinc}(x) = \sin(\pi x)/\pi x$, n is the sample index, k is the subcarrier index, and $N_\beta(n,k)$

14

is the noise caused by the clock frequency offset with variance [13]

$$\mathrm{Var}\left[N_\beta(n,k)\right] \approx \frac{\pi^2}{3}(k\beta)^2 \qquad\qquad (2.19)$$

The degradation grows at a rate proportional to the square of the product of the clock offset β and the subcarrier index *k*. The phase rotation increases with symbol index *n*. The clock offset β is small in most practical case, sinc($\pi\beta k$)≈1. Thus, the effect of sampling clock offset can be approximated as an additional phase rotation. Under wireless communication conditions $N_\beta(n,k)$ can be neglected (*βk << 1*), but the phase rotation increases with symbol index and cannot be neglected.

Table 2.2 lists the parameters of several wireless communication standards about the effect of sampling clock offset. In DAB system, the number of symbols in a frame is fixed as 76 and the modulation scheme is DQPSK. Therefore, the sampling clock offset is not a serious problem in this system. However, when the frame size is not fixed or the number of subcarriers is large, we suggest that the sampling clock synchronization is needed, especially for 8K-mode DVB-T.
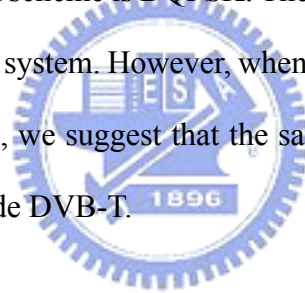
Table 2.2 Parameter comparison among various OFDM standards.

	F_s	N subcarrier	Modulation
802.11n	20MHz	64 128	BQPSK QPSK 16QAM 64QAM
DAB	2.048MHz	2048 512 256 1024	DQPSK
DVB	9.14MHz 8MHz 6.8MHz	2048 8192	QPSK 16QAM 64QAM
802.16a	22.857MHz	2048	QPSK 16QAM 64QAM

F_s : the sampling frequency. $N_{subcarrier}$: the number of subcarriers.

Chapter 3

Introduction to IEEE 802.11n

3.1 Overview of IEEE 802.11 Standard

The original version of the standard IEEE 802.11 [5] released in 1997 specifies two raw data rates of 1 and 2 megabits per second (Mbps) to be transmitted via infrared (IR) signals or in the Industrial Scientific Medical frequency band at 2.4 GHz. IR remains a part of the standard but has no actual implementations.

In the IEEE 802.11 family, the most popular techniques are those defined by the a, b, and g amendments to the original standard; security was originally included, and was later enhanced via the 802.11i amendment. Other standards in the family (c–f, h–j, n)

are service enhancement and extensions, or corrections to previous specifications. 802.11b [15] was the first widely accepted wireless networking standard, followed by 802.11a [6] and 802.11g [16].

802.11a standard uses the 5 GHz band, while 802.11b and 802.11g standards use the unlicensed 2.4 GHz band. Operating in an unregulated frequency band, 802.11b and 802.11g equipments will result in interference from microwave ovens, cordless phones, and other appliances using the same 2.4 GHz band.

The IEEE 802.11 standard was adopted in 1997. Since then, several extensions to the standard have been developed, and more are emerging. The complete family of the current and emerging 802.11 standards are listed in Table 3.1.

3.2 History and Current Status of IEEE 802.11n
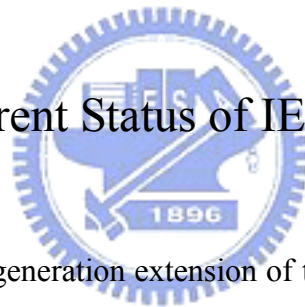
The 802.11n is the next generation extension of the physical layer. It is expected that 802.11n will support throughput rates (useful data rates) of over 100 Mbps. The standard is still in the earlier development phase. Among the proposed approaches to provide such high data rates are smart antenna technology, enhanced modulation, and increased channel bandwidth (using both 2.4 and 5GHz bands).

IEEE 802.11n standard process has three stages: the stage 1 is the preparation stage from Jan. 2002 to Sep. 2002; the stage 2 is the IEEE 802.11 High Throughput Study Group (HTSG) from Sep. 2002 to Sep. 2003; the stage 3 is the IEEE 802.11n Task Group from Sep. 2003 to Sep. 2005 (expected).

Table 3.1 Summary of IEEE 802.11 standards.

Standard	Description	Status
802.11	The original 1 Mbps and 2 Mbps 2.4 GHz RF and IR standard	Released in 1997
802.11a	54Mbps,5GHz	Ratified in1999, shipping products in 2001
802.11b	Enhancements to 802.11 to support 5.5 and 11 Mbps, 2.4GHz	Ratified in1999
802.11c	Access point bridging	Completed
802.11d	Regulatory extensions	Completed
802.11e	Quality of Service	Completed
802.11f	Inter Access Point roaming	Completed
802.11g	54Mbps,2.4GHz (backwards compatible with b)	Completed in 2003
802.11h	Transmit power control, Dynamic frequency selection	Completed
802.11i	Enhanced security	Completed
802.11j	Japanese regulatory extensions	Completed
802.11k	Radio resource measurement	Ongoing
802.11m	Maintenance	Ongoing
802.11n	Higher throughput improvements (100+ Mbps)	Expected completion in 2006-2007

In January 2004, IEEE announced that it had formed a new 802.11 Task Group (TGn) to develop a new amendment to the 802.11 standard for local-area wireless networks. The real data throughput rate will be at least 100 Mbps (which may require an even higher raw data rate at the physical layer), and should be up to 4–5 times faster than 802.11a or 802.11g, and perhaps 20 times faster than 802.11b. It is projected that 802.11n will also offer a better operating distance than current networks. There are two competing variants of the 802.11n standard: WWiSE (backed by companies including Broadcom) and TGn Sync (backed by Intel, Philips and others). The standardization process is expected to be completed by the end of 2006.

802.11n builds upon previous 802.11 standards by adding MIMO (multiple-input multiple-output). The additional transmitter and receiver antennas allow for increased data throughput through spatial multiplexing and increased range by exploiting the spatial diversity, perhaps through coding schemes like Alamouti coding.

TGnSync [4] and WWiSE [3] are the two major proposals for IEEE 802.11n Task Group. Both proposals emphasize compatibility with existing installed base, building on experience with interoperability in 802.11a/g and previous 802.11 amendments. Therefore, in the next section, we review the physical layer of wireless LAN 802.11a systems based on OFDM technology.

3.3 Physical Layer of IEEE 802.11a

The 802.11a standard, introduced at the same time as 802.11b, is intended for the 5 GHz license-free UNII band and provides data rates up to 54 Mbps. The 5 GHz band has an advantage of large bandwidth allocated for the unlicensed operations. There are

455 MHz available (5.15 – 5.35 MHz and 5.470 – 5.725 MHz) for use by WLAN systems in Europe. This allows 19 non-overlapping channels in the 5 GHz band versus 3 non-overlapping channels in the 2.4 GHz band.

802.11a is based on Orthogonal Frequency Division Multiplexing (OFDM) modulation, and allows to achieve higher data rates within about the same channel bandwidth as 802.11b.

The IEEE 802.11a operates at a sampling rate of 20M Hz and uses 64-point FFT. The OFDM frame duration is 80 samples where 64 is for data while 16 is cyclic prefix. Since the symbol rate on each subcarrier is slower than the original data rate, the OFDM technique is particularly efficient in time dispersive environments. The 802.11a OFDM signal consists of 52 carriers. Data are sent on 48 carriers simultaneously, with 4 carriers used as pilots to aid in channel estimation at the receiver. The main system parameters of IEEE 802.11a Wireless LAN standard are listed in Table 3.2.

Using different modulation QAM combined with punctures of convolutional encoder, variable data rate can be achieved with a minimum 6Mbps and maximum 54 Mbps. Table 3.3 shows supported data rates. Various data rates are provided by changing the redundancy in the error correction coding and by changing modulation scheme.

Adopted in 2003, the 802.11g extension enables 54 Mbps data rates, the same data rate as provided by the 802.11a standard, but now in the 2.4 GHz band. This is achieved by using the same data rates and modulation formats as used in the 802.11a standard. Additionally, the 802.11g standard is backward compatible with the 802.11b standard, i.e. the 802.11b modulation formats and data rates are supported.

Table 3.2 The parameters of IEEE 802.11a system.

Sampling rate	20MHz
Number of FFT points	64
Number of data subcarriers	48
Number of pilot subcarriers	4
Subcarrier spacing	0.3125 MHz (=20MHz/64)
OFDM symbol period	4 μ s (80 samples)
Cyclic prefix period	0.8 μ s (16 samples)
FFT symbol period	6.2 μ s (64 samples)
Modulation scheme	BPSK,QPSK,16QAM,64QAM
Coding	1/2 convolutional, constraint length 7, optional puncturing
Data rate	6, 9, 12, 18, 24, 36, 48, 54 Mbps
Short training sequence duration	8 μ s
Long training sequence duration	8 μ s
Long training symbol GI duration	1.6 μ s

Table 3.3 Data rates and rate-dependent parameters of 802.11a standard.

Data rate, Mbps	Modulation	Coding rate	Coded bits per subcarrier	Coded bits per OFDM symbol	Data bits per OFDM symbol
6	BPSK	1/2	1	48	24
9	BPSK	3/4	1	48	36
12	QPSK	1/2	2	96	48
18	QPSK	3/4	2	96	72
24	16-QAM	1/2	4	192	96
36	16-QAM	3/4	4	192	144
48	64-QAM	2/3	6	288	192
54	64-QAM	3/4	6	288	216

In IEEE 802.11a standard, a frame is composed of three fields. Figure 3.1 shows the frame format of IEEE 802.11a. The preamble field is modulated with QPSK, and no interleaving and scrambling. Figure 3.2 shows the preamble format and the possible arrangement of the synchronization and channel estimation for the receiver. In the preamble field, the preambles are composed of ten repeated short symbols and two repeated long symbols. Both the total durations of short training symbols and long training symbols are 8 μs. The SIGNAL field is modulated with BPSK, interleaving, but no scrambling. Because the SIGNAL contains the most important information of the packet, every synchronization and channel estimation must be done before decoding of the SIGNAL. In the DATA field, modulation and coding rate depend on the information carried by SIGNAL field, interleaving and scrambling are executed.
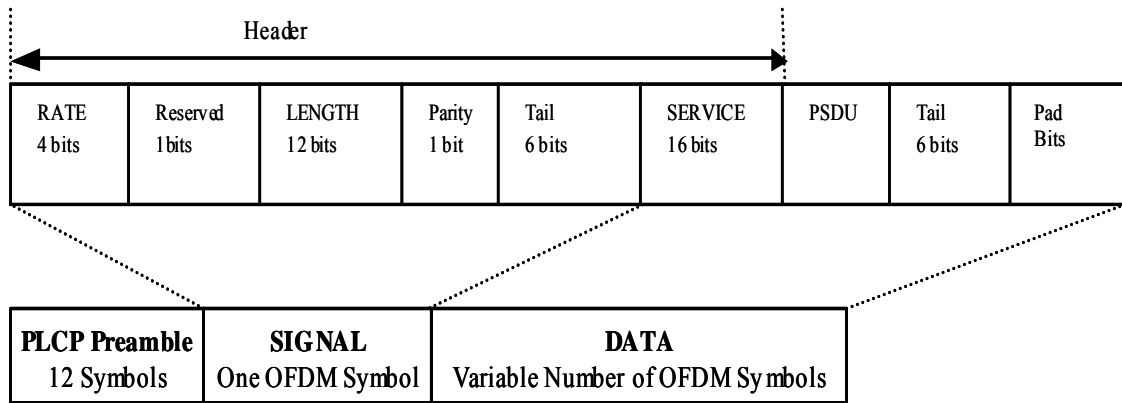
Figure 3.1 The frame format of IEEE 802.11a standard.
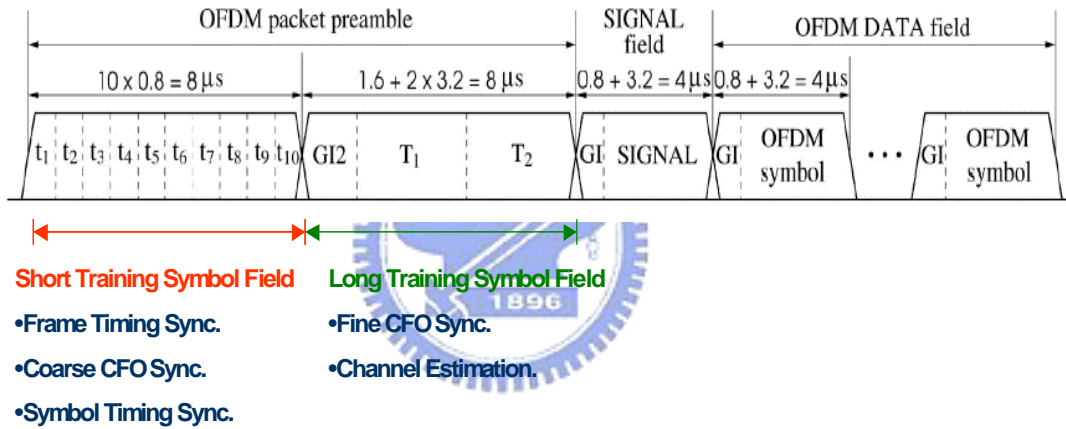
Figure 3.2 The training symbol structure of 802.11a.

3.4 Preamble Format of IEEE 802.11n

For the purpose of compatibility with the 802.11 legacy devices, the legacy part of preamble format in WWiSE and TGnSync proposals is the same as in 802.11a. If the legacy preambles are transmitted from multiple antennas, the mapping of this single spatial stream to multiple antennas has to be done such that beamforming in far-field is mitigated. One method for achieving this is to use a cyclical delay diversity (CDD) mapping. The cyclical delay is used by both WWiSE and TGnSync proposals. In WWiSE proposal, the CDD format of each spatial data stream is disclosed clearly in the documents. For the reason given above, we will use WWiSE cyclical delay format for simulation.

Green-field preambles and mixed-mode preambles are the two preamble types in WWiSE proposal. Green-field preambles operate with only 11n devices, but mixed-mode preambles are capable of operation in presence of legacy 11a/g device. Green-field preambles have greater efficiency than mixed-mode preambles. Figure 3.3 shows the two preamble types of WWiSE proposal. Since the synchronization procedures have to be operated before the SIGNAL field, we must use the preambles in front of the SIGNAL field to correct the timing and frequency offsets.

STRN	GI2	LTRN	SIGNAL-N	GI2	LTRN
STRN 400 ns cs	GI2₁	LTRN 1600 ns cs	SIGNAL-N 1600 ns cs	GI2₁	LTRN 1600 ns cs
STRN 200 ns cs	GI2₂	LTRN 100 ns cs	SIGNAL-N 100 ns cs	–GI2₂	–LTRN 100 ns cs
STRN 600 ns cs	GI2₃	LTRN 1700 ns cs	SIGNAL-N 1700 ns cs	–GI2₃	–LTRN 1700 ns cs
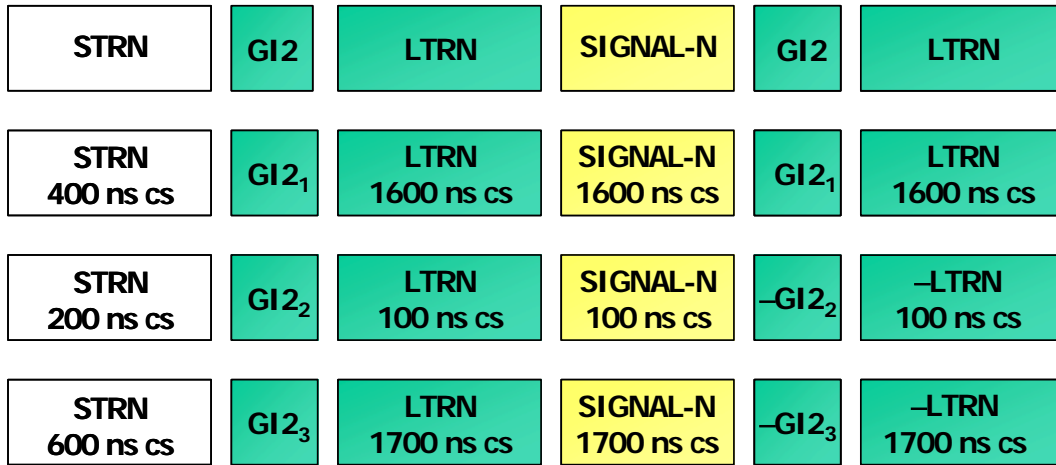
(a) Green-field preamble format.

STRN	GI2	LTRN	SIGNAL	
STRN 400 ns cs	GI2₄	LTRN – 100 ns cs	SIGNAL – 100 ns cs	2, 3 or 4 transmitter green-field long training and SIGNAL-N
STRN 200 ns cs	GI2₅	LTRN 100 ns cs	SIGNAL 100 ns cs	
STRN 600 ns cs	GI2₆	LTRN 200 ns cs	SIGNAL 200 ns cs	
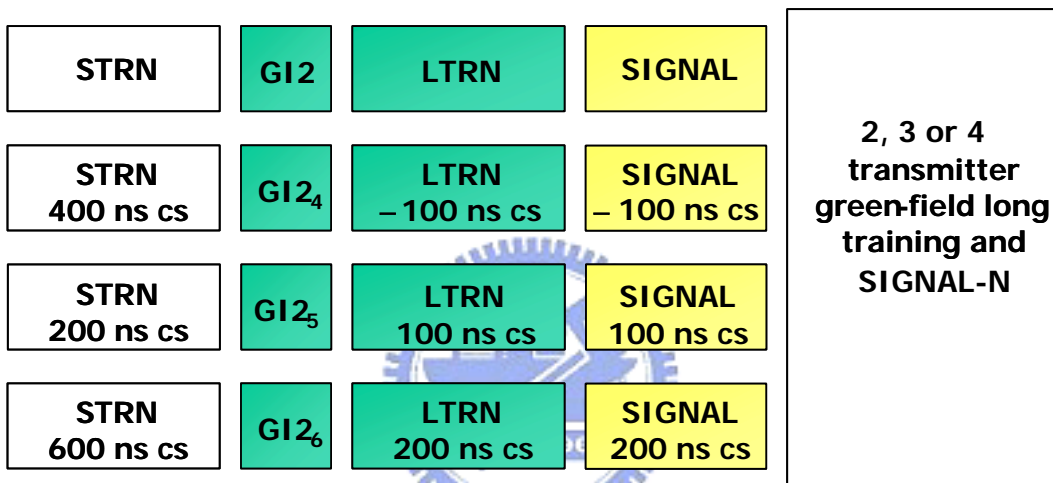
(b) Mixed-mode preamble format.

Figure 3.3 The preamble formats in WWiSE 802.11n proposal.

Note: STRN: the short training sequence. LTRN: the long training symbol. GI2: the

guard interval of the successive long training symbol. SIGNAL: the signal field

capable of 11a. SIGNAL-N: the new signal field in 11n.

Chapter 4

Synchronization Techniques for MIMO OFDM Systems

Synchronization plays an important role in the receiver design of MIMO-OFDM systems. The tasks of synchronization include frame detection, symbol timing synchronization, carrier frequency offset synchronization, and sampling clock offset synchronization. The methods for synchronizations in MIMO-OFDM systems are similar to SISO-OFDM systems. However, we can combine more estimated parameter samples from multiple receiver antennas than SISO cases and obtain more accurate synchronization.

First, we have to detect the frame start, then estimate the coarse symbol timing, and the coarse frequency offset. Finally, we estimate the fine frequency offset, the fine

symbol timing, and the sampling clock offset. Figure 4.1 shows the flow of synchronization scheme.

Step 1: Detect the beginning of frame.

Step 2: Use the short training sequence to estimate the coarse CFO, average the coarse CFOs estimated by multiple receiver antennas to obtain the more accurate estimation, and compensate the coarse CFO.

Step 3: Use the short training sequence to estimate the coarse symbol timing.

Step 4: Use the long training sequence to estimate the fine CFO, average the fine CFOs estimated by multiple receiver antennas to obtain the more accurate estimation, and compensate the fine CFO.

Step 5: Use the pilots in data symbol to estimate the clock offset, average the clock offsets estimated by multiple receiver antennas to obtain more accurate estimations, and compensate the clock offsets.

We will discuss those schemes further in the following sections.

```
                    ┌─────────────────────────┐
                    │ Frame Timing Estimation │
                    └─────────────────────────┘
                ┌──────────────────┴──────────────────┐
                ▼                                      ▼
    ┌──────────────────────┐            ┌──────────────────────────┐
    │ Coarse CFO Estimation│            │  Coarse Symbol Timing    │
    │                      │            │      Estimation          │
    └──────────────────────┘            └──────────────────────────┘
                │                                      │
                ▼                                      │
    ┌──────────────────────────┐                      │
    │ MIMO Combing of Coarse CFO│                     │
    └──────────────────────────┘                      │
                │                                      │
                ▼                                      │
    ┌──────────────────────────┐                      │
    │  Compensate the Coarse CFO│                     │
    └──────────────────────────┘                      │
                └──────────────────┬──────────────────┘
                                   ▼
                    ┌─────────────────────────┐
                    │   Fine CFO Estimation   │
                    └─────────────────────────┘
                                   ▼
                    ┌─────────────────────────┐
                    │ MIMO Combing of Fine CFO│
                    └─────────────────────────┘
                                   ▼
                    ┌─────────────────────────┐
                    │ Compensate the Fine CFO │
                    └─────────────────────────┘
                                   ▼
                    ┌─────────────────────────┐
                    │Fine Symbol Timing Estimation│
                    └─────────────────────────┘
                                   ▼
                    ┌─────────────────────────┐
                    │ Clock Offset Estimation │
                    └─────────────────────────┘
                                   ▼
                    ┌─────────────────────────┐
                    │   MIMO Combing of COE   │
                    └─────────────────────────┘
                                   ▼
                    ┌─────────────────────────┐
                    │ Compensate the Clock Offset│
                    └─────────────────────────┘
                                   ▼
                    ┌─────────────────────────┐
                    │  Channel Estimation & S-T│
                    │        decoding         │
                    └─────────────────────────┘
```
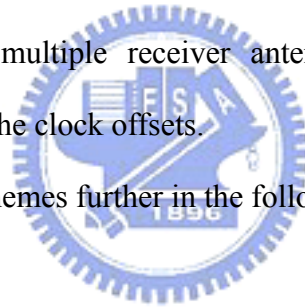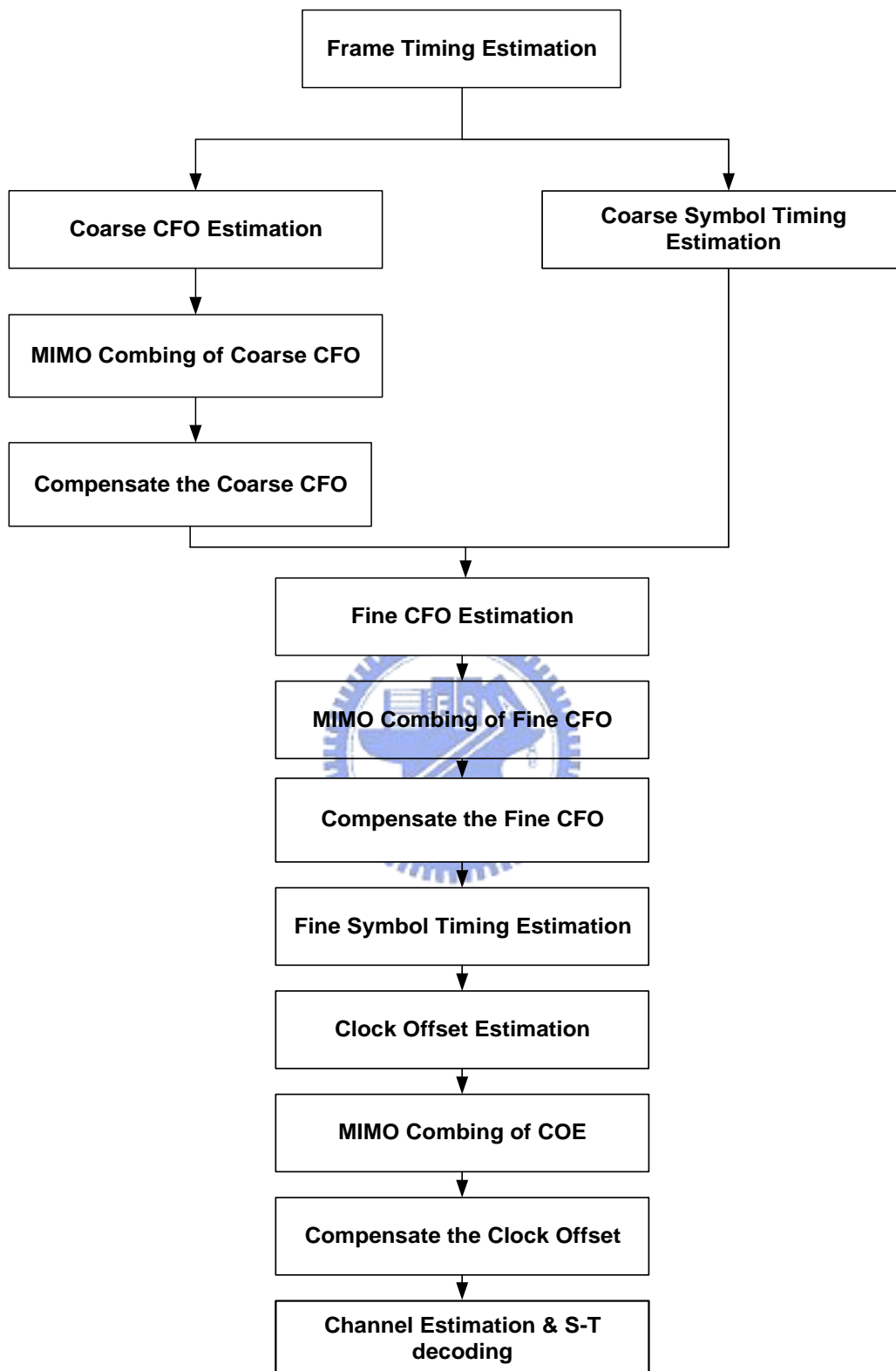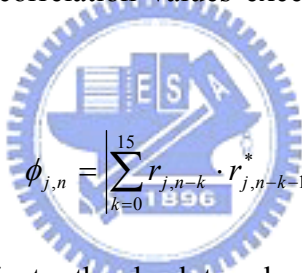
Figure 4.1 The flow chart of the synchronization scheme.

4.1 Frame synchronization [44]

The short training sequence is used to detect the frame start by defining a signal power threshold. The power threshold is obtained by the moving-average of signal power of the following equation:

$$THR_{j,n} = c \times \sum_{k=0}^{N_w} r_{j,n-k} \cdot r^*_{j,n-k} \tag{4.1}$$

where j is the receiver antenna index, n is the sample index, N_w is the moving average window length, and c is the selected power level. Then the moving average of auto-correlation output defined in (4.2) for STRN is compared with the calculated power threshold. If the auto–correlation values exceed the threshold consecutively, frame detection is declared.

$$\phi_{j,n} = \left| \sum_{k=0}^{15} r_{j,n-k} \cdot r^*_{j,n-k-16} \right|^2 \tag{4.2}$$

In Figure 4.2, the solid line indicates the absolute values of the auto–correlation outputs (4.2), the doted line indicates the threshold (4.1). Figure 4.3 shows the block diagram of frame detection.
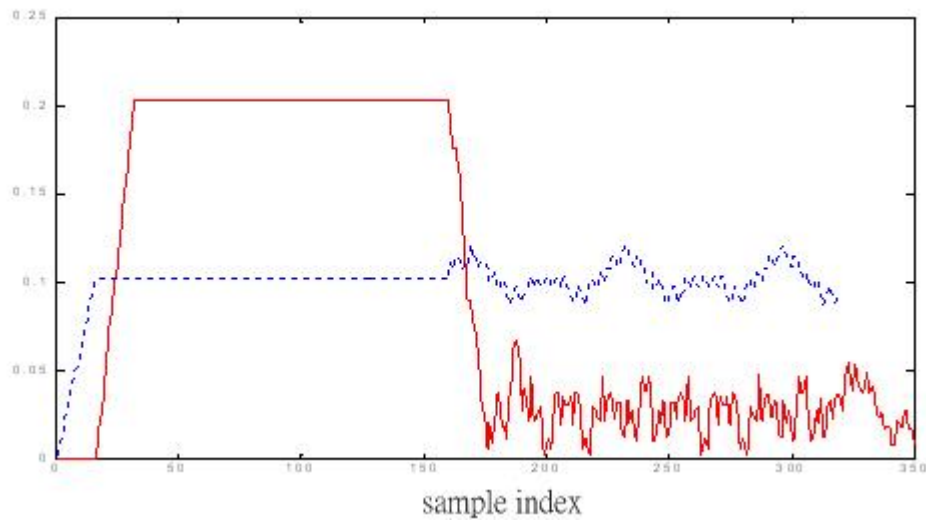
Figure 4.2 The power threshold values and the auto-correlation outputs vs. sample
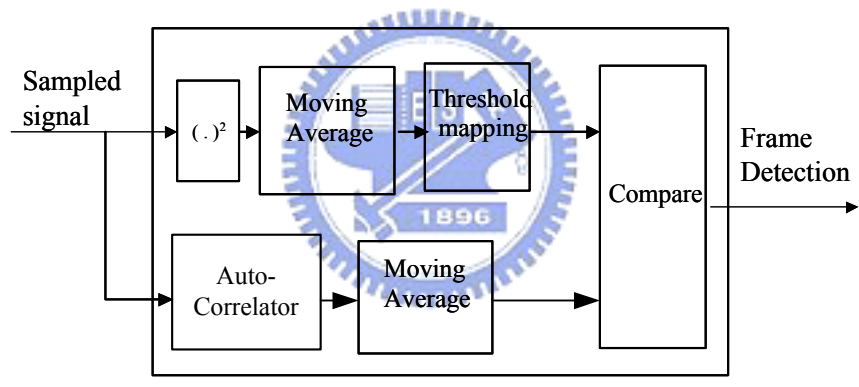
index for STRN in the frame detection [44].

Figure 4.3 The functional block diagram of the frame detection [44].

4.2 Carrier Frequency Synchronization

4.2.1 Conventional Methods for Carrier Frequency Synchronization [17,18,19,20,21]

The short training sequence is used to estimate the coarse carrier frequency offset. Since the format of short preamble, $r_{j,n-k}$ and $r_{j,n-k-16}$ are repeated data which have the same phase; the phase of $\phi_{j,n}$ is affected by the carrier frequency offset so that the coarse frequency can be estimated. Furthermore, the frequency offset estimation of up to +/- 2 subcarrier spacings can be obtained base on the phase of the auto-correlation function in Equation (4.3).

The angle can be detected in three different ways, the conventional method is to detect the peak location or a fixed location of the auto-correlation outputs [17,18,19,20,21]. Equations (4.5) and (4.6) indicate the two methods respectively.

$$\phi_{j,n} = \sum_{k=0}^{15} r_{j,n-k} \cdot r_{j,n-k-16}^{*} \tag{4.3}$$

$$M_j = \arg\max_{n} \{\phi_{j,n}\} \tag{4.4}$$

$$\Delta f_{coarse} = \frac{1}{2\pi * 16 * T_s} \angle \left\{ (1/N_{RX}) \sum_{j=1}^{N_{RX}} \phi_{M_j} \right\} \tag{4.5}$$

$$\Delta f_{coarse} = \frac{1}{2\pi * 16 * T_s} \angle \left\{ (1/N_{RX}) \sum_{j=1}^{N_{RX}} \phi_{N_j} \right\} \tag{4.6}$$

where N_{RX} is the number of the receiver antennas, j is the antenna index, and N_j is a proper fixed index.

After compensating the coarse frequency offset, one can use the long training sequence, which is formed by two repeated 64-point long training symbols and a 32-point guard interval, for fine carrier frequency offset.

$$\psi_{j,n} = \sum_{k=0}^{63} r_{j,n-k} \cdot r_{j,n-k-64}^{*} \qquad (4.7)$$
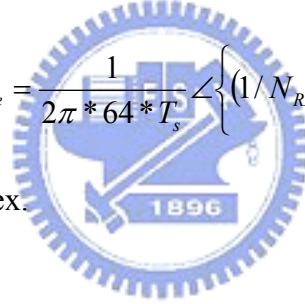
Like the method of the coarse frequency offset estimation, there are two conventional ways to detect the angles of the auto-correlation outputs. We describe the two ways in Equations (4.9) and (4.10) respectively.

$$P_j = \arg\max_{n} \{\psi_{j,n}\} \qquad (4.8)$$

$$\Delta f_{fine} = \frac{1}{2\pi * 64 * T_s} \angle \left\{ (1/N_{RX}) \sum_{j=1}^{N_{RX}} \psi_{P_j} \right\} \qquad (4.9)$$

$$\Delta f_{fine} = \frac{1}{2\pi * 64 * T_s} \angle \left\{ (1/N_{RX}) \sum_{j=1}^{N_{RX}} \psi_{Q_j} \right\} \qquad (4.10)$$

where Q_j is a proper fixed index.

4.2.2 The Proposed Smoothed Method for Carrier Frequency Synchronization

Since the short training sequence is composed of ten repeated short training symbols and the long training sequence is formed by two repeated 64-point long training symbols and a 32-point guard interval, we can average some selected points of the auto-correlation output to decrease the noise effect.

$$\Delta f_{coarse} = \frac{1}{2\pi * 16 * T_S} \angle \{ (1/N_{RX}) \sum_{j=1}^{N_{RX}} avg(\sum_{n \in win1} \phi_{n,j}) \} \qquad (4.11)$$

$$\Delta f_{fine} = \frac{1}{2\pi * 64 * T_S} \angle\{(1/N_{RX})\sum_{j=1}^{N_{RX}} avg(\sum_{n \in win2} \psi_{n,j})\} \qquad (4.12)$$

where win_1 and win_2 are the selected ranges for the auto-correlation outputs. We will compare the performance of the three methods in the next chapter.

Figure 4.4 shows the ranges we selected for the averaging window. The solid line indicates the absolute value of the auto-correlation outputs for the short training sequence (4.3), and the doted line indicates the absolute value of the auto-correlation outputs for the long training sequence (4.7).
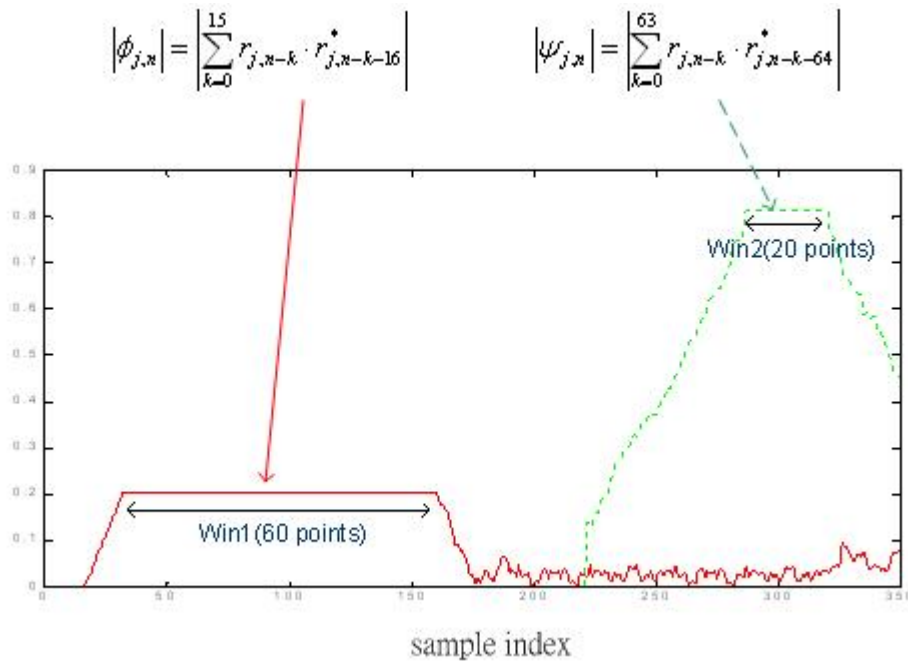

Figure 4.4 The selected ranges of the coarse (win_1) and fine (win_2) frequency

estimation.

4.3 Symbol Timing Synchronization

4.3.1 Coarse Symbol Timing Synchronization

As the number of spatial data stream increases, the matched filtering method [22,23,24] used in 802.11a fails. Figure 2.4 shows the phenomenon clearly. The dotted line indicates the absolute value of the auto-correlation outputs (4.13), and the solid line indicates the absolute value of the matched-filter outputs (4.14). We assume that there is only one receiver antenna in Figure 4.5.

$$\left|\phi_{j,n}\right| = \left|\sum_{k=0}^{15} r_{j,n-k} \cdot r_{j,n-k-16}^{*}\right| \tag{4.13}$$

$$\left|T(n)\right| = \left|\sum_{m=0}^{15} r_{j,n-m} \cdot ref_{15-m}\right| \tag{4.14}$$

As the number of the spatial data stream increases, the matched-filter outputs will decrease accordingly. For the reason given above, one can't use the matched filter output values to estimate symbol timing reliably. The double-sliding-window method [45] can be used to overcome this problem. These two consecutive sliding windows accumulate the absolute values of the short training sequence auto-correlation outputs defined in Equation (4.13).

$$A_{j,n} = \sum_{k=0}^{15} \left|\phi_{j,n-k-16}\right| \tag{4.15}$$

$$B_{j,n} = \sum_{k=0}^{15} \left|\phi_{j,n-k}\right| \tag{4.16}$$

$$ratio_{j,n} = \frac{A_{j,n}}{B_{j,n}} \tag{4.17}$$

where A and B are the two sliding window values. In the sliding process, B window

will slide in the long training field, while A window still in the short training field; the $ratio_{j,n}$ reaches its maximal value. One can use the location of peak value to indicate the coarse symbol timing.
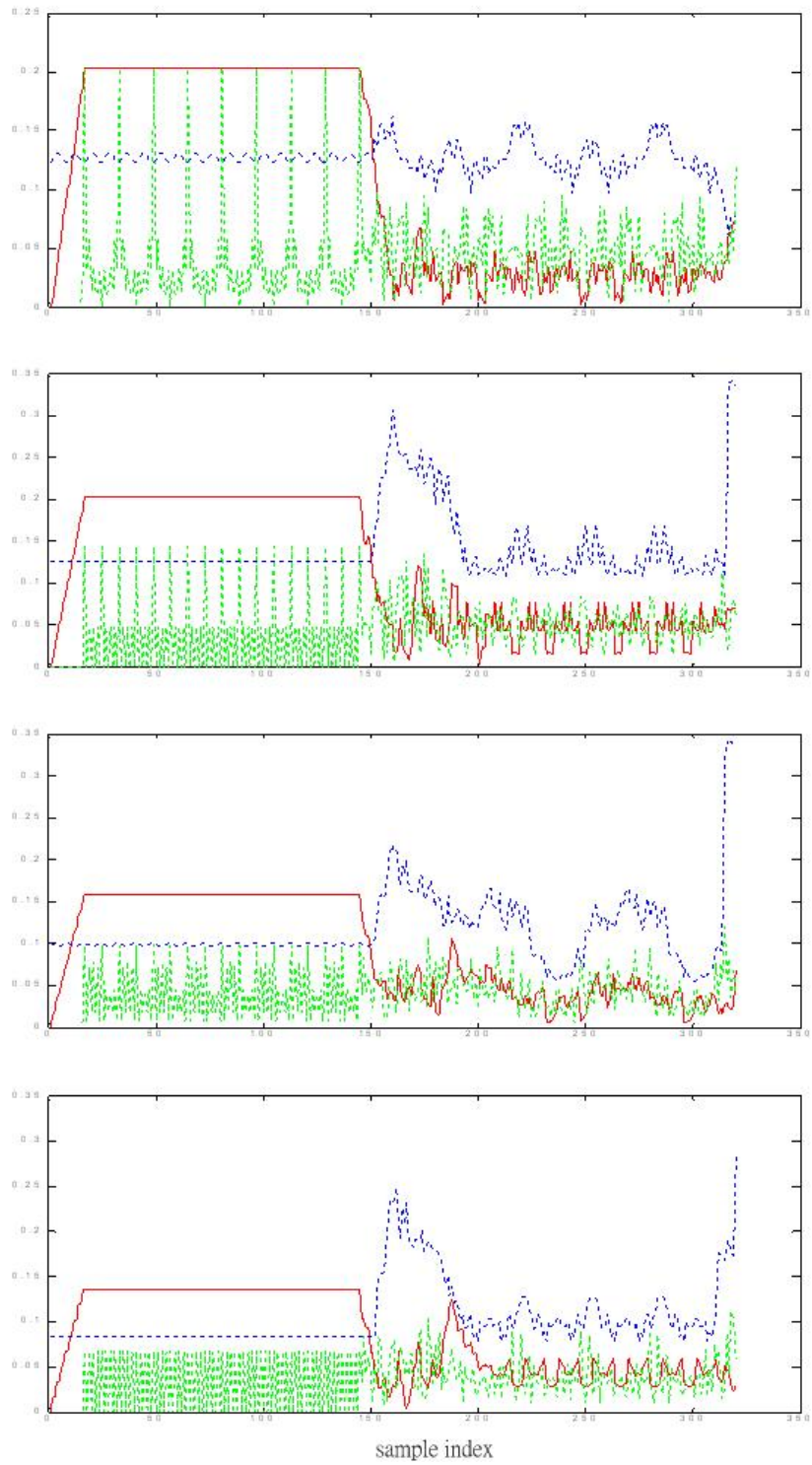
Figure 4.5 Auto-correlation outputs and matched filter outputs, using the

conventional symbol synchronization technique. (a) N_T=1. (b) N_T=2. (c) N_T=3. (d)

N_T=4.

37

4.3.2 Fine Symbol Timing Synchronization

4.3.2.1 Conventional Method for Fine Symbol Timing Synchronization [25, 26]

After the stage of coarse symbol timing estimation, it is assumed that the estimation error is within +/- 5 samples. As the number of transmitter antennas is more than three, the accuracy is not enough for the equalizer. For the reason mentioned above, one has to further increase symbol timing accuracy.

In the literature [25,26], detecting the first path of the channel impulse response is the major method to estimate the fine symbol timing. The conventional method [25, 26] uses IFFT operation to transfer the estimated channel frequency response to time domain, then detect the first path whose magnitude is larger than a proper threshold as follows.

$$R_{n,k} = FFT(r_{m,k}) = X_{n,k}H_{n,k} + W_{n,k} \tag{4.18}$$

$$\hat{H}_{n,k} = \frac{R_{n,k}X_{n,k}^*}{|X_{n,k}|^2} = R_{n,k}X_{n,k}^* \tag{4.19}$$

$$\hat{h}_n(m) = IFFT(\hat{H}_{n,k}), \qquad m = 1,2,....,N \tag{4.20}$$

$$\delta = \left\{ m \left\| \hat{h}_n(m) \right\| > \eta, m = 1,2,....,N \right\} \tag{4.21}$$
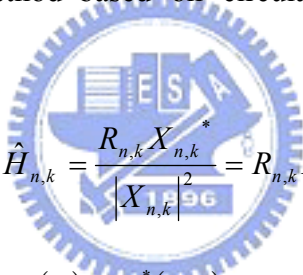
$$\eta = \alpha \times \max\left(\left\|\hat{h}_n(m)\right\|\right) \tag{4.22}$$

where n is the frame index, k is the subcarrier index, $R_{n,k}$ is the received data after FFT, m is the sample time index, $X_{n,k}$ is the all-pilot preamble in the long training field, $|X_{n,k}|^2 = 1$ $k=1,2,...,N$ in 802.11n, η is the threshold, δ is the estimated fine symbol

timing, α is 0.5 in our simulation. We define this conventional method as conventional frequency-domain fine symbol timing synchronization (CFDFS) method in the following.

4.3.2.2 The Proposed Method for Fine Symbol Timing Synchronization

The CFDFS method with IFFT should introduce serious round-off errors, this phenomenon can be analyzed further in the future. For the reason, without resorting to frequency-domain operations we propose the direct time-domain fine symbol timing synchronization (DTDFS) method based on circular convolution to estimate the channel impulse response.

$$\hat{H}_{n,k} = \frac{R_{n,k}X_{n,k}^{*}}{|X_{n,k}|^{2}} = R_{n,k}X_{n,k}^{*}$$

$$\hat{h}_{n}(m) = r_{n}(m) \otimes x_{n}^{*}(-m) \qquad m = 1,2,....,N \qquad (4.23)$$

Since the estimation error is less than +/- 5 samples after the coarse symbol timing estimation, we can reduce the computation complexity by directly calculating time-domain circular convolution outputs in the range of the +/- 5 points as shown by Equation (4.24) and adjust the fine symbol timing. This simplified method is defined as simplified direct time-domain fine symbol timing synchronization (STDFS) method.

$$\hat{h}_{n}(m) = r_{n}(m) \otimes x_{n}^{*}(-m) \qquad m = 1,2,..,5 \ \& \ N-4,N-3,..,N \qquad (4.24)$$

Table 4.1 shows the computation complexity of the three methods, we can find that the computation complexity of the proposed STDFS method is almost the same as the conventional IFFT method or even less, when the number of subcarriers is large and

the range for adjustment is small. Furthermore, the proposed method based on simplified circular convolution uses the received signal for computation directly. For the reason, the proposed SFDFS method should reduce the round-off errors in practice.

Table 4.1 The computational complexity comparison of fine symbol timing estimation.

Methods	No. of multiplications	No. of additions
CFDFS [25, 26] (Radix-2)	$N \times \log_2 N + N$	$2N \times \log_2 N$
Proposed DTDFS	N^2	$N(N-1)$
Proposed STDFS	$L \times N$	$L(N-1)$

N: FFT length   L: the range for fine search and adjustment

4.4 Sampling Clock Offset Synchronization

4.4.1 Sampling Clock Offset Estimation

4.4.1.1 Conventional Methods for Sampling Clock Offset Estimation [27,28]

Sampling clock offset results in a slow shift of sampling instants, and rotating the phase of data carried by subcarriers. For analyzing the sample clock offset, we define the normalized sampling clock offset as β in (4.25). The *n*-th received symbol with clock offset *β* can be expressed as (4.26)

$$\beta = \frac{T_s{'}-T_s}{T_s} \tag{4.25}$$

where T_s and $T_s^{`}$ are the transmitter sampling period and the receiver sampling period respectively.

$$R_{n,k} = H_k X_{n,k} e^{j2\pi k\beta n \frac{N+Ng}{N}} + N_{n,k} \tag{4.26}$$

$$X_{n,k} = |X_{n,k}| e^{j\phi_{n,k}}, \qquad X_{n-1,k} = |X_{n-1,k}| e^{j\phi_{n-1,k}} \tag{4.27}$$

where N is the number of subcarriers, and N_g is the length of guard interval.

The method for sampling clock synchronization adopted here for acquisition uses the phase difference between two consecutive training symbols [27] as shown in Equation (4.28).

$$
\begin{aligned}
Z_{n,k} &= R_{n,k} R_{n-1,k}^* \\
&= H_k X_{n,k} e^{j2\pi k\beta n \frac{N+Ng}{N}} \left(H_k X_{n-1,k} e^{j2\pi k\beta(n-1)\frac{N+Ng}{N}} \right)^* \\
&= |H_k|^2 |X_{n,k}|^2 e^{j(\phi_{n,k}+2\pi k\beta n \frac{N+Ng}{N})} e^{-j(\phi_{n-1,k}+2\pi k\beta(n-1)\frac{N+Ng}{N})} \\
&= |H_k|^2 |X_{n,k}|^2 e^{j(\phi_{n,k}-\phi_{n-1,k}+2\pi k\beta \frac{N+Ng}{N})}
\end{aligned}
\tag{4.28}
$$

In the equation, by conjugating the *(n-1)*-th symbol and multiply it with the *n*-th symbol, the channel phase will be eliminated and the phase difference can be calculated. With the information of the phase difference, the sampling clock offset can be estimated using the information with various feasible ways. (4.29), (4.30), and (4.31) show three possible estimation schemes.

Method 1 [27]:

$$\widehat{\beta} = \underset{\substack{k_1 \neq k_2 \\ k_1,k_2 \in pilot\ tones \\ carrying\ known\ data}}{average} \left[\frac{N}{2\pi(k_2-k_1)(N+Ng)} ((\angle Z_{n,k_2} - \phi_{n,k_2} + \phi_{n-1,k_2}) - (\angle Z_{n,k_1} - \phi_{n,k_1} + \phi_{n-1,k_1})) \right]$$

$$\tag{4.29}$$

Method 2 [28]:

$$Y_{right} = \underset{k \subset right}{average}[Z_{n,k}]$$

$$Y_{left} = \underset{k \subset left}{average}[Z_{n,k}]$$

$$\hat{\beta} = \frac{N}{2\pi(N + Ng)} \times \frac{1}{C}(\angle Y_{right} - \angle Y_{left}) \qquad (4.30)$$

Method 1 directly averages the overall phase-difference cases between any two pilot subcarriers and estimate the clock offset. Method 2 first separates the pilot carriers in two groups, then finds the difference of their averaged phase values, and finally estimates the clock offset.

4.4.1.2 The Proposed Method for Sampling Clock Offset Estimation

Method 1 treats the all phase-difference cases between any two pilot subcarriers equally and estimate the sampling clock offset by averaging all the offset estimate samples. In Equation (4.29), we can find the phase differences of various pilot pairs are divided by different constants. Since the constant is the distance the pilot carriers, the ability of eliminating noise effect increases with the distance between the pilot carriers.

Owing to that the ability of eliminating noise effect increases with the distance between the pilot carriers, we propose the method (4.31) by only using the most separated pilot carriers to estimate the sampling clock offset and simultaneously reduce the computational complexity.

Proposed Method :

$$\hat{\beta} = \left[\frac{N}{2\pi(k_{max} - k_{min})(N + Ng)} ((\angle Z_{n,k_{max}} - \phi_{n,k_{max}} + \phi_{n-1,k_{max}}) - (\angle Z_{n,k_{min}} - \phi_{n,k_{min}} + \phi_{n-1,k_{min}})) \right]$$

(4.31)

When the number of receiver antennas is more than one, each receiver antenna performs the sampling clock offset estimation independently first. Since all the reception data streams use the same sampling clock oscillator, one can average all the estimated clock offset samples and obtain a low-noise clock offset estimation, then finally performs the clock offset compensation for all the reception data streams with the same clock offset.

4.4.2 Sampling Clock Offset Compensation

As soon as the sampling clock offset has been estimated, there are various feasible schemes to compensate the offset, such as adjusting the sampling frequency of ADC in the continuous-time domain, or rotating the FFT outputs in frequency domain, or correcting the timing by using a resampling interpolator in discrete-time domain.

The VCXO (Voltage-controlled crystal oscillator) is used to control the sampling frequency of ADC. However, a VCXO usually has higher cost and higher noise jitter than a XO (Crystal oscillator). For these reasons, the scheme is not suitable for 802.11n.

When the sampling clock offset compensation is performed by rotating the FFT outputs in frequency domain, no over-sampling is needed and the timing error correction can be done by a single complex multiplication. However, this method works well only when a small frequency offset is guaranteed. Accurate and expensive

XOs are required at both transmitter and receiver sides. Furthermore, the cost of updating the rotor coefficients is quite high. If a look-up table is used for coefficient updating, it will require a large memory.

Various interpolators [29,30,31,32,33] have been proposed to usually perform the compensation by time-domain resampling and applied to the compensation structure, shown in Figure 4.6. With a free-running oscillator, timing correction can be done by some discrete-time signal processing techniques on sample sequence. A simple interpolator may be a finite impulse response filter (FIR) that produces a fractional delay. The interpolator coefficients are time-varying, because they depend on the timing error to correct. Due to the time-varying filtering, the FFT output data suffers a time-varying phase rotation and attenuation. Such distortion introduces additional noises, especially on high frequency subcarriers that are close to the Nyquist frequency. The noise can be reduced by over-sampling the received signal or by using a high-order interpolator. However, due to the high sampling rate and high computation complexity, those schemes might not be suitable for very high-rate systems.
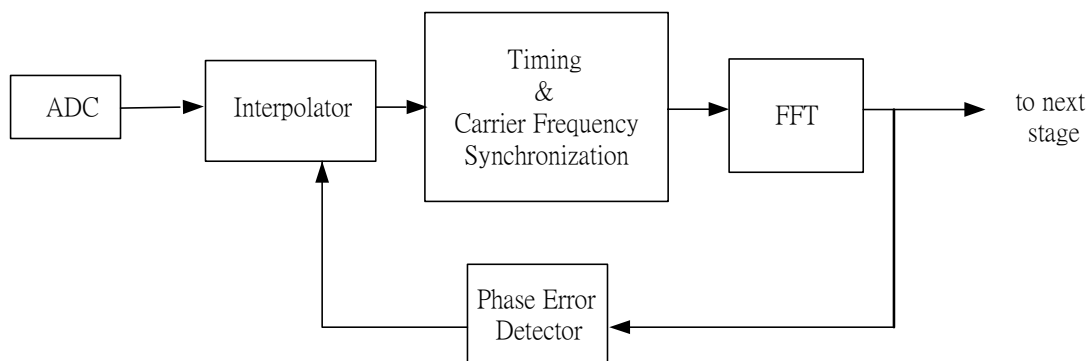

Figure 4.6 Structure of discrete-time resampling timing correction.

In the following subsections, we will find suitable interpolators with low complexity and high performance.

4.4.2.1 Designs of Digital Resampling [43]

Figure 4.7 shows an ideal fractional delay (FD) block, which can be viewed as a digital version of a continuous- time delay line. Therefore, the impulse response of an LSE (least square error) fractional delay filter is a shifted and sampled sinc function.

$$h(n) = \sin c(n - D) \qquad (4.32)$$

where n is the sample index and D is the delay with a fractional part $d = D\text{-}floor(D)$ and an integral part $floor(D)$.
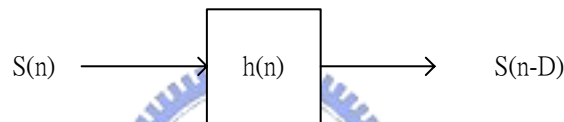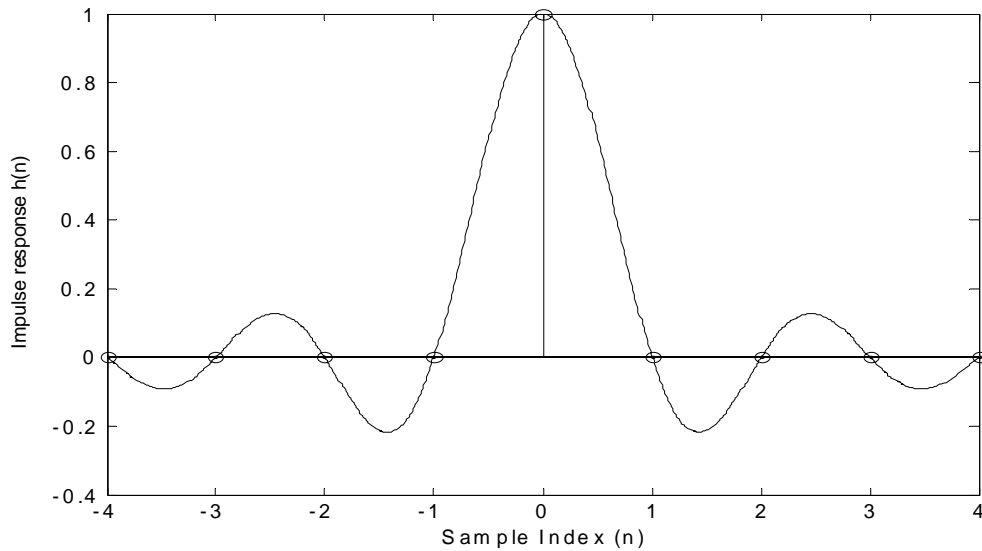

Figure 4.7 The discrete- time delay block.

(a)

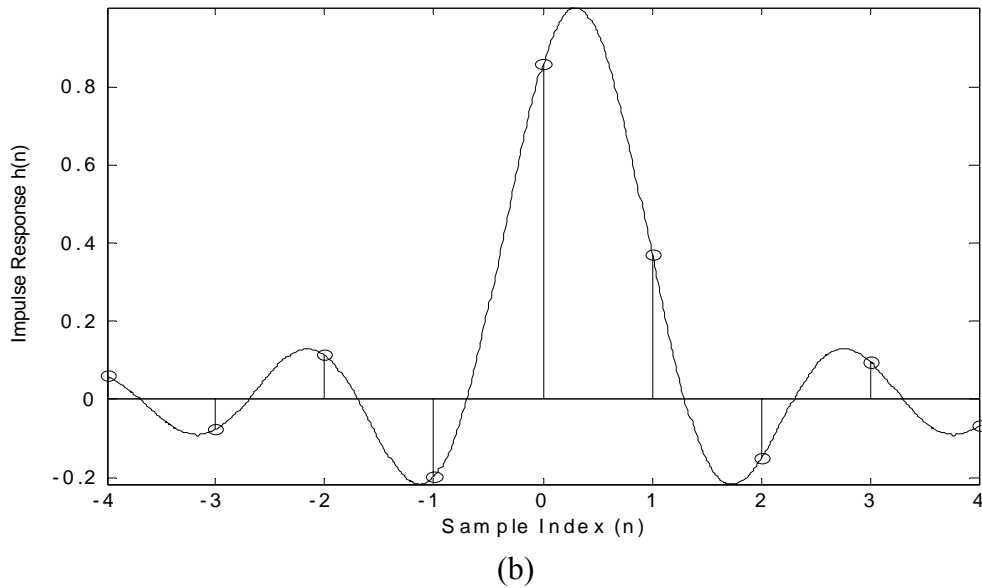**Figure 4.8** The impulse response of the sinc interpolator with the delay

(a) d=0.0, (b) d=0.3

Figure 4.8 shows the LSE sinc impulse response when *d=0* and *d=0.3*. Since the impulse response of sinc fraction delay filter is infinite-length, it should be truncated for practical applications. Since its impulse response is not absolutely summable and suffers from the well-known Gibbs phenomenum, the sinc FD filter is impractical with poor performance.

To design an efficient FD filter, finite-length, truncated and properly windowed sinc functions with small Gibbs phenomenon are desired. Signal interpolation is conceptually equal to a resampling operation on reconstructed signals in continuous-time domain. In order to obtain the reconstructed continuous-time signals, least-square interpolation, equirriple interpolation, windowed-sinc interpolation and polynomial-based interpolation are frequently used. In this work, we use various interpolators to perform the compensations, and assess theier performances when applied to IEE802.11n standard.

Since a resampling interpolator's coefficients are time-varying, the introduced

time-varying phase rotation and attenuation are difficult to be equalized. Before simulating these interpolators, let's check the frequency responses of some popular interpolators [43].

Figure 4.9 (a) (b) (c) (d) shows the responses of 4-tap fractional-delay FIR filters, which are obtained from several major windowed sinc functions. Figure 4.10 (a), (b), (c), and (d) shows the responses of various 4-tap fractional-delay FIR filters, which are designed by general least-squares approximation, equirriple approximation, Lagrange interpolation and Cubic B-spline interpolation, respectively. Since the received data will be demodulated and decoded in frequency domain, frequency responses of those interpolators are important. Due to time-varying coefficients, a fractional delay FIR has different frequency responses with respect to the fractional part d of input delay D.

The magnitude response error and phase delay error will introduce additional noise. Therefore, an interpolator with small phase error and small magnitude response error is desired. In Figure 4.9 and 4.10, we can find that the magnitude distortion of the Cubic B-spline interpolation is serious at the high frequency band especially. For the reason, the performance of the Cubic B-spline interpolation is poor in its application to IEEE 802.11n. We will simulate these interpolation designs in the next chapter and find suitable designs for IEEE 802.11n.
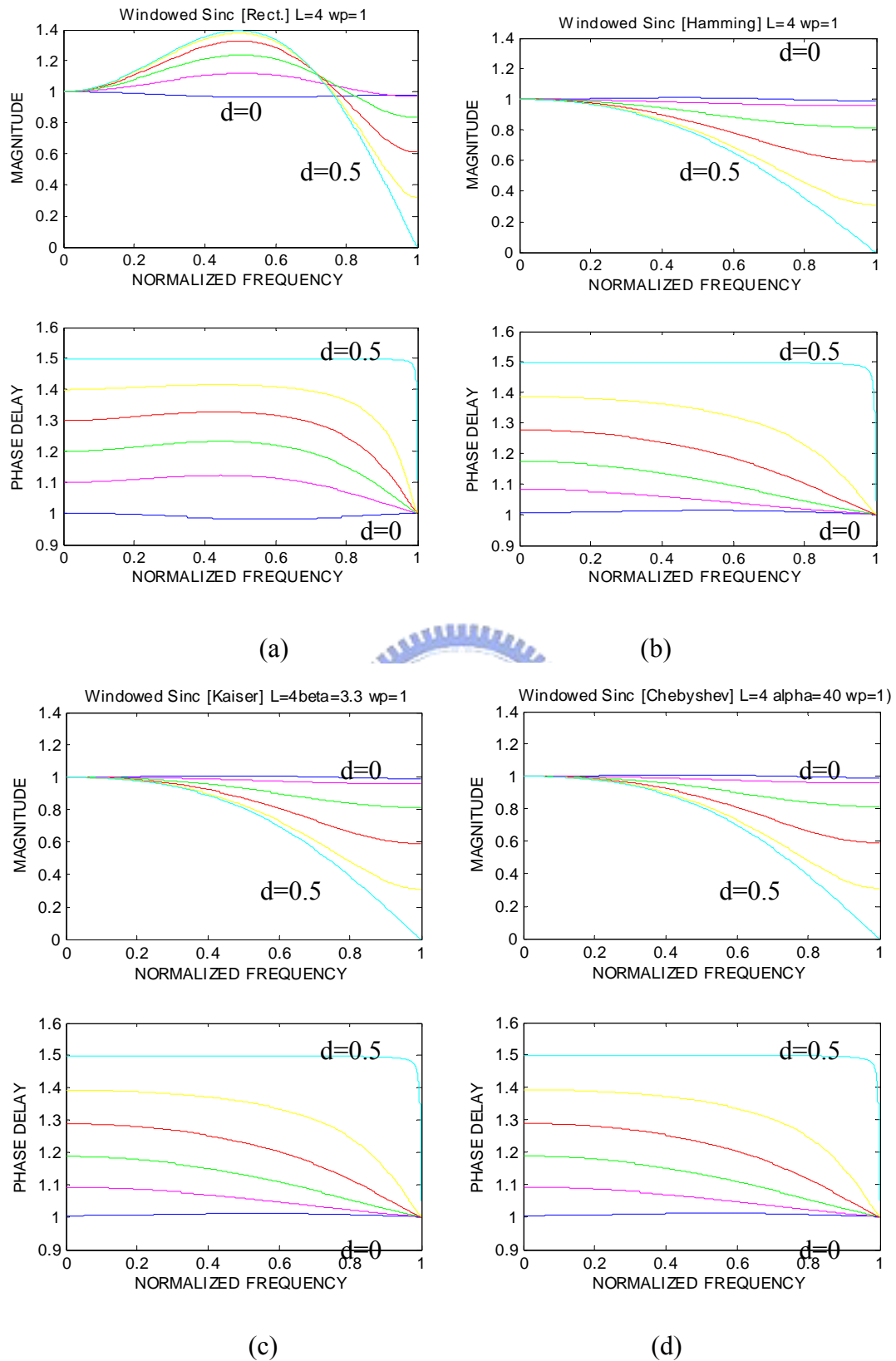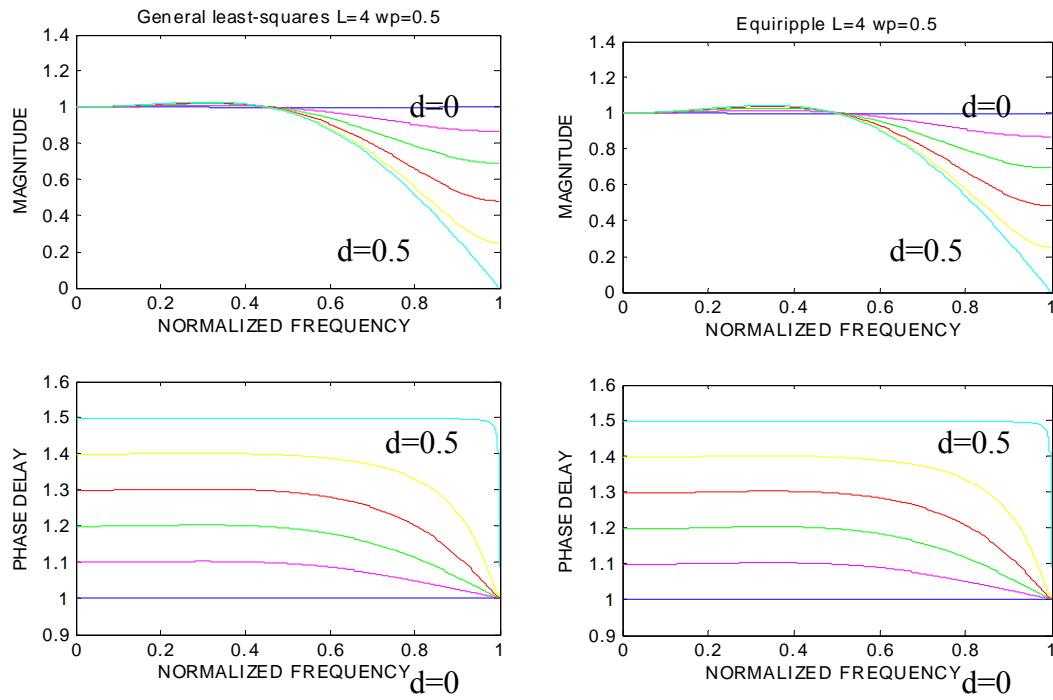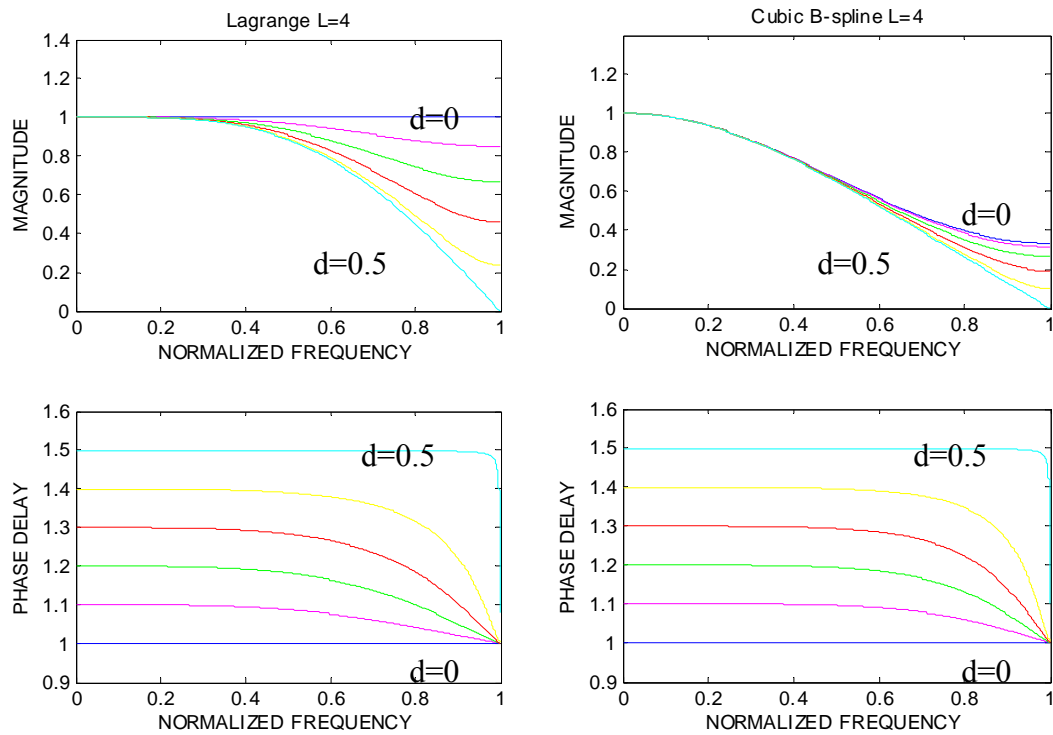
Figure 4.9 Frequency responses of windowed Sinc FIR FD filters: (a) Rectangular window; (b) Hamming window; (c) Kaiser window; (d) Chebyshev window.

Figure 4.10 Frequency responses of various FIR FD filters: (a) General least-squares

approximation; (b) Equirriple approximation; (c) Lagrange interpolation; (d) Cubic

4.4.2.2 Discrete-time Over-resampling Techniques [34,35,36]

The windowed-sinc and some other FD interpolators obtained from other key design techniques, discussed in the previous section are realized by resampling the reconstructed analog functions from the original discrete samples. Intuitively, pure digital signal interpolation is a fundamental approach for FD interpolation. One can perform digital "over-sampling" operations [35,36] to achieve high-resolution FD. With N-times oversampling, one can split a unit delay into N divisions, as shown in Figure 4.11. In the figure, $H(z)$ is an Nth-band low-pass filter. Note that the proto-type filter $H(z)$ can be obtained by various optimization techniques, such as least-square and equirriple approximations.

$$R(z) \longrightarrow \boxed{\uparrow N} \longrightarrow \boxed{H(z)} \longrightarrow \widetilde{R}(z)$$

Figure 4.11 Block diagram of a digital N-times over-sampling LTI interpolator.

An N-branch polyphase interpolator can be formed by polyphase decomposition techniques. In (4.33), z-transforms are used to derive a polyphase interpolator. Note that each branch is an L-tap filter, which is shown in Figure 4.12.

$$\begin{aligned}
H(z) &= h(0) + h(1)z^{-1} + ... + h(LN-1)z^{-(LN-1)} \\
&= H_0(z) + z^{-1}H_1(z^N) + z^{-2}H_2(z^N) + ... + z^{-(N-1)}H_{N-1}(z^N)
\end{aligned} \tag{4.33}$$

where $\quad H_i(z^N) = h(i) + h(i+N)z^{-N} + h(i+2N)z^{-2N} + ... + h(i+(L-1)N)z^{-(L-1)N} \quad$ and

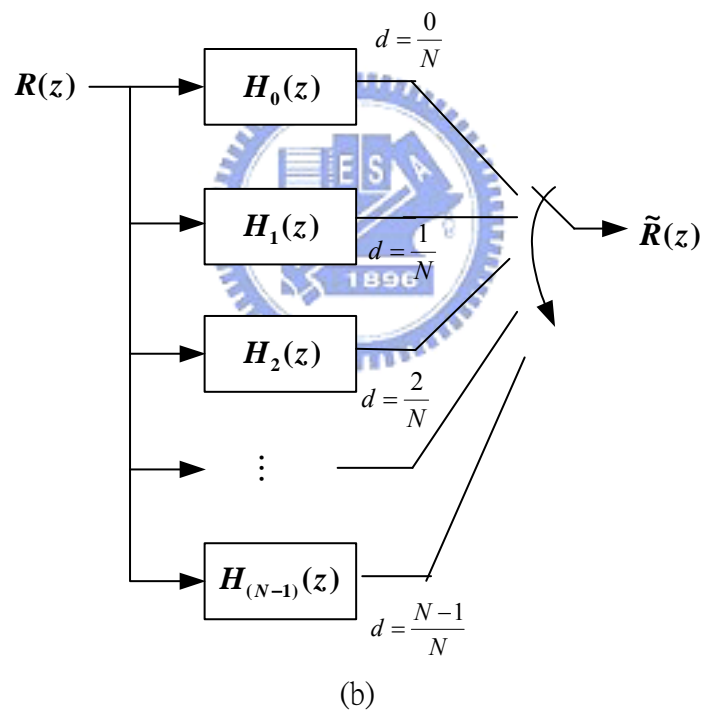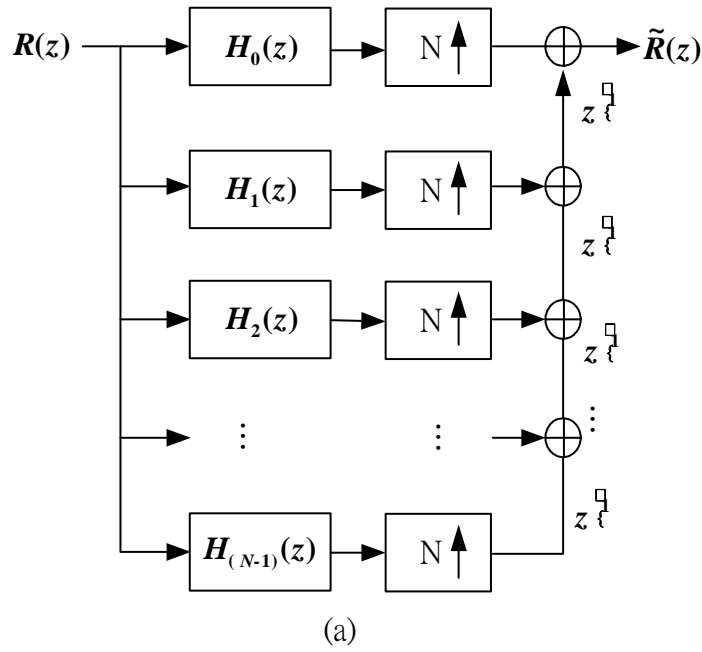$i \in \{0, 1, ..., (N-1)\}$.

(a)

(b)

Figure 4.12 An oversampling interpolator in polyphase form: (a) the simplified

polyphase structure; (b) and an equivalent polyphase interpolator with an output

commutator.

In the upsampling polyphsae structure, the kth branch is responsible for generating the samples with fractional delay of k/N, where $k \in \{0, 1, ..., (N-1)\}$. Although the upsampling operation is computationally intensive especially for high-resolution fractional delays, the complexity can be greatly reduced by only considering those required FD sampled. The upsampling-based digital FD interpolators are disadvantageous in that it can not realize arbitrary delay. To solve this problem, a well-known approach [28] is to approximate each branch's polyphase filter coefficient as a pth-order polynomial of fractional delay d, as shown in (4.34). Usually, the approximating polynomial of the filter coefficients is obtained by fitting them in the least-squares sense.

$$h(n) = \sum_{m=0}^{p} c_{m,n} d^m, \ n \in \{0, 1, 2, ..., (L-1)\} \tag{4.34}$$

where $c_{m,n}$ are real-valued coefficients of those approximating polynomials. Then the resulting interpolator will be able to produce continuous delay. In addition, the well-known Farrow structure can be applied to the approximating polynomials for low-complexity realization, which will be shown in the next section.

4.4.2.3 Farrow Structure [34,35,42]

Increasing filter length of an interpolator results in better phase and magnitude responses. However, the filter length of an interpolator cannot be arbitrarily long. Here are two reasons: First, the length of composite channel, including the transmission channel and the interpolator of the receiver, must be shorter than the guard interval to avoid ISI. Second, longer interpolator length means higher computational and hardware

complexities. After some trials, we found that a 4-tap interpolator is sufficient. To further reduce the complexity, Farrow structure is applied to implement the interpolator.

Since the fractional delay to be compensated varies over time. The design of high-quality FD filter is difficult. Farrow [35] suggested that every filter coefficient of the FIR FD filter could be expressed as a *P*th-order polynomial in the variable delay parameter *D*.

Design a set of filters approximating a fractional delay in the desired range, $0 \le d \le 1$, and then approximate each coefficient as a *P*-order polynomial of fractional delay. The transfer function of the filter becomes

$$
\begin{aligned}
H(z) &= \sum_{n=0}^{L-1} h(n) z^{-n} \\
&= \sum_{n=0}^{L-1} \left[\sum_{m=0}^{p} c_{m,n} d^m \right] z^{-n} = \sum_{m=0}^{p} \left[\sum_{n=0}^{L-1} c_{m,n} z^{-n} \right] d^m \\
&= \sum_{m=0}^{p} C_m(z) d^m
\end{aligned}
\tag{4.35}
$$

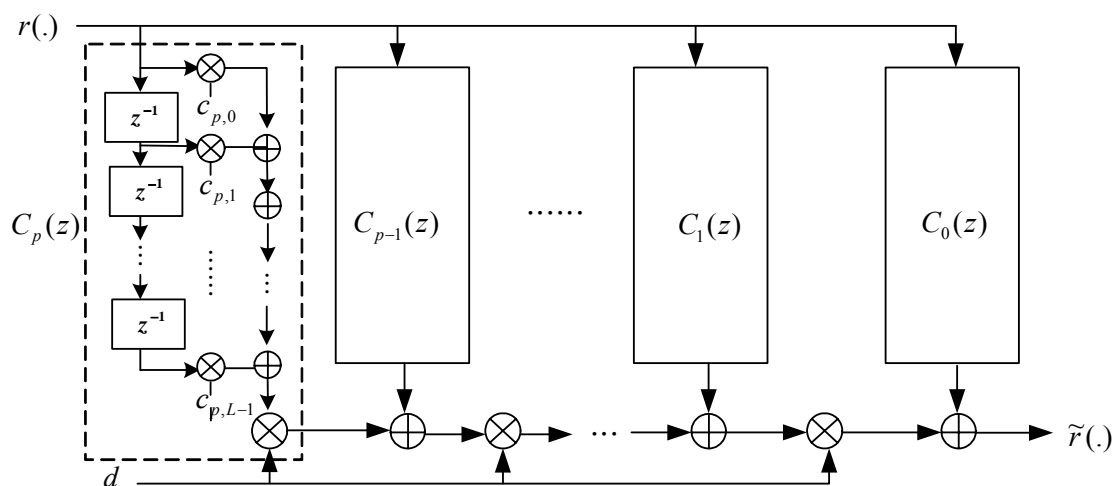This is the well-known Farrow structure, which is illustrated in Figure 4.13.

Figure 4.13 Farrow structure of an interpolator.

53

An interpolator with Farrow polynomial approximation coefficients is able to produce continuous delays. The delay accuracy is affected by the filter length L and the Farrow polynomial order P. After some trials, we found that it is sufficient for an interpolator with $L=4$ and $P=3$. Note that polynomial-based interpolator, such as Lagrange interpolation and B-spline interpolation, can be directly realized by the Farrow structure without further approximation.

Chapter 5

Simulation Results and Comparisons

In this chapter, we will evaluate the performance of the discussed and proposed synchronization techniques, by applying them to IEEE 802.11n system, and compare the performances among various methods.

5.1 Simulation Environment

As for the simulation environment, AWGN channel and multi-path channel are assumed. Since IEEE 802.11 uses 5GHz band as the major carrier, we utilize the power delay profile measured in indoor environment at 5.3 GHz by using a wideband sounder

for simulations. Table 5.1 shows the model parameters.

Table 5.1 Channel Model Parameters [37] for simulations.

Tap No.	Delay (ns)	Power (dB)	Amplitude Distribution	Doppler Spectrum
1	0	0	Rayleigh	Classical / Flat
2	36	-5	Rayleigh	Classical / Flat
3	84	-13	Rayleigh	Classical / Flat
4	127	-19	Rayleigh	Classical / Flat

We assume the total transmit power from the multiple transmit antennas is the same as the transmit power from the single transmit antenna. It is also assumed the fading amplitudes from each transmit antenna to each receive antenna are mutually uncorrelated Rayleigh fading, and the averaged signal power at each receive antenna from each transmit antenna are the same.

Table 5.2 shows the simulated MIMO system parameters. Low mobility is considered in our simulations. Therefore, the Doppler shift frequency is set to 150Hz which corresponds to a mobility of 32 km/hr. We simulate some carrier frequency offset cases from 0.6 to 1.9 carrier spacing, and find the results are almost the same. For the reason, the carrier frequency offset is assumed as 1.35 carrier spacing in the following simulation environment. As for sampling clock offset, we simulate the clock offset by Lagrange interpolator with 50 interpolation taps, and try 50, 100, and 200 ppm cases, and find the results are almost the same. Therefore, the sampling clock offset is set to 100 ppm in the following simulations.

Table 5.2 Simulated MIMO system parameters.

Sample period	50 ns
Total number of carriers	64
The number of pilot carriers	4
The number of data carriers	48
Symbol period	4μs
Guard interval	0.8μs
Modulation	QPSK
Sampling frequency	20 MHz
Carrier spacing	312.5 kHz
Doppler shift frequency	150 Hz
Carrier frequency offset	1.35 carrier spacing
Sampling clock offset	100 ppm
Max. no. of T_x/R_x antennas	4/3

5.2 Performance of Frame Detection

Figure 5.1 shows the frame timing error probability due to frame detections. When the estimated frame timing is over 32 samples away from the exact frame start, we define it is a failed frame detection. The probability of failed frame detections is called the frame timing error probability. As the number of transmitter antennas is increased, the cyclical shift will also increase the difficulty of detecting the frame start. Even in

this condition, we can see that the probability is close to zero when the SNR is higher than 5 db.
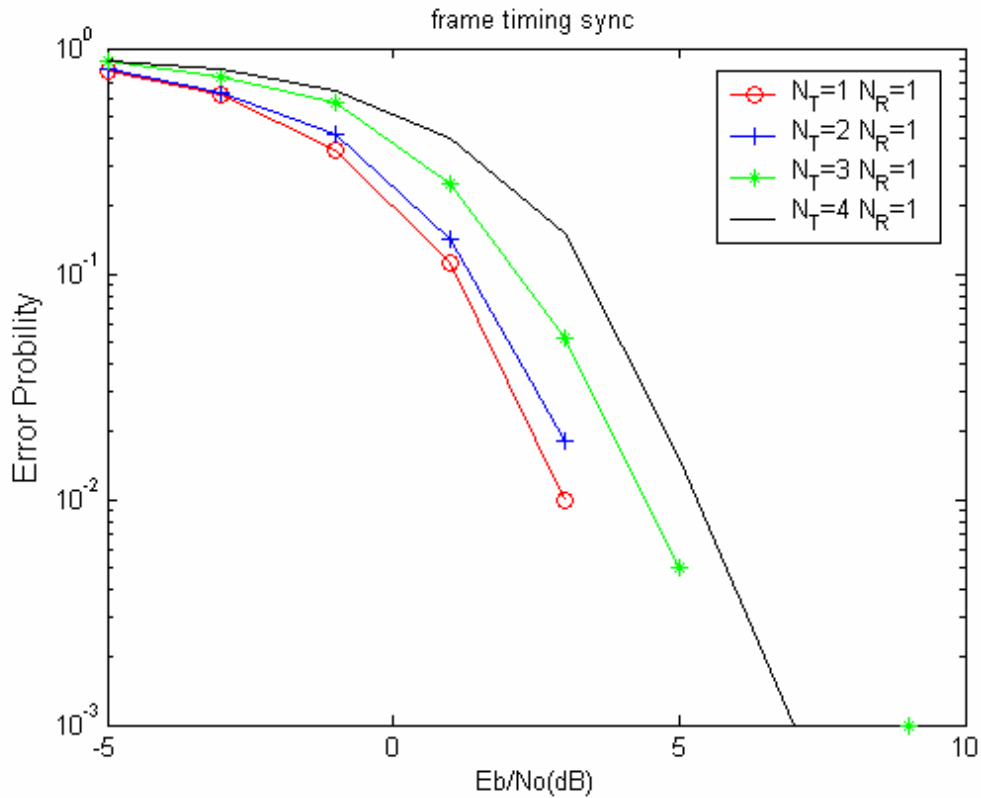

Figure 5.1 The frame timing error probability vs. SNR due to the frame detection for various numbers of T_x antennas.

Figure 5.2 shows the estimated frame timing distribution due to the frame detection when the SNR is 10 db. The numbers of receiver antennas are all one. When the number of receiver antennas is more than one, the receiver antennas perform the frame detections independently. We can detect the frame start successfully, even if the number of spatial data streams is up to four.
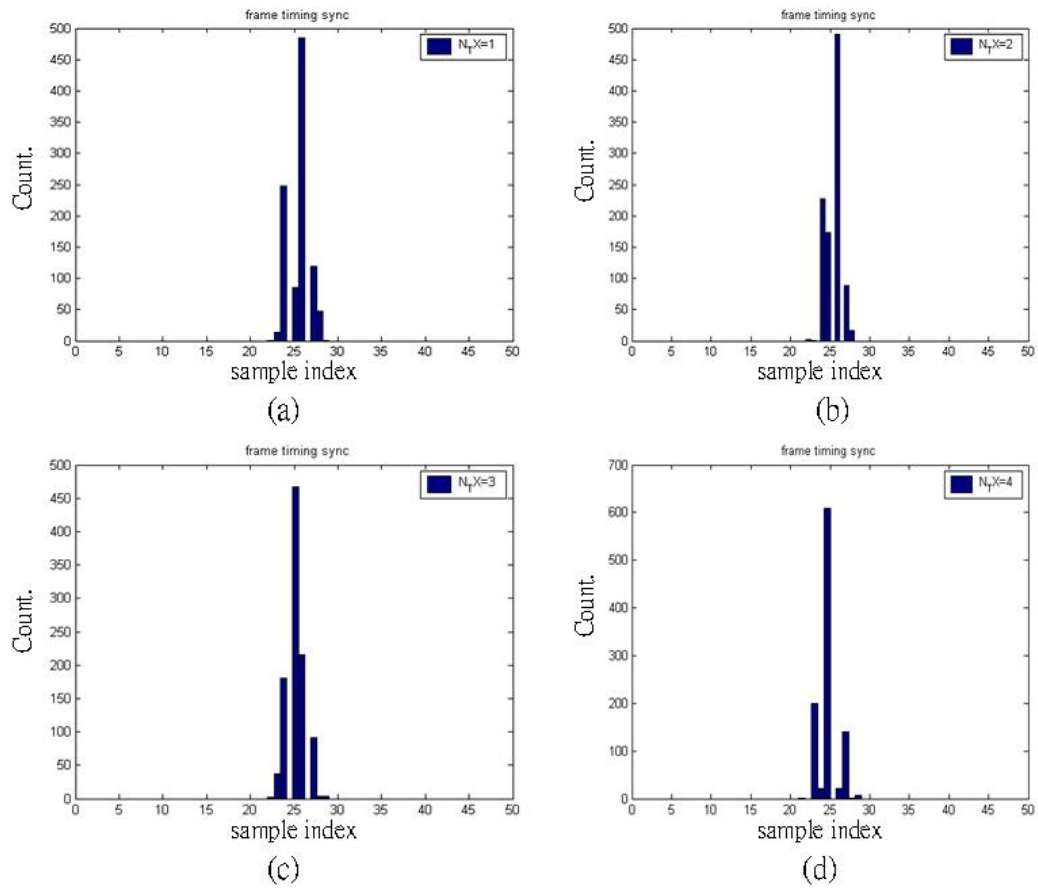
Figure 5.2 The estimated frame timing distribution due to the frame detection, SNR = 10 db, (a) $N_T X=1$. (b) $N_T X=2$. (c) $N_T X=3$. (d) $N_T X=4$.
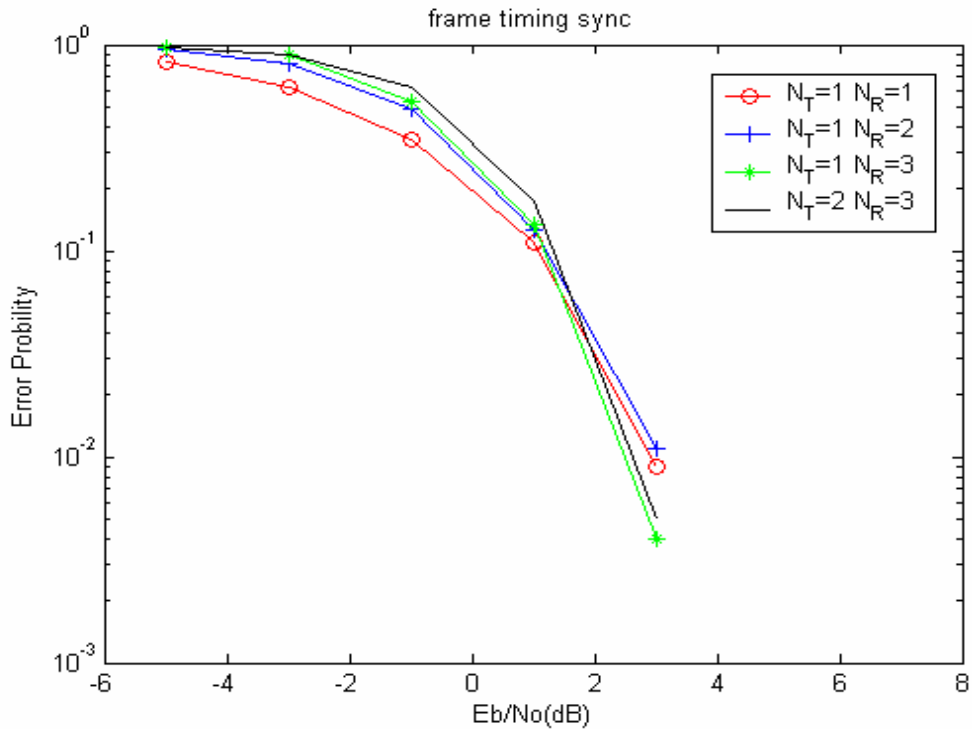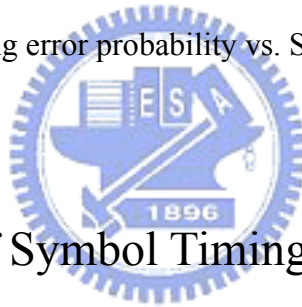
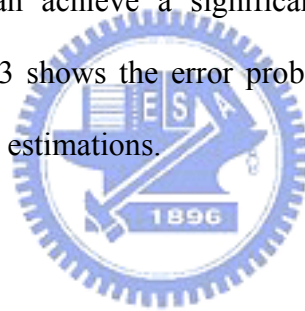**Figure 5.3** The frame timing error probability vs. SNR due to the frame detection.

5.3 Performances of Symbol Timing Synchronization

For the reason mentioned above, the performance decreases slightly when the number of transmitter antenna increases. Figure 5.4 shows the probability of timing errors when there are more than 5 samples estimated error. We can find that the method we used for the coarse symbol timing estimation still works well even if the number of the spatial data stream is up to four.

Figure 5.5 shows the comparison among the three methods discussed in chapter 4 for the fine symbol timing estimation. The DTDFS method achieves the similar performance with the conventional CFDFS method. Its major drawback is the high computation complexity compared with the conventional CFDFS method. However,

the STDFS method has a smaller complexity than the previous two methods. Since the search range of fine symbol synchronization can be reduced to +/- 5 samples by the coarse symbol timing estimation, we can reduce the computation complexity significantly. The performance of the simplified time-domain method is also similar to the other two methods mentioned above as shown in Figure 5.5.

In Figure 5.7 and 5.8, it is assumed that the number of receiver antennas is one and the SNR is 10db. Figure 5.6 shows the MSE comparison between the coarse and fine symbol timing estimations by using the double-sliding-window method and the STDFS method, respectively. In this figure, the numbers of the transmit antennas and receive antennas are set to two and three, respectively. In these figures, we can find that the fine symbol timing estimation can achieve a significant improvement of the timing estimation accuracy. Table 5.3 shows the error probability comparison between the coarse and fine symbol timing estimations.
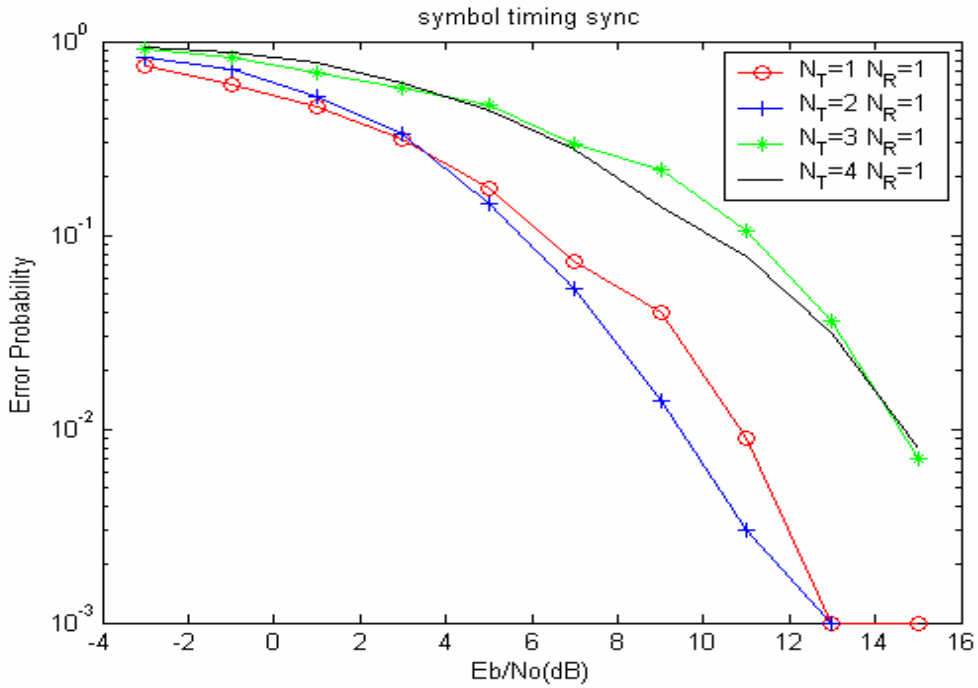
Figure 5.4 The symbol timing error probability vs. SNR due to the coarse symbol timing estimation with the double-sliding-window method for various T_x antenna numbers.
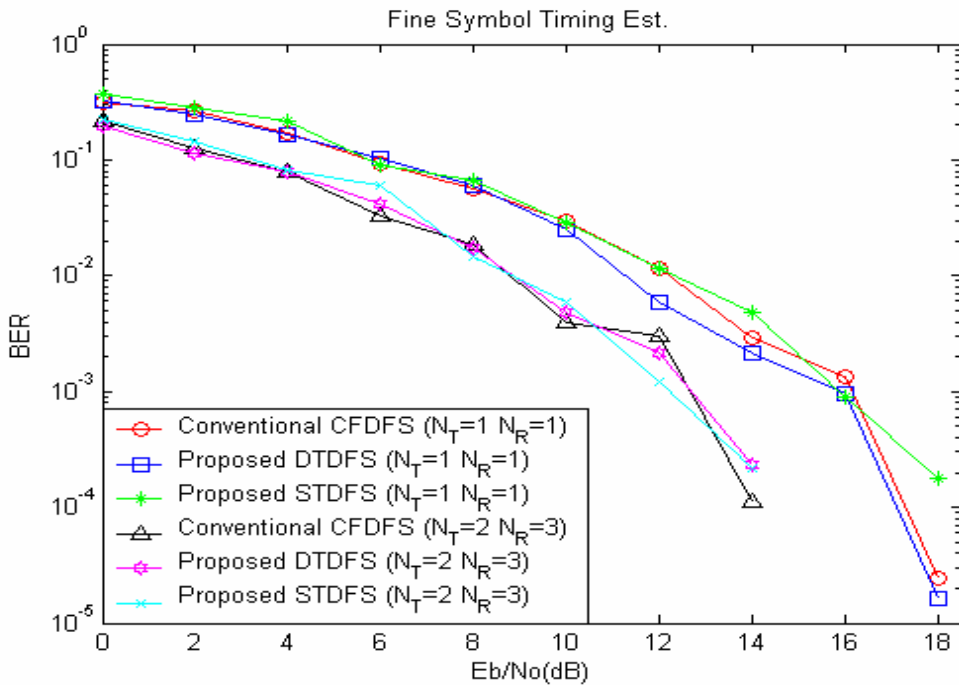

Figure 5.5 BER performance vs. SNR comparison of the discussed methods for fine symbol timing estimation.
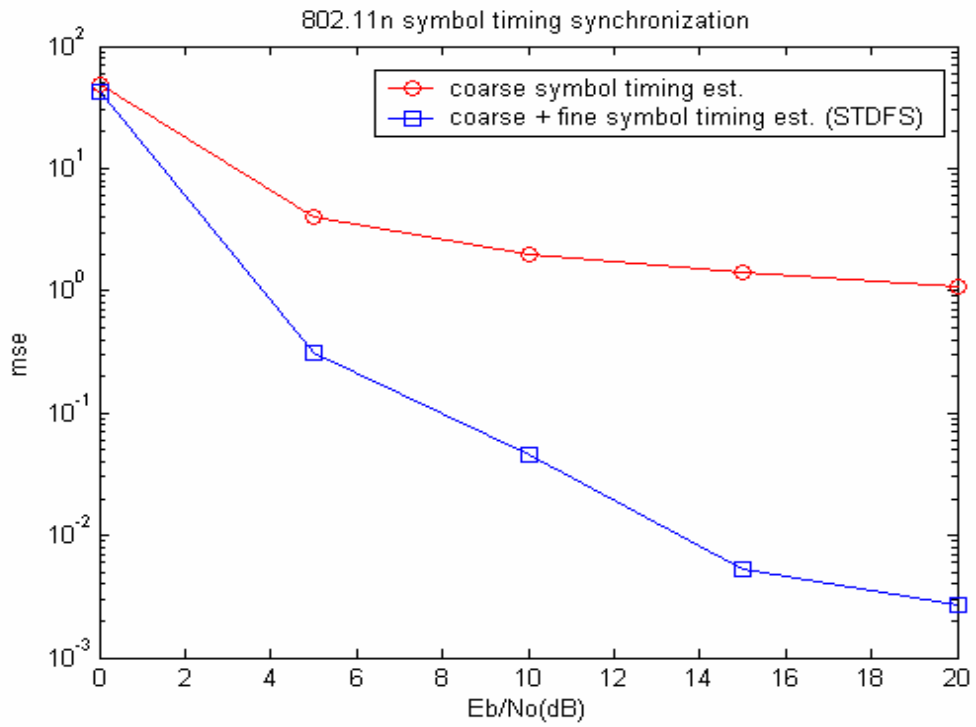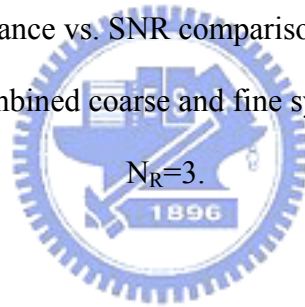
Figure 5.6 MSE performance vs. SNR comparison between the coarse symbol timing estimation and the combined coarse and fine symbol timing estimation, N_T=2, N_R=3.
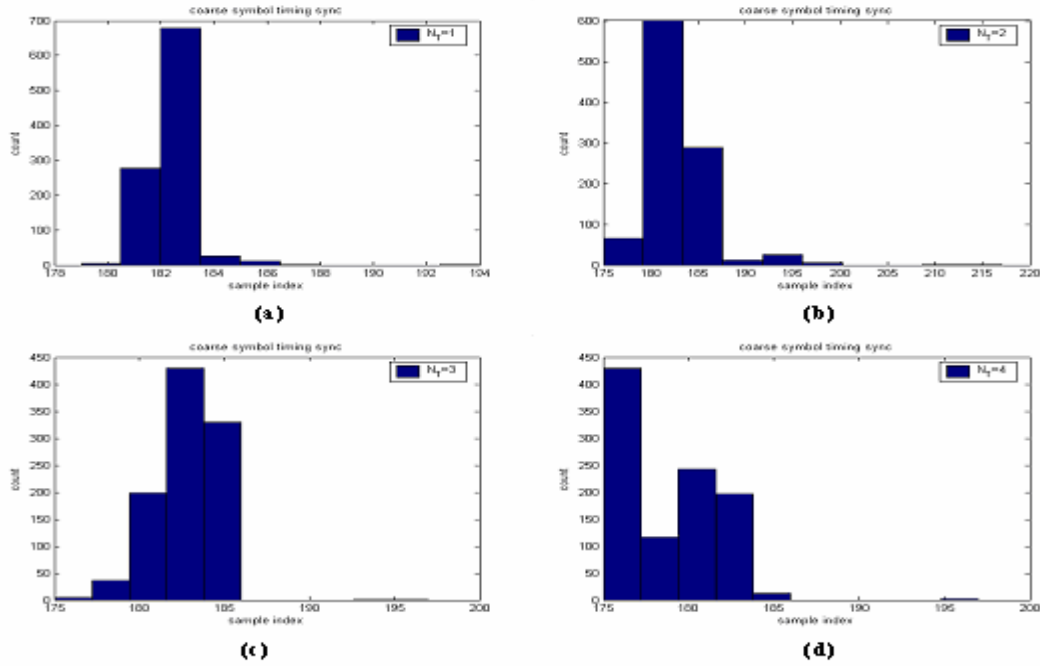
Figure 5.7 The estimated symbol timing distribution due to the coarse symbol timing synchronization, SNR = 10 db, (a) N_T=1. (b) N_T=2. (c) N_T=3. (d) N_T=4.
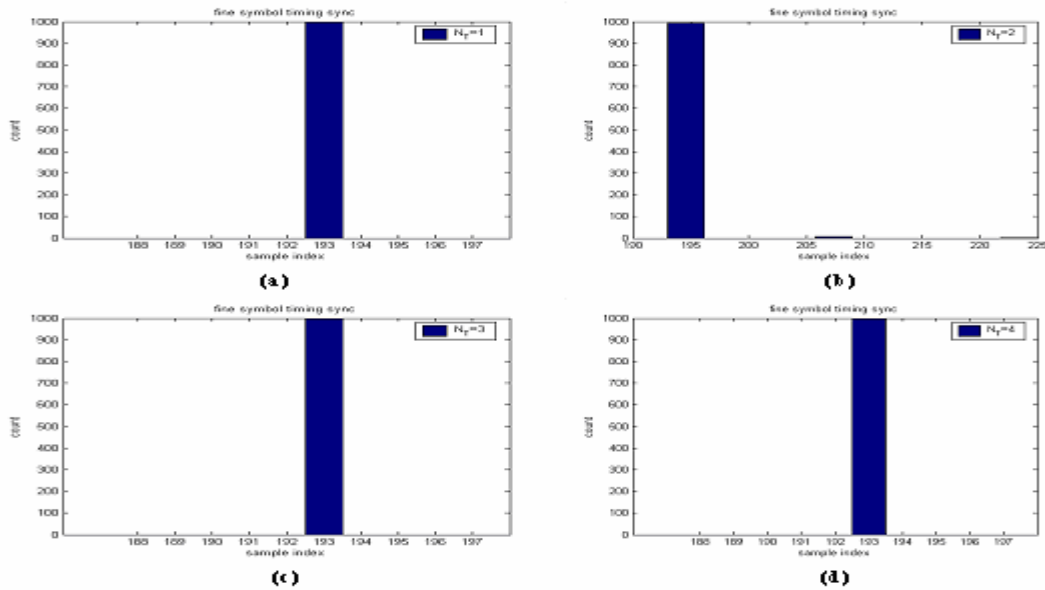

Figure 5.8 The estimated symbol timing distribution due to the fine symbol timing synchronization with STDFS method, SNR = 10 db, (a) N_T=1. (b) N_T=2. (c) N_T=3. (d) N_T=4.

Table 5.3 The error probability of the symbol timing estimation.

(a) The error is within +/-3 samples

$N_T \times N_R$ SNR	1×1		2×3		4×3	
	Coarse	Fine	Coarse	Fine	Coarse	Fine
0	0.5787	0.4260	0.3413	0.1720	0.4633	0.2480
5	0.2013	0.0847	0.1100	0.0127	0.1840	0.0213
10	0.0133	0.0047	0.0180	0.0020	0.0220	0.0027
15	0	0	0.0027	0	0	0
20	0	0	0	0	0	0
25	0	0	0	0	0	0

(b) The error is within +/-1 samples

$N_T \times N_R$ SNR	1×1		2×3		4×3	
	Coarse	Fine	Coarse	Fine	Coarse	Fine
0	0.7407	0.4553	0.6847	0.1987	0.6987	0.2773
5	0.4513	0.0933	0.5020	0.0193	0.4380	0.0340
10	0.1433	0.0053	0.3987	0.0027	0.2500	0.0087
15	0.0073	0	0.3407	0.0013	0.0733	0.0020
20	0	0	0.2713	0.0007	0.0053	0
25	0	0	0.1480	0	0	0

5.4 Performance of Carrier Frequency Offset Synchronization

Here, we test the three discussed methods. We can see the improvement of the proposed smoothed method clearly in Figure 5.9. Since the other two methods use only one point to estimate the frequency offset, they have high noise level. The new smoothed method can significantly reduce the noise effect. However, the performance improvement comes at extra hardware cost.
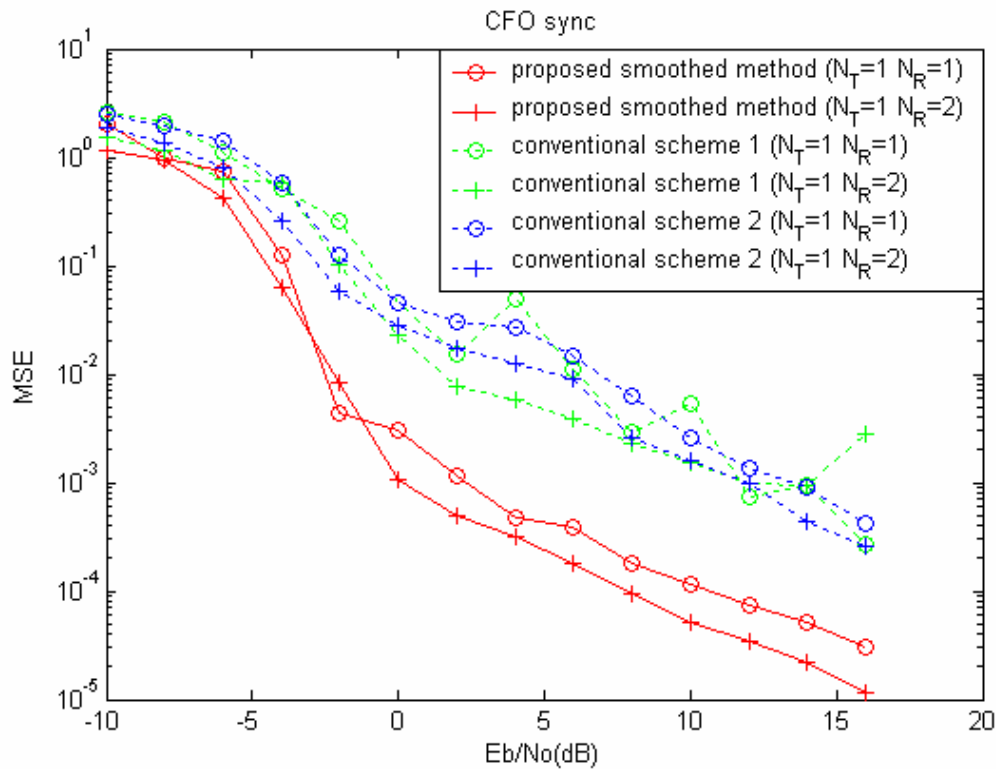

Figure 5.9 MSE performances vs. SNR comparison of the proposed new method and other conventional methods for the carrier frequency offset estimation.

Figure 5.10 shows the performance of the coarse frequency offset estimation and the overall combined coarse and fine carrier frequency offset estimation, using the proposed smoothed method.

Figure 5.10 MSE performances vs. SNR comparison of the coarse frequency offset estimation and the overall combined coarse and fine carrier frequency offset estimation, using the proposed smoothed method.
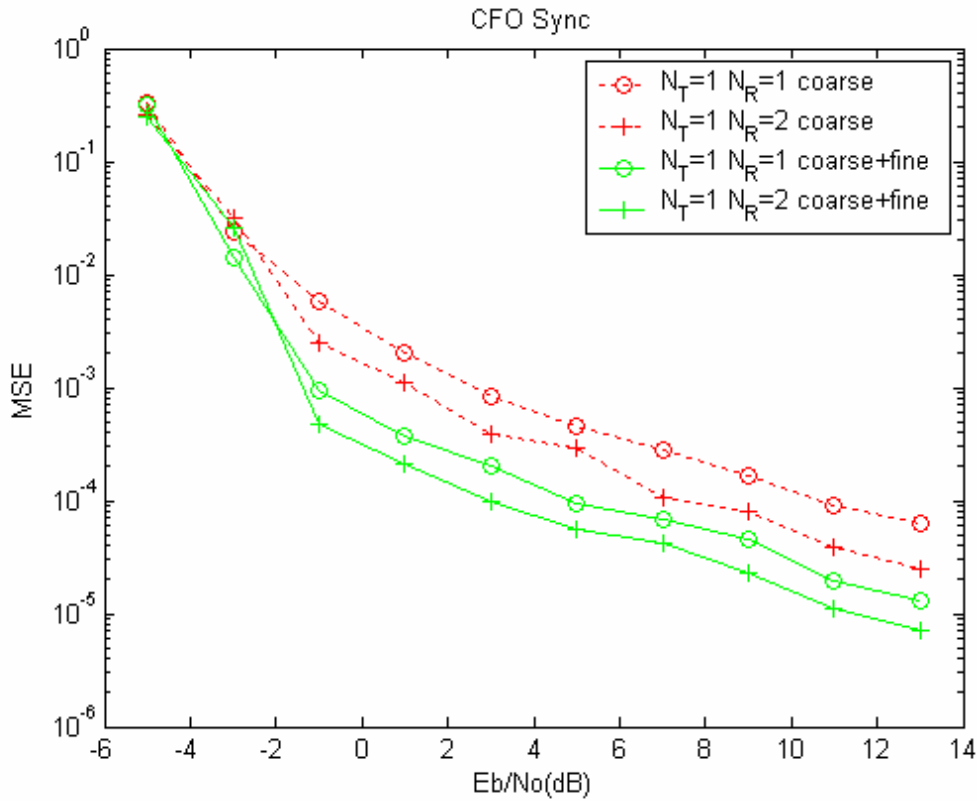
Figure 5.11 shows the performance of the coarse carrier frequency offset estimations. Figure 5.12 shows the performance of the coarse and fine carrier frequency offset estimations. In Figures 5.11, 5.12, we can find that the performance will be better when the number of the receiver antenna increases. However, there is no improvement when the number of the spatial data streams increases.
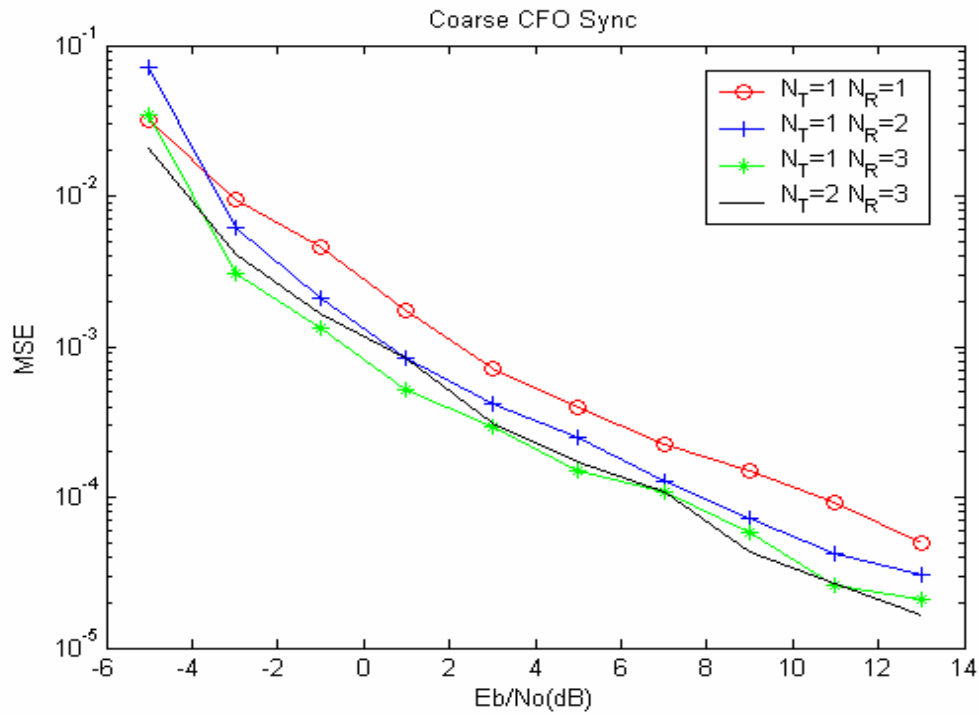
Figure 5.11 MSE performances vs. SNR of the coarse carrier frequency offset

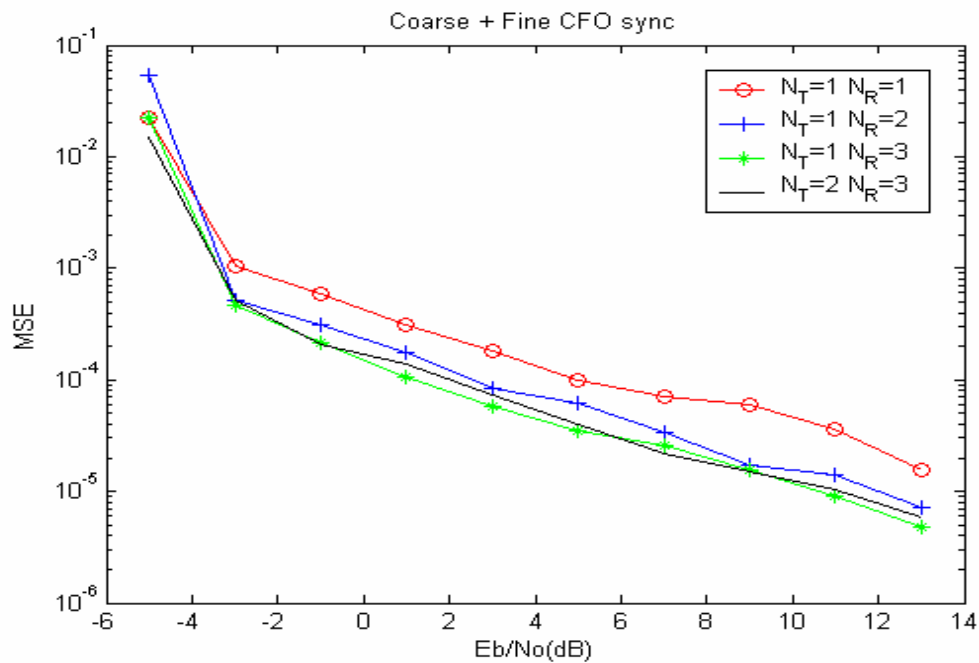estimation, using the proposed smoothed method.

Figure 5.12 MSE performances vs. SNR of the overall combined coarse and fine

carrier frequency offset estimation, using the proposed smoothed method.

5.5 Performance of Sampling Clock Offset Synchronization

5.5.1 Performance of Sampling Clock Offset Estimation

In Section 4.4.1, we discussed three feasible methods for the sampling clock offset estimation. The fist one directly averages all the possible cases to estimate the clock offset. For lower complexity considerations, the second method separates the pilot carrier in two groups first, then finds each group's averaged phase and finally obtains the offset estimation from the phase difference. The proposed third method reduces the computational complexity by using the most apart pilot carriers to estimate the sampling clock offset.

Figure 5.13 shows the comparison between the three methods. It shows that the new method achieves the similar performance compared to the other two methods, with less computational complexity.
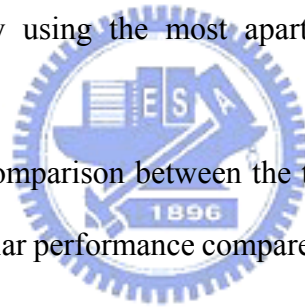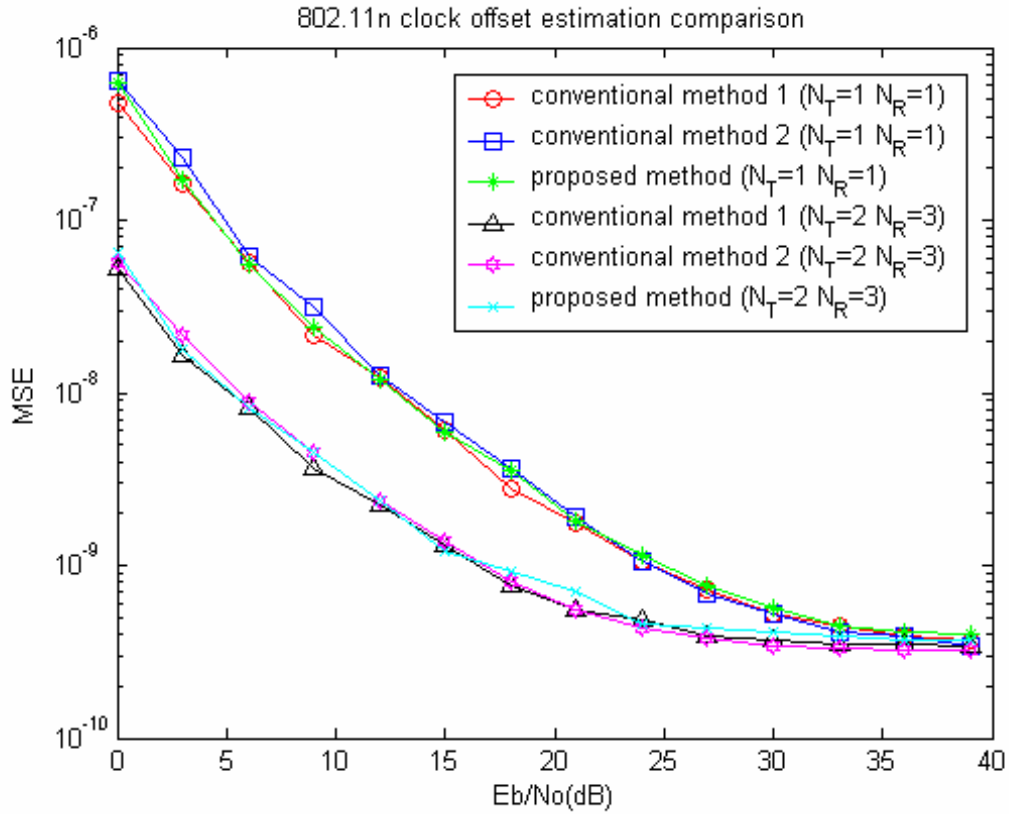
Figure 5.13 MSE performance vs. SNR comparison of the new and two conventional
methods for sampling clock offset estimation.

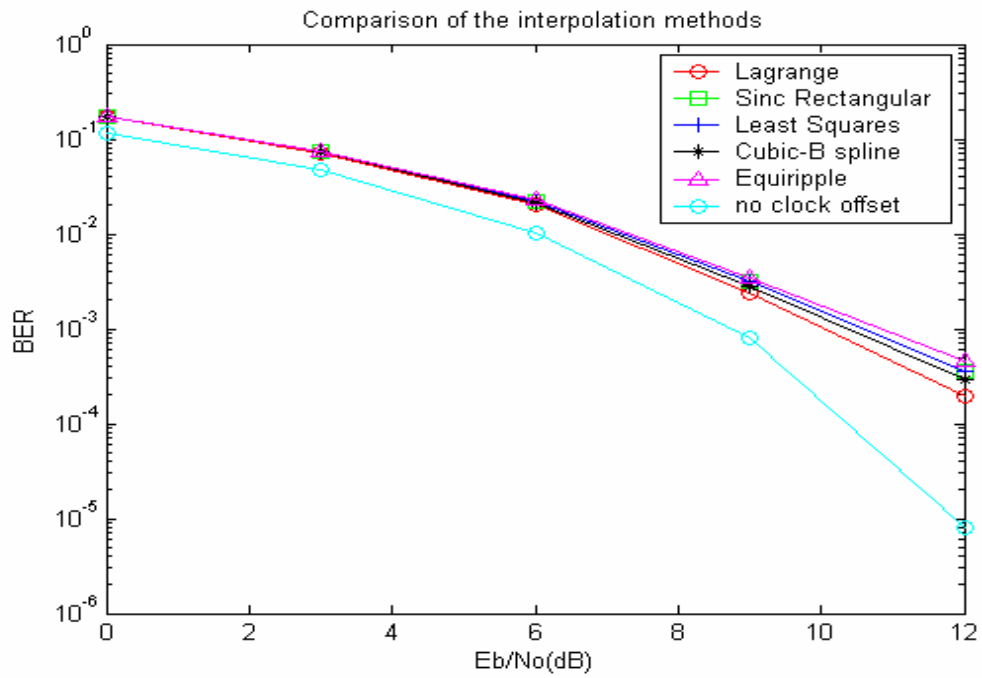# 5.5.2　Performance　of　Sampling　Clock　Offset Compensation

In Figure 5.14 – 5.17, we can find the performances of various interpolators are similar when $N_T \leq 2$. However, the performance of the Cubic B-spline interpolator is poor when $N_T = 3$ or 4, because of serious magnitude distortions at the high frequency band. Hence, the Cubic B-spline interpolator is less suitable for IEEE 802.11n systems.

Although the Cubic B-spline is not suitable for IEEE 802.11n system, it has good performance in some other systems, which adopt bit loading process, because of coherent magnitude responses.

Furthermore, the performances of windowed-sinc interpolator with rectangular window, least-square interpolator, and equiripple interpolator are slightly better than the other interpolators. Table 5.4 shows the computational complexity comparison of various interpolator designs [43]. One can find that the computational complexities of polynomial-based interpolators, Lagrange and Cubic B-spline, are slightly lower than the other interpolators. Unfortunately, the Cubic B-spline interpolator performance is poor when applied to IEEE 802.11n system. However, the Lagrange interpolator has similar performance compared to the three interpolators. For the reason, we suggest that Largange interpolator design is more suitable when applied to IEEE 802.11n standard.

Table 5.4 The computational complexity comparison of various interpolator designs for sampling clock offset compensation. L=4, P=3. [43]

Interpolator type	$+/-$	$\times$ constant	$<<$ *or* $>>$	$\times$
Lagrange	11	2	4	3
Cubic B-spline	11	4	3	3
Equiripple General LS. Windowed- Sinc	15	16	0	3

(a)
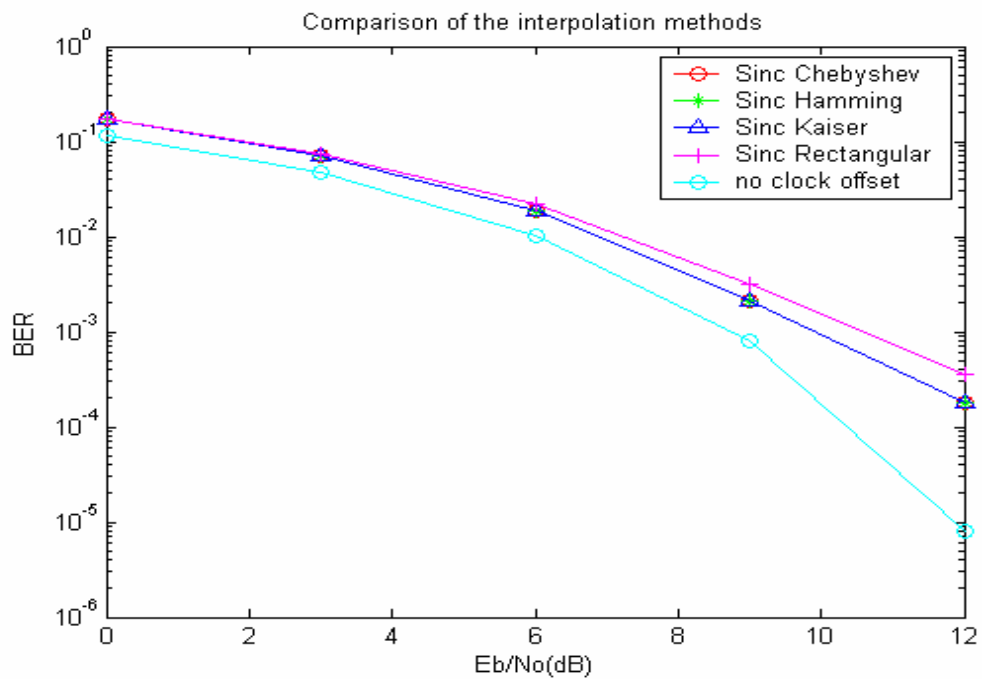
(b)

Figure 5.14 BER performance vs. SNR comparison of the major resampling

interpolators for sampling clock offset compensation. $N_T=1$, $N_R=1$.

(a) polynomial-based interpolators. (b) windowed sinc interpolators.

(a)

(b)

Figure 5.15 BER performance vs. SNR comparison of the major resampling

interpolators for sampling clock offset compensation. N_T=2, N_R=1.

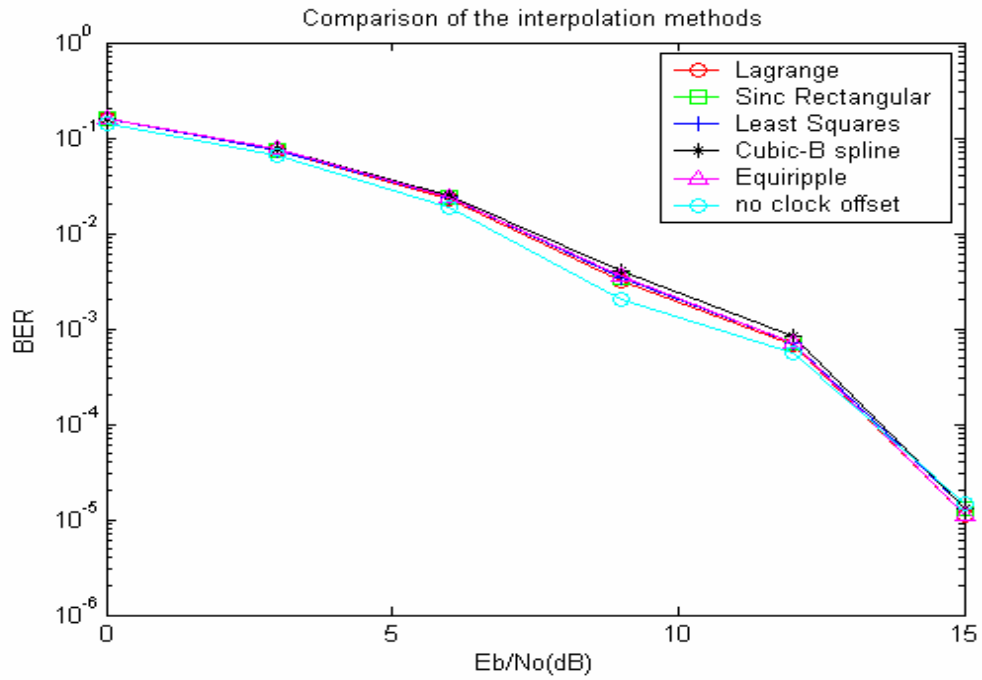(a) polynomial-based interpolators. (b) windowed sinc interpolators.

73

(a)

(b)

Figure 5.16 BER performance vs. SNR comparison of the major resampling

interpolators for sampling clock offset compensation. N_T=3, N_R=1.

(a) polynomial-based interpolators. (b) windowed sinc interpolators.
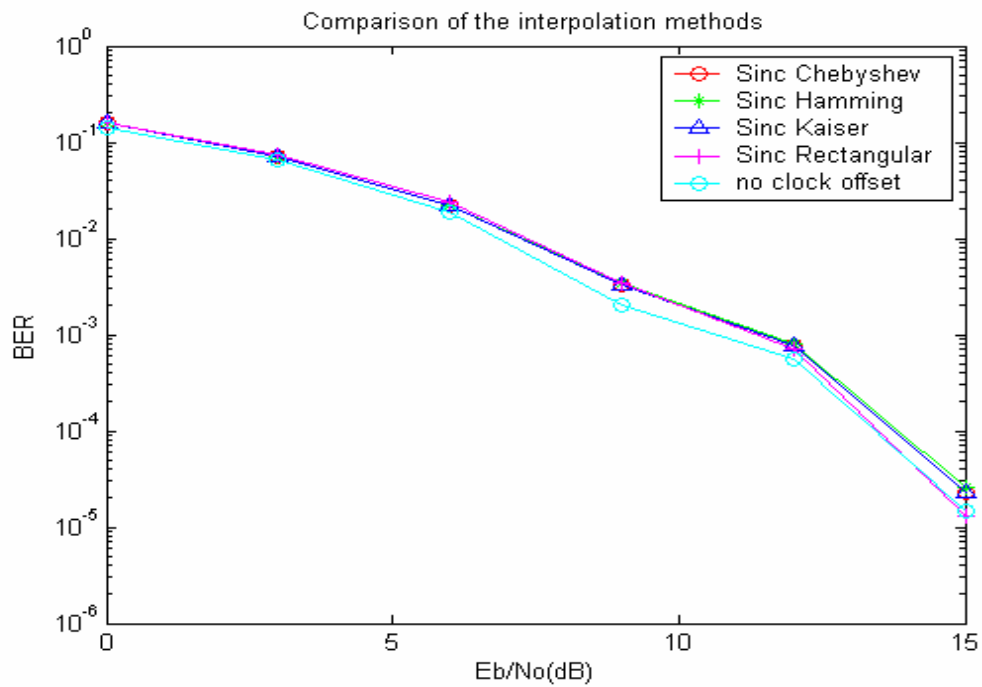
(a)

(b)

Figure 5.17 BER performance vs. SNR comparison of the major resampling

interpolators for sampling clock offset compensation. N_T=4, N_R=1.

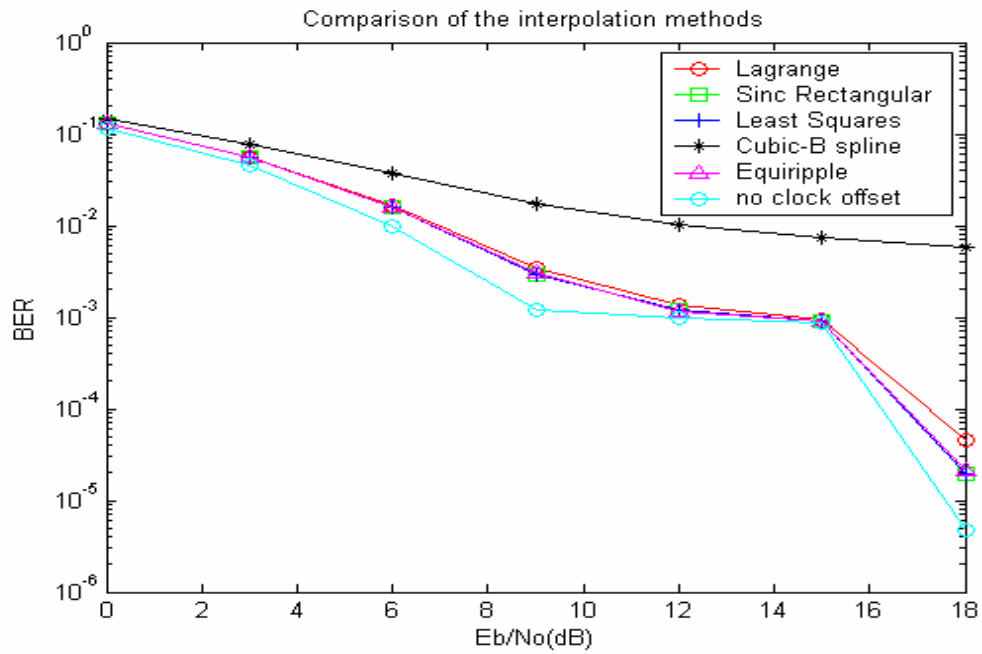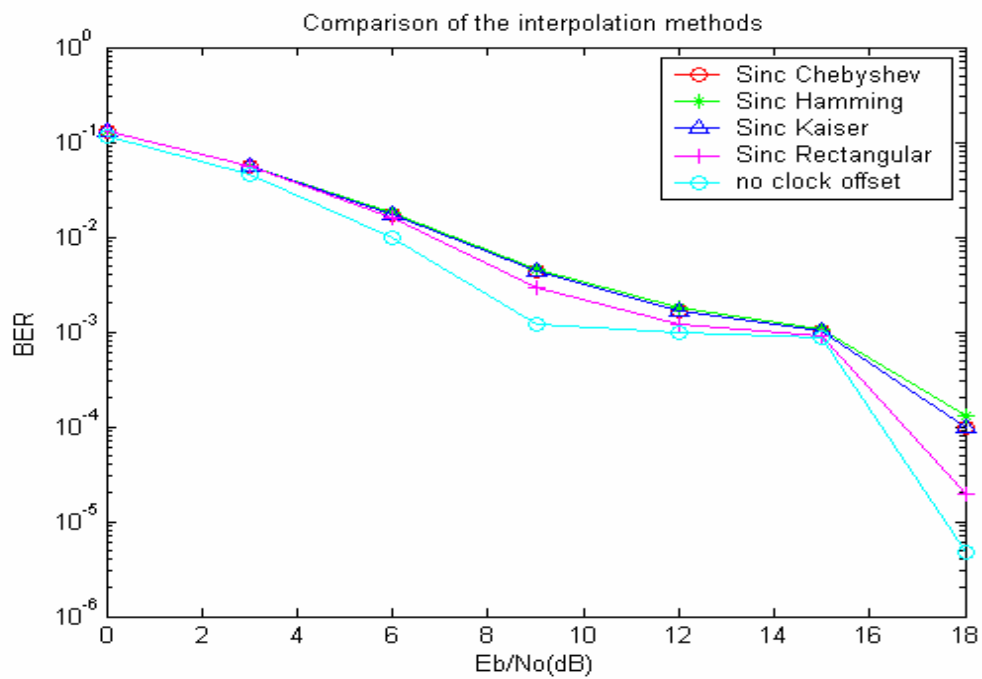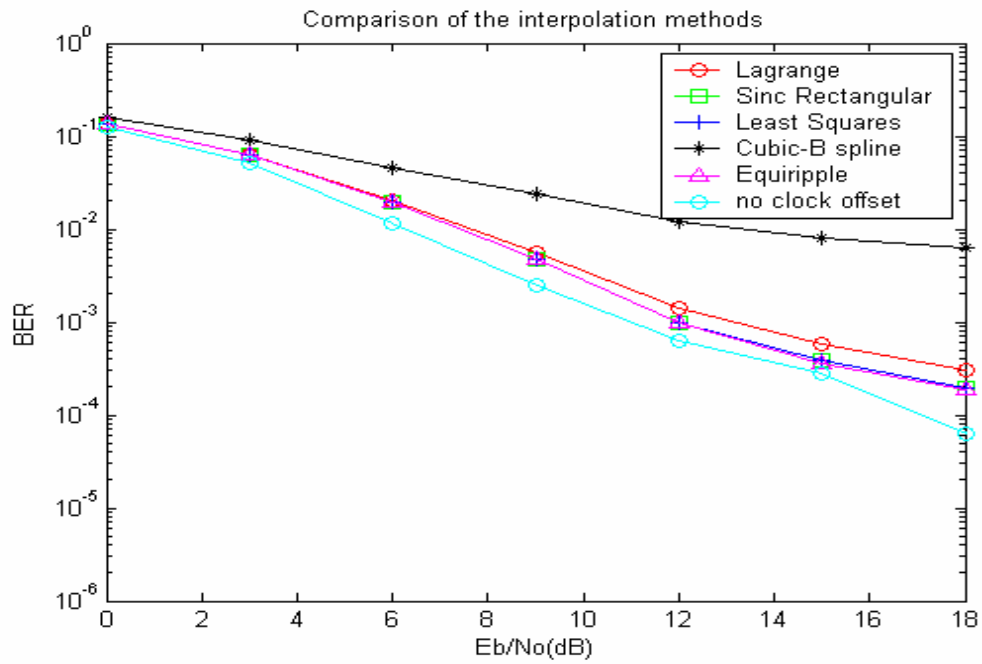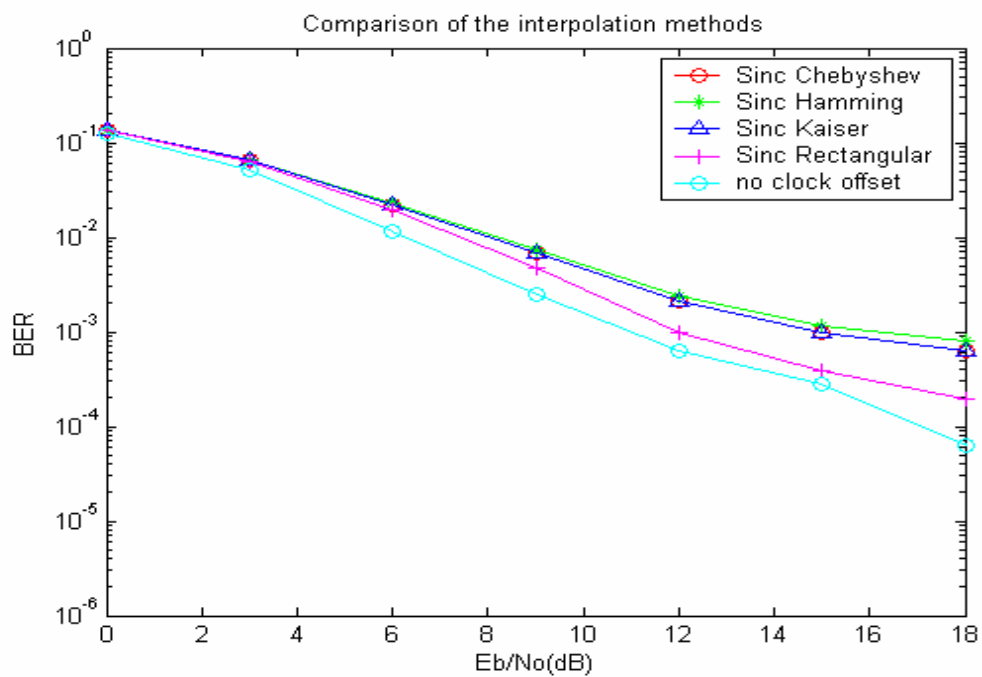(a) polynomial-based interpolators. (b) windowed sinc interpolators.

Chapter 6
Conclusion

In this thesis, we investigate all the required synchronization operations and related design techniques that jointly achieve frame, symbol, carrier, and sampling clock synchronizations for MIMO OFDM systems. Practical designs are applied to IEEE 802.11n receiver. Performances are measured under the multi-path fading channels. From simulation results, the proposed synchronization schemes work well in those conditions.

In the frame detection, we use the received signal power to set the threshold, and detect the beginning of a frame by comparing the auto-correlation outputs to the threshold [44]. In the symbol timing synchronization, we perform the coarse symbol timing estimation first with the double-sliding-window method [45], then adjust the

symbol timing by the proposed STDFS method to get the more accurate symbol timing estimation than the current techniques. Owing to the features of the IEEE 802.11n preambles which are composed of ten repeat short training symbols in the short training field, two repeat long training symbols, and the guard interval in the long training filed, we utilize the features to enhance the carrier frequency offset estimation by averaging the available auto-correlation outputs, and take advantage of the multi channel diversity of MIMO for better synchronization. In the sampling clock offset estimation, we reduce the computation complexity by only computing the most apart pilots, and take advantage of the multi channel diversity of MIMO for better synchronization. As for the sampling clock compensation, we find that Lagrange interpolation is suitable for IEEE 802.11n standard.

In the future, we will further reduce the computational complexity of the overall synchronization schemes. In the frame synchronization, we will try to reduce the computational complexity by only comparing the real part of the auto-correlation outputs, or by averaging the squares of the auto-correlation real part and the auto-correlation imaginary part. Furthermore, we will try to max-ratio combine the possible pilot pairs to obtain more accurate clock offset estimation, and max-ratio combine the estimated MIMO multi channel parameter samples for better parameter estimation. Up to now, the simulations are conducted by floating-point operations. We will also conduct fixed-point simulations so that we can reflect practical software and hardware realization and implementation designs.

References

[1] R. W. Chang, "Synthesis of band limited orthogonal signals for multichannel data transmission," Bell Syst. Tech. J., vol. 45, pp. 1775-1796, Dec. 1966.

[2] G. J. Foschini and M. J. Gans, "On limits of wireless communications in a fading environment when using multiple antennas," Wireless Pers. Commun., vol. 6, no. 3, pp. 311–335, Mar. 1998.

[3] WWiSE, "WWiSE proposal" http://www.wwise.org/

[4] TGnSync, "TGnSync proposal" http://www.tgnsync.org/home

[5] IEEE Std. 802.11, Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, 1999 edition.

[6] IEEE Std. 802.11a-1999, Wireless LAN Medium Access Control and Physical Layer specifications: High-speed physical layer in the 5 GHz band.

[7] R. V. Nee and R. Prasad, OFDM for Wireless Multimedia Communications. Boston: Artech House, 2000.

[8] H. Nogami, and T. Nagashima, "A frequency and timing period acquisition technique for OFDM System," Proc. IEEE Personal, Indoor, Mobile Radio Commun., pp. 1010-1015, 1995.

[9] T. Pollet, M. V. Bladel, and M. Moeneclaey, "BER sensitivity of OFDM systems to carrier frequency and weiner phase noise," IEEE Trans. Commun., vol. 43, pt. 1, pp. 191-193, Feb.- Apr. 1995.

[10] A. Palin and J. Rinne, "Symbol synchronization in OFDM system for time selective channel conditions," Proc. Electronics, Circuits and Systems, 1999, Proceedings of ICECS '99, The 6th IEEE International Conference on, vol. 3, pp. 1581 – 1584, Sept. 1999.

[11] B. Yang, K. B. Letaief, R. S. Cheng, and Z. Cao, "Timing recovery for OFDM

transmission," Selected Areas in Communications, IEEE Journal on, vol. 18, pp. 2278 - 2291, 2000.

[12] F. Classen and H. Meyr, "Frequency synchronization algorithms for OFDM systems suitable for communication over frequency selective fading channels," Vehicular Technology Conference, 1994 IEEE 44th, vol.3, pp. 1655 - 1659, June 1994.

[13] M. Speth, D. Daecke and H. Meyr, "Minimum overhead burst synchronization for OFDM based broadband transmission," Global Telecommunications Conference, 1998, GLOBECOM 98, The Bridge to Global Integration, IEEE, Volume: 5 , pp. 2777 – 2782, Nov. 1998.

[14] E. Kuusmik, "Wireless Lan integration into mobile phone," Master's Thesis, Department of Signals and Systems, Chalmers University of Technology, Finland, September 2004.

[15] ANSI/IEEE Standard 802.11b, 1999 Edition.

[16] ANSI/IEEE Standard 802.11g, 2003 Edition.

[17] Y. Asai, S. Kurosaki, T. Sugiyama, and M. Umehira, "Precise AFC scheme for performance improvement of SDM-COFDM," Proc. Vehicular Technology Conference, 2002. Proceedings. VTC 2002-Fall. 2002 IEEE 56th, vol. 3, pp. 1408 – 1412, Sept. 2002.

[18] T. C. W. Schenk and A. van Zelst, "Frequency synchronization for MIMO OFDM wireless LAN systems," Proc. Vehicular Technology Conference, 2003. VTC 2003-Fall, 2003 IEEE 58th, vol. 2, pp. 781 – 785, Oct. 2003.

[19] S. Johansson, M. Nilsson, and P. Nilsson, "An OFDM timing synchronization ASIC," Proc. Electronics, Circuits and Systems, 2000. ICECS 2000. The 7th IEEE International Conference on, vol.1, pp. 324 – 327, Dec. 2000.

[20] J. Liu and J. Li, "Parameter estimation and error reduction for OFDM-based WLANs," Mobile Computing, IEEE Transactions on, vol. 3, pp. 152-163, 2004.

[21] A. Fort, J.-W. Weijers, V. Derudder, W. Eberle, and A. Bourdoux, "A performance and complexity comparison of auto-correlation and cross-correlation for OFDM burst synchronization," Proc. Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP '03). 2003 IEEE International Conference on, vol. 2, pp. 341-344, April 2003.

[22] A. N. Mody and G. L. Stuber, "Receiver implementation for a MIMO OFDM system," Proc. Global Telecommunications Conference, 2002. GLOBECOM '02. IEEE, vol. 1, pp. 716 – 720, Nov. 2002.

[23] A. N. Mody and G. L. Stuber, "Synchronization for MIMO OFDM systems," Proc. Global Telecommunications Conference, 2001. GLOBECOM '01. IEEE, vol. 1, pp. 509 - 513, Nov. 2001.

[24] G. L. Stuber, J. R. Barry, S. W. McLaughlin, Y. Li, M. A. Ingram, and T. G. Pratt, "Broadband MIMO-OFDM wireless communications," Proceedings of the IEEE, vol. 92, pp. 271-294, 2004.

[25] K. Taura, M. Tsujishita, M. Takeda, H. Kato, M. Ishida, and Y. Ishida, "A digital audio broadcasting (DAB) receiver," Consumer Electronics, IEEE Transactions on, vol. 42, pp. 322-327, 1996.

[26] E. Zhou, X. Zhang, H. Zhao, and W. Wang, "Synchronization algorithms for MIMO OFDM systems," Proc. Wireless Communications and Networking Conference, 2005 IEEE, vol. 1, pp. 18 – 22, March 2005.

[27] N. P. Sands and K. S. Jacobsen, "Pilotless timing recovery for baseband multicarrier modulation," IEEE Journal on Selected Areas in Communications, Volume: 20, Issue: 5, pp. 1047 – 1054, June 2002.

[28] M. Speth, S. Fechtel, G. Fock, and H. Meyr, "Optimum receiver design for OFDM-based broadband transmission .II. A case study," Communications, IEEE Transactions on, vol. 49, pp. 571-578, 2001.

[29] T. Pollet, P. Spruyt and M. Moeneclaey, "The BER performance of OFDM systems using non-synchronized sampling," Proc. IEEE Global Telecommunications Conference, pp. 253-257, 1994.

[30] T. Pollet and M. Peeters, "Synchronization with DMT modulation," IEEE Communications Magazine, Volume: 37 Issue: 4, pp. 80 -86, April 1999

[31] F. M. Gardner, "Interpolation in digital modems-part I: fundamentals," IEEE Transactions on Communication, vol. 41, pp. 502-508, Mar 1993.

[32] L. Erup, F. M. Gardner, R. A. Harris, "Interpolation in digital modems-part II: implementation and performance," IEEE Transactions on Communications, vol. 41, pp. 502-508, Jun 1993.

[33] E. Martos-Naya, J. Lopez-Fernandez, L. D. del Rio, M.C. Aguayo-Torres and J. T. E. Munoz, "Optimized interpolator filters for timing error correction in DMT systems for xDSL applications," IEEE Journal on Selected Areas in Communications, Volume: 19, Issue: 12, pp. 2477 –2485, Dec. 2001.

[34] T. I. Laakso, V. Valimaki, Matti Karjalanen and U. K. Laine, "Splitting the unit delay," IEEE Signal Processing Magazine, pp. 30-60, Jan 1996.

[35] C. W. Farrow, "A continuously variable digital delay element," Proc. IEEE Int. Symp. Circuit and System, pp. 2641-2645, Jun 1988.

[36] N. J. Fliege, Multirate Digital Signal Processing: Multirate Systems, Filter banks, Wavelets, Chichester, John Wiley, 1994.

[37] X. Zhao, J. Kivinen, and P. Vainnikainen, "Tapped delay line channel models at 5.3 GHz in indoor environments," Proc. Vehicular Technology Conference, 2000.

IEEE VTS-Fall VTC 2000. 52nd, vol. 1, pp. 1 − 5, Sept. 2000.

[38] Y. Li, N. Seshadri, and S. Ariyavisitakul, "Channel estimation for OFDM systems with transmitter diversity in mobile wireless channels," Selected Areas in Communications, IEEE Journal on, vol. 17, pp. 461-471, 1999.

[39] Y. Li, "Simplified channel estimation for OFDM systems with multiple transmit antennas," Wireless Communications, IEEE Transactions on, vol. 1, pp. 67-75, 2002.

[40] S. M. Alamouti, "A simple transmit diversity technique for wireless communications," Selected Areas in Communications, IEEE Journal on, vol. 16, pp. 1451-1458, 1998.

[41] V. Tarokh, H. Jafarkhani, and A. R. Calderbank, "Space-time block coding for wireless communications: performance results," Selected Areas in Communications, IEEE Journal on, vol. 17, pp. 451-460, 1999.

[42] V. Valimaki and T. I. Laakso, "Principles of fractional delay filters," IEEE International Conference on Acoustics, Speech, and Signal Processing, 2000. ICASSP '00, Proceedings., vol. 6 , pp. 3870 − 3873, June 2000.

[43] M. H. Lai, "Clock synchronization of VDSL system," Master's Thesis, Department of Electronics Engineering & Institute of Electronics, National Chiao Tung University, Taiwan, May 2004.

[44] C. H. Cho, "Frame synchronization and digital automatic gain control for OFDM Wireless LAN System," Master's Thesis, Department of Electronics Engineering & Institute of Electronics, National Chiao Tung University, Taiwan, July 2002.

[45] J. Terry and J. Heiskala, OFDM Wireless LANs: a theoretical and practical guide, 2002.

自　　傳

劉佳旻，1980 年 11 月 29 日出生，台灣省台中市人。2003 年自國立交通大學電子工程學系獲得學士學位，隨即進入國立交通大學電子研究所攻讀碩士學位，研究興趣為訊號處理與數位通訊，碩士論文題目為「多重輸入輸出正交分頻多工系統之同步設計研究」。