CHANNEL ESTIMATION AND SYNCHRONIZATION FOR OFDM AND OFDMA SYSTEMS

WANG ZHONGJUN

NATIONAL UNIVERSITY OF SINGAPORE
2008
CHANNEL ESTIMATION AND SYNCHRONIZATION FOR
OFDM AND OFDMA SYSTEMS

WANG ZHONGJUN
(M. Eng., National University of Singapore)
(M. Sc., Shanghai Jiao Tong University)

A THESIS SUBMITTED
FOR THE DEGREE OF DOCTOR OF PHILOSOPHY
DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING
NATIONAL UNIVERSITY OF SINGAPORE
2008
To

Grace Wang Ruiqi, my dearest daughter
Acknowledgment

I would like to thank my supervisors Professor Yan Xin and Professor George Mathew for their constant guidance and encouragement throughout the period of this research work. Without their help and advice completion of the thesis would not have been possible.

I wish to thank Professor Xiaodong Wang, Columbia University, with whom I have had the good fortune to collaborate. I have benefited a lot from his inspirational guidance.

I also wish to thank my mentor Mr. Masayuki Tomisawa and my fellow colleagues in Wipro Techno Centre (Singapore), for encouraging me to carry out my research work. Their understanding and support were essential to the completion of my study.

Special thanks goes to my fellow graduate students Jinhua Jiang, Lan Zhang, Yan Wu and Feifei Gao who have always been willing to discuss and exchange ideas and help me a lot in my study. I owe them a great deal for their friendship.

Last but not least, I thank my parents and my wife for their love and support which played an instrumental role in the completion of this project.
# Contents

Acknowledgment i  
Contents ii  
Summary vi  
List of Tables viii  
List of Figures ix  
List of Abbreviations xiii  
List of Symbols and Operators xvi  

Chapter 1. Introduction 1  
1.1 Introduction to OFDM Based Systems 1  
1.2 Motivation for the Present Work 3  
1.3 Contributions of This Thesis 4  
1.3.1 Channel Estimation in OFDM Systems 4  
1.3.2 Phase Error Suppression for Multi-Band OFDM-UWB Systems 7  
1.3.3 CFO Estimation for SISO-OFDMA and MIMO-OFDMA Uplink 8  
1.4 Organization of the Thesis 10  

Chapter 2. ML Channel Estimation in OFDM Systems 12  
2.1 OFDM System Model 12
### Chapter 6. Phase Error Suppression for Multi-Band OFDM-UWB Systems 61

6.1 SFO Estimation and Compensation ............................. 61
   6.1.1 Basic Algorithm for SFO Estimation ......................... 62
   6.1.2 Weighted SFO Estimation ................................. 64
   6.1.3 Combined SFO Estimation ............................... 65
   6.1.4 Two-Dimensional SFO Compensation ..................... 68

6.2 CPE Estimation and Correction .................................. 69
   6.2.1 Weighted CPE Estimation ................................. 69
   6.2.2 Smoothed CPE Estimation ............................... 70
   6.2.3 Analysis of MSE Reduction Performance ................. 72
   6.2.4 CPE Correction ................................. 74

6.3 Numerical Results ............................................. 76

6.4 Concluding Remarks ........................................... 83

### Chapter 7. ML Estimation in OFDMA Systems 84

7.1 Signal Model for Generalized OFDMA Uplink ..................... 85

7.2 Existing ML Based CFO Estimators ............................. 87

7.3 Cramér–Rao Bound (CRB) ...................................... 90

7.4 Convergence Property of ML Estimation ....................... 92

7.5 Concluding Remarks ........................................... 92

### Chapter 8. New Approach for OFDMA Uplink CFO Estimation 94

8.1 Divide-and-Update Frequency Estimator (DUF E) .................. 94
   8.1.1 Step 1 – Primitive CFO Estimation ......................... 95
   8.1.2 Step 2 – Divide-and-Update CFO Adjustment .............. 96
   8.1.3 Computation of $\Phi(\omega^{(i+1)})$ ....................... 98

8.2 Further Discussion on DUF E .................................. 100
   8.2.1 Choices of $\mathcal{G}(\cdot)$ ............................... 100
   8.2.2 An Example Illustrating the Convergence Behavior of DUF E ... 101
   8.2.3 Remarks on Joint CFO and Channel Estimation ............ 103
Summary

The development of robust and high-performance channel estimation and synchronization algorithms plays an important role in the area of multicarrier/multiuser wireless communications. In this dissertation, we investigate some critical issues associated with the development of these algorithms for orthogonal frequency division multiplexing (OFDM) and OFDM multiple-access (OFDMA) systems.

This thesis consists of three parts. In the first part, the maximum likelihood (ML) solution for channel estimation in OFDM systems is investigated. The mean-squared error (MSE) performance of the conventional ML estimator (MLE) is analyzed and is shown to be linearly related to the effective length of channel impulse response (ELCIR). Tracking the variation in ELCIR is thus very important for conventional MLE for achieving optimum estimation. But, incorporating a run-time update of ELCIR into the ML estimator turns out to be computationally expensive. Therefore, a modified ML channel estimator, which systematically combines the ML estimation with a frequency-domain smoothing technique, is proposed. The proposed modification is presented in two forms, namely, optimum-smooth MLE (OMLE) and iterative-smooth MLE (IMLE). The proposed method introduces no extra complexity, and its performance has been proved using theoretical analysis and simulations to be robust to variation in ELCIR. Numerical results are provided to show the effectiveness of the proposed estimator under time-invariant and time-variant channel conditions.

In the second part of this thesis, we propose an efficient channel estimation and phase error suppression technique for multi-band OFDM based ultra-wideband (UWB) communications. The channel estimator is based on a simple least-square algorithm,
but enhanced with a novel channel frequency response (CFR) weighted decision-directed detection technique as well as a frequency-domain smoothing operation. The proposed phase error suppression scheme consists of a clock recovery loop and a common phase error (CPE) mitigation mechanism. The clock recovery loop performs estimation of sampling frequency offset (SFO) and its two-dimensional (time and frequency) compensation, while the CPE mitigation deals with phase errors caused by residual carrier-frequency offset (CFO) and SFO as well as phase noise. The SFO and CPE estimators use the pilot-tone and CFR based approaches, each of which employs a robust error reduction scheme and involves neither angle calculation nor division, and thus they are of low-complexity. Analytical and numerical results are provided to show that the proposed scheme is of high performance and robust even under highly noisy multipath channel conditions.

In the third part of this thesis, we devote our effort to ML approaches for joint estimation of CFO, timing error, and channel response of each active user in both single-input single-output (SISO) and multiple-input multiple-output (MIMO) OFDMA systems. In particular, we focus our investigation on ML CFO estimation for the OFDMA uplink with generalized carrier-assignment scheme (GCAS), which is believed to be the most challenging task in OFDMA applications. In this study, we propose a new approach, namely the divide-and-update frequency estimator (DUFEM). The proposed approach outperforms the existing solutions, the so-called alternating-projection frequency estimator (APFE), and its simplified form, the approximate APFE (AAPFE), in the sense that the DUFEM has the lowest computational complexity while maintaining the high estimation accuracy feature of the ML solution. We achieve this by decomposing the practically almost infeasible dense grid-search required in the APFE into an iterative approach with affordable complexity and by transforming the inverse of a large matrix into a series of matrix inversions of small dimensions using the Woodbury matrix identity. Performance and complexity comparisons are provided with comprehensive numerical simulations to show the effectiveness of the proposed method.
List of Tables

5.1 Summary of the proposed multi-stage channel estimation scheme. . . . . 48
5.2 Required computational complexity for CFR estimation per subband in a frame. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 56
7.1 Summary of the APFE scheme. . . . . . . . . . . . . . . . . . . . . . . 89
7.2 Summary of the AAPFE scheme. . . . . . . . . . . . . . . . . . . . . . 90
8.1 CRB’s dependence on the number of subcarriers per user (SNR = 22 dB). 105
8.2 Required computational load (excluding matrix inversions). . . . . . . 113
8.3 Comparison of the required computational load between DUFE and AAPFE. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 114
9.1 Description of the DUFE based CFO estimation for MIMO-OFDMA uplink. 120
9.2 Computational complexities of the DUFE based and the AAPFE based MIMO CFO estimators (excluding matrix inversions) for $N_q \times N_r$ MIMO OFDMA system. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 125
9.3 Ratio of the total computational complexities of the DUFE based and the AAPFE based $2 \times 2$ MIMO CFO estimators. . . . . . . . . . . . . . . . 126
List of Figures

2.1 NMSE performance for different values assumed for ELCIR. ............. 17

3.1 NMSE performance comparison of CMLE and OMLE. ..................... 22

3.2 Illustration of the proposed IMLE. ........................................... 25

3.3 NMSE performance comparison of CMLE and IMLE. ..................... 26

3.4 NMSE performance comparison for various channel estimation methods. 29

3.5 FER performance comparison for various channel estimation methods. ... 30

3.6 FER performance comparison when the channel is time-variant. ....... 31

4.1 (a) Illustration of the OFDM-UWB frame structure; (b) Example of TFC 
    for the $m$th multi-band OFDM symbol group with TFC $= 1$. ............ 34

5.1 NMSE ratio, $R_{\text{mse}}^{1,2}$, versus smoothing parameter $\alpha_h$ and $\text{SNR}_r$ under 
    various channel environments. (a) CM1, (b) CM2, (c) CM3, and (d) CM4. 46

5.2 NMSE performance comparison for channel estimates based on a single 
    header OFDM symbol obtained using different methods, under (a) CM1, 
    (b) CM2, (c) CM3, and (d) CM4 (Ana, Sim, CW, KH and SA are 
    abbreviations for analytical, simulation, CFR-weighted, known header 
    and simple average, respectively.). ........................................... 51

5.3 Analytical NMSE ratio, $R_{\text{mse}}$, under various channel environments with 
    $\alpha_h = 0.1$ and $\beta_h = 0.05$. ............................................ 54
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.4</td>
<td>NMSE performance comparison for various channel estimation methods under (a) CM1, (b) CM2, (c) CM3, and (d) CM4 (Sim, Ana, and PD are abbreviations for simulation, analytical and proposed, respectively.)</td>
<td>58</td>
</tr>
<tr>
<td>5.5</td>
<td>FER performance comparison for various channel estimation methods under (a) CM1 &amp; CM2, (b) CM3 &amp; CM4.</td>
<td>59</td>
</tr>
<tr>
<td>6.1</td>
<td>Block diagram of the proposed phase error suppression scheme.</td>
<td>62</td>
</tr>
<tr>
<td>6.2</td>
<td>An example of the SFO tracking using the proposed clock recovery loop.</td>
<td>69</td>
</tr>
<tr>
<td>6.3</td>
<td>Normalized deviation of CPE estimation, $\sigma_E$, when two types of smoothing filters are used at various values of $\epsilon$ with $\beta = 6$ KHz.</td>
<td>73</td>
</tr>
<tr>
<td>6.4</td>
<td>MSE-reduction ratio, $\eta_{\text{mse}}$, varies as a function of $\alpha_c$, $\epsilon$, and $\sigma_w^2$ with $\beta = 6$ KHz, when 1st-order low-pass filtering is used.</td>
<td>75</td>
</tr>
<tr>
<td>6.5</td>
<td>MSE-reduction ratio, $\eta_{\text{mse}}$, varies as a function of $\alpha_c$, $\epsilon$, and $\sigma_w^2$ with $\beta = 6$ KHz, when 2nd-order low-pass filtering is used.</td>
<td>75</td>
</tr>
<tr>
<td>6.6</td>
<td>MSE performance comparison for various SFO estimation methods under CM1 and CM2.</td>
<td>77</td>
</tr>
<tr>
<td>6.7</td>
<td>MSE performance comparison for various SFO estimation methods under CM3 and CM4.</td>
<td>78</td>
</tr>
<tr>
<td>6.8</td>
<td>CPE tracking performance comparison for 1st-order and 2nd-order inter-OFDM-symbol smoothing methods with known CFR under CM1.</td>
<td>79</td>
</tr>
<tr>
<td>6.9</td>
<td>CPE tracking performance comparison for 1st-order and 2nd-order inter-OFDM-symbol smoothing methods with estimated CFR under CM1.</td>
<td>79</td>
</tr>
<tr>
<td>6.10</td>
<td>FER performance comparison for various CPE tracking methods with $\epsilon = 0.01$ and $\beta = 6$ KHz under CM1 and CM3.</td>
<td>80</td>
</tr>
<tr>
<td>6.11</td>
<td>FER performance comparison for various CPE tracking methods with $\epsilon = 0.01$ and $\beta = 6$ KHz under CM2 and CM4.</td>
<td>80</td>
</tr>
<tr>
<td>6.12</td>
<td>FER performance of the overall system under various assumptions on phase error and channel condition.</td>
<td>82</td>
</tr>
<tr>
<td>6.13</td>
<td>FER performance comparison for SCO tracking using 21 and 6 pilot-tone pairs.</td>
<td>82</td>
</tr>
</tbody>
</table>
### List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.1</td>
<td>Discrete-time equivalent baseband model of the GCAS OFDMA uplink.</td>
<td>85</td>
</tr>
<tr>
<td>7.2</td>
<td>Illustration of convergence of the exact ML estimate of $\omega$.</td>
<td>93</td>
</tr>
<tr>
<td>8.1</td>
<td>Illustration of the user grouping in the $(i + 1)$th iteration. Users 1 and 2 with $\Delta^<em>(k)\hat{\delta}_\omega^{(i)} &lt; 0$, and Users 3 and 4 with $\Delta^</em>(k)\hat{\delta}_\omega^{(i)} \geq 0$.</td>
<td>98</td>
</tr>
<tr>
<td>8.2</td>
<td>Illustration of the two-step adjustment of CFO estimation in each iteration of the DUF algorithm.</td>
<td>102</td>
</tr>
<tr>
<td>8.3</td>
<td>Convergence performance of various CFO estimators ($N = 128$, $K = 4$, $\Delta f_{\text{max}} = 0.32$, and SNR = 22 dB).</td>
<td>106</td>
</tr>
<tr>
<td>8.4</td>
<td>Convergence behavior of various CFO estimators ($N = 256$, $K = 8$, $\Delta f_{\text{max}} = 0.32$, and SNR = 22 dB).</td>
<td>108</td>
</tr>
<tr>
<td>8.5</td>
<td>Convergence behavior of various CFO estimators ($N = 512$, $K = 8$, $\Delta f_{\text{max}} = 0.48$, and SNR = 22 dB).</td>
<td>108</td>
</tr>
<tr>
<td>8.6</td>
<td>Accuracy of various CFO estimators versus SNR in the presence of all users with equal power ($N = 128$, $K = 4$, and $\Delta f_{\text{max}} = 0.32$).</td>
<td>109</td>
</tr>
<tr>
<td>8.7</td>
<td>Accuracy of DUF versus SNR in the presence of all users with equal power ($N = 256$, $K = 8$, $\Delta f_{\text{max}} = 0.32$; and $N = 512$, $K = 8$, $\Delta f_{\text{max}} = 0.48$).</td>
<td>110</td>
</tr>
<tr>
<td>8.8</td>
<td>Accuracy of various CFO estimators versus SNR in the presence of near-far effect ($N = 128$, $K = 4$, and $\Delta f_{\text{max}} = 0.32$).</td>
<td>111</td>
</tr>
<tr>
<td>8.9</td>
<td>Accuracy comparison for the DUF estimator in the presence of all users with equal power and in the presence of near-far effect with two near users (high-SNR, $\lambda_1 = \lambda_2 = 4$) and two far users (low-SNR, $\lambda_3 = \lambda_4 = 1$). $N = 128$, $K = 4$, and $\Delta f_{\text{max}} = 0.32$.</td>
<td>111</td>
</tr>
<tr>
<td>8.10</td>
<td>The ratio in percentage of matrix-inversion complexity of DUF to that of AAPFE.</td>
<td>113</td>
</tr>
<tr>
<td>9.1</td>
<td>Convergence performance of DUF and APFE for $2 \times 1$ MISO and $2 \times 2$ MIMO CFO estimations ($N = 256$, $K = 4$, $\Delta f_{\text{max}} = 0.32$, and SNR = 22 dB).</td>
<td>122</td>
</tr>
</tbody>
</table>
List of Figures

9.2 Accuracy of the proposed DUFE versus SNR in the presence of all users with equal power in a MIMO OFDMA system. . . . . . . . . . . . . . . . 123

9.3 The ratio (in percentage) of matrix-inversion complexity of the DUFE based $2 \times 2$ MIMO CFO estimator to that of the AAPFE based $2 \times 2$ MIMO CFO estimator, obtained using (9.20). . . . . . . . . . . . . . . . 124
**List of Abbreviations**

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Full Form</th>
</tr>
</thead>
<tbody>
<tr>
<td>AAPFE</td>
<td>Approximate Alternating-Projection Frequency Estimator</td>
</tr>
<tr>
<td>ADC</td>
<td>Analog-to-Digital Converter</td>
</tr>
<tr>
<td>ANSFO</td>
<td>Accumulated Normalized Sampling Frequency Offset</td>
</tr>
<tr>
<td>AP</td>
<td>Alternating-Projection</td>
</tr>
<tr>
<td>APFE</td>
<td>Alternating-Projection Frequency Estimator</td>
</tr>
<tr>
<td>AWGN</td>
<td>Additive White Gaussian Noise</td>
</tr>
<tr>
<td>BER</td>
<td>Bit Error Rate</td>
</tr>
<tr>
<td>BPSK</td>
<td>Binary Phase Shift Keying</td>
</tr>
<tr>
<td>BS</td>
<td>Base Station</td>
</tr>
<tr>
<td>CAS</td>
<td>Carrier-Assignment Scheme</td>
</tr>
<tr>
<td>CFO</td>
<td>Carrier Frequency Offset</td>
</tr>
<tr>
<td>CFR</td>
<td>Channel Frequency Response</td>
</tr>
<tr>
<td>CIR</td>
<td>Channel Impulse Response</td>
</tr>
<tr>
<td>CICIR</td>
<td>Carrier to Inter–Carrier Interference Ratio</td>
</tr>
<tr>
<td>CMLE</td>
<td>Conventional Maximum-Likelihood Channel Estimator</td>
</tr>
<tr>
<td>CNR</td>
<td>Carrier-to-Noise Ratio</td>
</tr>
<tr>
<td>COFDM</td>
<td>Coded Orthogonal Frequency-Division Multiplexing</td>
</tr>
<tr>
<td>CP</td>
<td>Cyclic Prefix</td>
</tr>
<tr>
<td>CPE</td>
<td>Common Phase Error</td>
</tr>
<tr>
<td>CRB</td>
<td>Cramer–Rao Bound</td>
</tr>
<tr>
<td>DAB</td>
<td>Digital Audio Broadcasting</td>
</tr>
<tr>
<td>DAC</td>
<td>Digital-to-Analog Converter</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Description</td>
</tr>
<tr>
<td>-------------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td>dB</td>
<td>Decibels</td>
</tr>
<tr>
<td>dBc</td>
<td>Decibels Relative to Carrier</td>
</tr>
<tr>
<td>DC</td>
<td>Direct Current</td>
</tr>
<tr>
<td>DD</td>
<td>Decision-Directed</td>
</tr>
<tr>
<td>DFT</td>
<td>Discrete Fourier Transform</td>
</tr>
<tr>
<td>DUFE</td>
<td>Divide-and-Update Frequency Estimator</td>
</tr>
<tr>
<td>DVB</td>
<td>Digital Video Broadcasting</td>
</tr>
<tr>
<td>DSRC</td>
<td>Dedicated Short-Range Communications</td>
</tr>
<tr>
<td>ELCIR</td>
<td>Effective Length of Channel Impulse Response</td>
</tr>
<tr>
<td>FCC</td>
<td>Federal Communications Commission</td>
</tr>
<tr>
<td>FEC</td>
<td>Forward Error Correction</td>
</tr>
<tr>
<td>FER</td>
<td>Frame Error Rate</td>
</tr>
<tr>
<td>FFT</td>
<td>Fast Fourier Transform</td>
</tr>
<tr>
<td>FIM</td>
<td>Fisher Information Matrix</td>
</tr>
<tr>
<td>FIR</td>
<td>Finite Impulse Response</td>
</tr>
<tr>
<td>GCAS</td>
<td>Generalized Carrier-Assignment Scheme</td>
</tr>
<tr>
<td>HIPERLAN</td>
<td>High Performance Local Area Network</td>
</tr>
<tr>
<td>IBI</td>
<td>Inter–Block Interference</td>
</tr>
<tr>
<td>ICAS</td>
<td>Interleaved Carrier-Assignment Scheme</td>
</tr>
<tr>
<td>ICI</td>
<td>Inter–Carrier Interference</td>
</tr>
<tr>
<td>IDFT</td>
<td>Inverse Discrete Fourier Transform</td>
</tr>
<tr>
<td>IEEE</td>
<td>Institute of Electrical and Electronics Engineers</td>
</tr>
<tr>
<td>IMLE</td>
<td>Iterative-Smooth Maximum-Likelihood Channel Estimator</td>
</tr>
<tr>
<td>LMMSE</td>
<td>Linear Minimum Mean-Squared Error</td>
</tr>
<tr>
<td>ISI</td>
<td>Inter–Symbol Interference</td>
</tr>
<tr>
<td>LS</td>
<td>Least Square</td>
</tr>
<tr>
<td>MAI</td>
<td>Multiple-Access Interference</td>
</tr>
<tr>
<td>MAN</td>
<td>Metropolitan Area Network</td>
</tr>
<tr>
<td>MISO</td>
<td>Multiple–Input Single–Output</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-------------</td>
</tr>
<tr>
<td>MIMO</td>
<td>Multiple–Input Multiple–Output</td>
</tr>
<tr>
<td>ML</td>
<td>Maximum-Likelihood</td>
</tr>
<tr>
<td>MLE</td>
<td>Maximum-Likelihood Channel Estimator</td>
</tr>
<tr>
<td>MMSE</td>
<td>Minimum Mean-Squared Error</td>
</tr>
<tr>
<td>MRC</td>
<td>Maximum Ratio Combining</td>
</tr>
<tr>
<td>MSE</td>
<td>Mean-Squared Error</td>
</tr>
<tr>
<td>NMSE</td>
<td>Normalized Mean Square Error</td>
</tr>
<tr>
<td>OFDM</td>
<td>Orthogonal Frequency-Division Multiplexing</td>
</tr>
<tr>
<td>OFDMA</td>
<td>Orthogonal Frequency-Division Multiple-Access</td>
</tr>
<tr>
<td>OMLE</td>
<td>Optimum-Smooth Maximum-Likelihood Channel Estimator</td>
</tr>
<tr>
<td>PDF</td>
<td>Probability Density Function</td>
</tr>
<tr>
<td>PHN</td>
<td>Phase Noise</td>
</tr>
<tr>
<td>PSK</td>
<td>Phase Shift Keying</td>
</tr>
<tr>
<td>QAM</td>
<td>Quadrature Amplitude Modulation</td>
</tr>
<tr>
<td>QoS</td>
<td>Quality of Service</td>
</tr>
<tr>
<td>QPSK</td>
<td>Quadrature Phase Shift Keying</td>
</tr>
<tr>
<td>SFO</td>
<td>Sampling Frequency Offset</td>
</tr>
<tr>
<td>SISO</td>
<td>Single–Input Single–Output</td>
</tr>
<tr>
<td>SNR</td>
<td>Signal-to-Noise Ratio</td>
</tr>
<tr>
<td>SVD</td>
<td>Singular Value Decomposition</td>
</tr>
<tr>
<td>TFC</td>
<td>Time-Frequency Code</td>
</tr>
<tr>
<td>USB</td>
<td>Universal Serial Bus</td>
</tr>
<tr>
<td>UWB</td>
<td>Ultra Wide-Band</td>
</tr>
<tr>
<td>WLAN</td>
<td>Wireless Local Area Network</td>
</tr>
<tr>
<td>ZP</td>
<td>Zero Padding</td>
</tr>
<tr>
<td>1-D</td>
<td>One-Dimensional</td>
</tr>
<tr>
<td>2-D</td>
<td>Two-Dimensional</td>
</tr>
<tr>
<td>4G</td>
<td>Fourth Generation</td>
</tr>
</tbody>
</table>
List of Symbols and Operators

$a$ lowercase letters are used to denote scalars
$a$ boldface lowercase letters are used to denote column vectors
$A$ boldface uppercase letters are used to denote matrices
$\star$ convolution of two sequences
$(\cdot)^T$ transpose of a vector or a matrix
$(\cdot)^*$ conjugate only of a scalar or a vector or a matrix
$(\cdot)^H$ Hermitian transpose of a vector or a matrix
$(\cdot)^{-1}$ inversion of a matrix
$[\cdot]_i$ $i$th entry of a vector
$[\cdot]_{m,l}$ $(m, l)$th entry of a matrix
$| \cdot |$ absolute value of a scalar or the cardinality of a set
$| \cdot |_e$ modulo-$c$ operation
$\| \cdot \|$ Euclidean norm of a vector
$\text{Trace}(\cdot)$ trace of a matrix
$E\{\cdot\}$ statistical expectation operator
$\text{Var}\{\cdot\}$ statistical variance operator
$p\{\cdot\}$ probability density function of an event
$\text{Pr}\{\cdot\}$ probability of an event
$\mathcal{N}(a, b)$ Gaussian random variable with mean $a$ and variance $b$
$\text{sgn}(x)$ sign of $x$ which equals 1, if $x \geq 0$, and, $-1$, otherwise
$\Re\{}$ real part of the argument
$\Im\{}$ imaginary part of the argument
$\text{diag}\{d_1, \ldots, d_P\}$ diagonal matrix with diagonal entries $d_1, \ldots, d_P$
<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$I_P$</td>
<td>$P \times P$ identity matrix</td>
</tr>
<tr>
<td>$i_P$</td>
<td>$P \times 1$ vector whose entries are all ones</td>
</tr>
<tr>
<td>$0_{P\times Q}$</td>
<td>$P \times Q$ matrix whose elements are all zeros</td>
</tr>
<tr>
<td>$0_P$</td>
<td>$P \times P$ all-zero matrix</td>
</tr>
<tr>
<td>$\mathbb{Z}^P_{P_2}$</td>
<td>finite integer set ${P_1, P_1 + 1, \ldots, P_2}$</td>
</tr>
<tr>
<td>$F_N$</td>
<td>$N$-point DFT matrix with the $(m, l)$th entry given by $[F_N]_{m,l} = (1/\sqrt{N}) \exp(-j2\pi ml/N)$, $0 \leq m, l \leq N - 1$</td>
</tr>
</tbody>
</table>
Chapter 1

Introduction

Orthogonal frequency division multiplexing (OFDM) plays an important role in a variety of modern communication systems. The objective of this thesis is to undertake an in-depth investigation of issues and solutions in the development of OFDM based wireless communication systems. In this chapter, the motivation of the present work and the contributions of this thesis are highlighted after a preliminary introduction of OFDM based systems. Finally, an overview of the text in this thesis is presented.

1.1 Introduction to OFDM Based Systems

OFDM is an effective technology to support high speed transmission over wireless channels with a relatively low complexity, and therefore has been widely used in many existing and developing standards such as digital audio broadcasting (DAB) [5], digital video broadcasting (DVB) [6, 7], high performance local area network (HIPERLAN) [8], IEEE 802.11a wireless local area network (WLAN) [9], IEEE 802.16a metropolitan area network (MAN) [10], and etc. Recently, with the allocation of unlicensed radio spectrum from 3.1 GHz to 10.6 GHz for ultra wideband use by the US Federal Communications Commission (FCC), the multi-band OFDM based ultra-wideband (UWB) systems have been proposed for achieving very high-rate wireless data transmission [11–14]. OFDM is also being pursued for dedicated short-range communications (DSRC) for road-side to
vehicle communications and as a potential candidate for fourth-generation (4G) mobile wireless systems [15–19].

OFDM converts a frequency-selective channel into a set of frequency flat subchannels. Even though the subcarriers associated with different subchannels in OFDM have the minimum frequency separation required to maintain orthogonality among their corresponding time-domain waveforms, the signal spectra corresponding to the different subcarriers overlap in frequency. Thus, the available bandwidth is used very efficiently in OFDM. It is a block modulation scheme where a block of $N$ information symbols is transmitted in parallel on $N$ subcarriers. The time duration of an OFDM symbol is $N$ times larger than that of a single-carrier system. An OFDM modulator can be implemented as an inverse discrete Fourier transform (IDFT) on a block of $N$ information symbols followed by a digital-to-analog converter (DAC). To mitigate the effects of inter-symbol interference (ISI) caused by channel time spread, each block of $N$ IDFT coefficients is typically preceded by a cyclic prefix (CP) or a guard interval consisting of $N_g$ samples, such that the length of the CP is at least equal to the channel length. Under this condition, linear convolution of the transmitted sequence and channel is the same as circular convolution. As a result, the effects of ISI can be easily and completely eliminated. Moreover, the approach enables the receiver to use fast signal processing transforms such as fast Fourier transform (FFT) for OFDM implementation. Because of these properties, OFDM systems are more advantageous over single-carrier systems and become desirable for many applications [16, 20].

A well-known application example in context of the OFDM technology is the aforementioned multi-band (MB) OFDM-based UWB communication, which has attracted considerable attention in the recent past [21–27]. The large bandwidth occupancy of UWB and high efficiency in spectrum utilization provided by OFDM make it possible for the OFDM-UWB technology to achieve very high channel capacity. In practice, this technology has been adopted to support high-speed short-range wireless connectivity among devices, e.g., certified wireless universal serial bus (USB) that aims to offer data rates up to 480 Mbps within three meters is based on the MB-OFDM UWB.
1.2 Motivation for the Present Work

Recently, there have also been another two important OFDM based technology developments in the area of wireless communications, namely the multiple-input multiple-output (MIMO) OFDM system, and the orthogonal frequency-division multiple-access (OFDMA) system. OFDM is combined with antenna arrays at the transmitter and receiver to increase the diversity gain and/or to enhance the system capacity over time-variant and frequency-selective channels, resulting in a MIMO configuration [16, 28–33]. In OFDMA systems, several users simultaneously transmit their data by modulating an exclusive set of orthogonal subcarriers. As advanced extensions to the traditional OFDM systems, both technologies can provide higher data throughput, higher bandwidth efficiency and more flexibility for network deployment, which make them good candidates for use in future broadband wireless communications [10, 34–39].

One of the critical design issues in OFDM systems is to achieve accurate and robust channel estimation under various hostile conditions, especially in the presence of time-varying distortions, such that the receiver can use the channel information to recover the transmitted signals with a trivial equalization process\(^1\) [41–48]. In addition, one of the major drawbacks of the OFDM scheme and its two extensions (i.e., MIMO-OFDM and OFDMA) is that they are sensitive to time misalignments, sampling frequency offsets (SFO’s) and carrier-frequency offsets (CFO’s). These offsets result in ISI, inter-carrier interference (ICI), and/or multiple-access interference (MAI), thereby limiting the performance [49–52]. These issues have brought in numerous design challenges that have become active research areas recently [53–68]. In this thesis, we focus on three such design issues. We first investigate channel estimation in OFDM systems including multi-band OFDM-based UWB systems. Secondly, we investigate phase error mitigation

\(^1\)In an OFDM system, equalization is usually performed using a one-tap frequency-domain equalizer with low complexity.
1.3 Contributions of This Thesis in multi-band OFDM-UWB systems. Thirdly, we study CFO estimation in single-input single-output (SISO) OFDMA and MIMO-OFDMA systems.

1.3 Contributions of This Thesis

In the following three sub-sections, we summarize the contributions of this thesis in the three areas highlighted above. In each sub-section, we first present a review of background, state of the art approaches in the literature, and highlights of the deficiencies of these approaches. This is followed by a summary of how the present work in this thesis successfully addresses these deficiencies.

1.3.1 Channel Estimation in OFDM Systems

OFDM systems transform high-rate data signals, which would otherwise suffer from severe frequency selective channel fading, into a number of orthogonal components before transmission, with the bandwidth of each component being less than the coherence bandwidth of the channel. By modulating them onto different subcarriers, each component experiences only frequency flat fading. As a result, together with a forward error correction (FEC) channel coding scheme, a simple one-tap equalizer can be used to combat the fading at each subcarrier. Further, in coded OFDM systems\(^2\), coherent detection is preferred for providing the channel decoder with proper constellation knowledge. This requires channel estimation and tracking, and it is usually done in frequency-domain, i.e., by estimating the channel frequency response (CFR)\(^3\).

Channel estimators developed for OFDM can be classified into two main categories: pilot assisted estimation [69–72] and blind or semi-blind channel estimation [73–83]. In pilot assisted approaches, pilot signals are embedded in certain subcarriers of each OFDM symbol. At the receiver, the channel components estimated using these pilots are interpolated for estimating the complete channel. These pilots can also be used to

\(^2\)OFDM systems with FEC coding are usually called coded OFDM (COFDM) systems in the literature.

\(^3\)Channel estimation for OFDM is seldom performed in time-domain due to the multi-carrier nature of OFDM systems.
1.3 Contributions of This Thesis

track channel variations. The blind schemes avoid the use of pilots, for achieving high spectral efficiency. This is achieved at the cost of higher implementation complexity and some amount of performance loss. The performance loss can be recovered to some extent by resorting to semi-blind approaches, which use a few pilots to eliminate the phase ambiguity problem that exists in blind approaches and to provide initial channel estimation. The pilot density in semi-blind approaches is much sparse compared to pilot assisted methods, thereby maintaining the feature of spectral efficiency.

In this thesis, we consider a semi-blind approach for channel estimation. In this approach, the pilot signals in each OFDM symbol are mainly used for phase offset tracking and correction. Channel estimation is done based on either block-type pilots available in the system, when the channel can be treated as invariant over a certain period of time, or virtual block-type pilots obtained using the well-known decision-directed (DD) coherent detection for time-varying channels [84, 85]. In both cases, either least square (LS) or minimum mean-squared error (MMSE) based algorithms can be adopted for CFR estimation. While LS is the simplest, it has the drawback of low noise reduction capability [86]. MMSE offers very good performance, but suffers from high complexity as well as strong dependence on channel statistics and signal-to-noise ratio (SNR).

Several modified MMSE or LS estimators for OFDM applications have been developed in [86–92], among which [87] provides a good overview. Edfors et al. [88] introduced a singular value decomposition (SVD) based frequency-domain linear MMSE (LMMSE) estimator using a low-rank approximation approach. However, it still requires knowledge of channel frequency correlation and SNR. In their earlier work in [89], a modified LS estimator which does not require knowledge of the channel frequency correlation was introduced. It is a low complexity approach with performance comparable to that of MMSE. A similar idea was explored in [90], and developed as a low complexity maximum-likelihood channel estimator (MLE). However, our analysis shows that the existing MLE requires knowledge of the effective length of channel impulse response (ELCIR), which includes delay spread and SNR as well as timing errors, for delivering good performance. Incorporating into MLE the adaptivity required to acquire and use this
knowledge results in significant computational complexity. Furthermore, as we show in Chapter 2, the estimation error in delay spread and/or SNR contributes significantly to the system performance degradation.

In this thesis, we propose a novel modification to the MLE to achieve robustness against estimation errors in delay spread, SNR, and symbol timing while retaining the computational simplicity of the MLE. The modified ML channel estimator systematically combines ML estimation with a frequency-domain smoothing technique. The proposed modification is presented in two forms, a so-called optimum-smooth MLE (OMLE) and iterative-smooth MLE (IMLE). The proposed method introduces no extra complexity, and its performance has been proved using theoretical analysis and simulations to be robust to variation in ELCIR.

A special scenario in the domain of channel estimation for OFDM systems is related to the OFDM-UWB system. The OFDM-based UWB system, as specified by the Wimedia Alliance [12], uses frame-based transmission. Typically, the UWB channel can be treated as invariant over the transmission period of one OFDM frame. The estimation of CFR thus can be accomplished using the channel estimation sequences included in the frame preamble. In this sense, many existing schemes, including LS, ML, or MMSE based algorithms, can be adopted for CFR estimation. As the OFDM-UWB is expected to deliver reliable service even under very low SNR conditions (less than 0 dB) [24], simply applying the LS algorithm to the channel estimation sequences may not yield a CFR estimate with acceptable accuracy. Again, in this case, both ML and MMSE offer high estimation accuracy, but suffer from high computational complexity as well as strong dependence on channel statistics and SNR. The ML estimator introduced in [90] (including our modified versions), for example, either requires to pre-store a large matrix in memory or performs matrix inversion in realtime. This requirement, of course, is prohibitive in low-power and low-cost wireless UWB devices. In [93], a scheme, which is termed a time-domain least-squares channel estimator, is proposed for achieving low complexity channel estimation in OFDM-UWB applications. However, carefully examining the actual computational complexity of this method shows that it also
1.3 Contributions of This Thesis

suffers from similar drawbacks of the ML solution discussed above and thus it may not be suitable for practical implementation of a high speed and low cost UWB device. In this thesis, we present a novel channel estimation scheme which is tailored for OFDM-UWB applications. The proposed estimator is LS based, but enhanced with a frequency-domain smoothing operation as well as a simple yet effective decision-directed (DD) coherent detection process. The proposed scheme outperforms existing solutions [90, 92, 93] in the sense that it achieves estimation accuracy comparable to that of the ML solution while maintaining low computational complexity in an order similar to that of conventional LS solutions.

1.3.2 Phase Error Suppression for Multi-Band OFDM-UWB Systems

To practically realize MB-OFDM UWB, one needs to cope with numerous design challenges, particularly in receiver designs such as symbol timing, CFO and SFO compensation, as well as CFR estimation. In addition to the aforementioned development of an efficient channel estimation algorithm that is critically important to system performance and can be efficiently implemented in practice, in this thesis, we also address two important synchronization related design issues.

We first consider the SFO caused by sampling clock frequency mismatch between transmitter and receiver. Since the analog-to-digital converter (ADC) in a UWB device operates at high sampling rates (at least 528 MHz), even a small SFO could result in phase-shift in the received data at all carriers. The accumulated phase-shift over a certain period tends to be significant and will degrade the system performance considerably if it is not well tracked and compensated [59, 94]. To remove the effect of SFO in OFDM systems, various schemes have been proposed [52, 64, 65, 95, 96], and most of them aim to deal with SFO tracking for applications with relatively low processing speed [e.g., the IEEE 802.11a WLAN] and require calculation of actual angles, division operations, and/or even matrix manipulation. Thus, they are not suitable for OFDM-UWB systems where low-cost, low-power and high-speed implementation is crucial. For example, due
1.3 Contributions of This Thesis

to high complexity and high power consumption, the maximum-likelihood (ML) phase tracking approach is known to be prohibitive in this case [95–99].

In this thesis, we present a novel sampling clock recovery technique. We propose a low-complexity and robust phase-shift estimation scheme which is interference resilient even under very low SNR conditions. In particular, we develop a simple SFO compensation scheme in place of the conventional time-domain interpolation which is implementation expensive in high-speed systems.

Secondly, we consider the common phase error (CPE) caused by residual CFO (i.e., after initial estimation and compensation of CFO), residual SFO and random Wiener phase noise (PHN) [50, 100]. Many schemes have been proposed to estimate the CPE’s in various OFDM systems, e.g., see [101–104] and the references therein. Due to the random characteristics of PHN, CPE estimation is usually based on each individual OFDM symbol, i.e., the CPE’s present in different OFDM symbols are estimated separately. In [103] and [104], the approaches using multiple OFDM symbols are also proposed for achieving improved CPE estimation. However, since these methods require manipulation of phases which requires high complexity, they may not be suitable for OFDM-UWB applications. We develop a simple CFR weighted CPE estimator and introduce an effective inter-OFDM-symbol smoothing scheme for enhancing the CPE tracking performance.

1.3.3 CFO Estimation for SISO-OFDMA and MIMO-OFDMA Uplink

Emerging as a promising technology for next generation wireless communication systems, the OFDMA has received considerable amount of research interest [10, 34–40]. An appealing feature of OFDMA is its capability to mitigate the effects of MAI. However, this can be achieved only if timing and frequency synchronization are in order. In particular, the CFO’s of an OFDMA system, if not properly estimated and compensated, result in ICI and MAI at the receiver [16, 100, 105]. In the downlink of an OFDMA system, the signals for different users are multiplexed by the same transmitter, and
1.3 Contributions of This Thesis

each user performs frequency synchronization via estimating and correcting the CFO between its own receiver and the base station (BS) transmitter. In such a scenario, the CFO estimation is relatively simple because only a single CFO needs to be handled. Many existing CFO estimation algorithms for OFDM systems are also applicable to this case [106–109, 111–115]. However, in the uplink of an OFDMA system, different users have possibly different CFO’s, all of which should be accurately estimated at the BS receiver. Estimating these CFO’s amounts to solving a complex multi-parameter estimation problem and is considered to be as a primary challenge in OFDMA receiver design.

Another appealing feature of OFDMA is its capability to optimally allocate system resources such as transmission power and spectrum, via dynamic subcarrier assignment. Roughly speaking, there are three major carrier-assignment schemes (CASs), namely, the subband-based CAS [34, 35], the interleaved CAS (ICAS) [36], and the generalized CAS (GCAS) [37, 39]. In the subband-based CAS, a number of continuous subcarriers are assigned to each user. In the interleaved CAS, the subcarriers allocated to each user are equi-spaced in the whole frequency band. The GCAS allows each user to select the subcarriers according to its quality of service (QoS) and channel conditions. Hence, compared to subband-based CAS and interleaved CAS, the GCAS offers more flexibility in subcarrier assignment for each user and is advantageous in optimizing system resource allocation [10, 116]. However, estimation and compensation of multiuser CFO’s and timing errors are more challenging in GCAS than that in subband-based CAS and interleaved CAS.

Under the assumption that each user transmits a training block at the beginning of an uplink frame, the CFO estimates are obtained using a ML approach in [39]. To reduce computational complexity of the ML approach, an alternating-projection (AP) algorithm is employed to replace a multi-dimensional search with a sequence of one-dimensional (1-D) searches [117]. This algorithm leads to a so-called AP frequency estimator (APFE), but still with fairly high computational complexity. Its simplified form, the approximate APFE (AAPFE), was also proposed in [39]. Compared with the APFE, the AAPFE has
lower complexity but suffers considerable performance degradation. Another suboptimal approach, which has lower computational complexity than the APFE, is described in [118]. The scheme achieves complexity reduction by approximating the inverse of a CFO-dependent matrix with the inverse of a predetermined matrix. Moreover, the scheme requires that the system has large number of subcarriers and satisfies certain conditions on the selection of training signals. In practice, such requirements may undermine the flexibility provided by GCAS.

In this thesis, we investigate the CFO estimation problem in the GCAS based OFDMA uplink. We develop a method, a so-called divide-and-update frequency estimator (DUFE), to overcome the aforementioned drawbacks of APFE and AAPFE. To achieve complexity reduction, we replace the computationally costly grid-search in APFE by a computationally efficient iterative algorithm, and transform the inversion of a large matrix into a series of inversions of much smaller matrices. Compared to APFE and AAPFE, the proposed scheme has lower computational complexity while maintaining high estimation accuracy similar to that of the exact ML solution.

Moreover, we also extend the use of the proposed DUFE scheme to GCAS based MIMO-OFDMA uplink. The inclusion of MIMO processing makes the problem even more complicated. In this thesis, we propose to decompose the MIMO-OFDMA CFO estimation into a series of multiple-input single-output (MISO) ML estimations, each of which adopts a DUFE based iterative approach. Compared with other existing approaches, the proposed scheme strikes much better performance-complexity tradeoffs.

### 1.4 Organization of the Thesis

The rest of this thesis, which consists of three parts, is organized as follows.

The first part of the thesis, comprising Chapters 2 and 3, is devoted to the study of ML channel estimation. In Chapter 2, the concept of ML channel estimation for OFDM systems is reviewed and the issues involved in existing solutions are investigated. In Chapter 3, a modified ML channel estimator is proposed. Its performance analysis as well
as the numerical results illustrating its performance are also presented in this chapter.

The second part of the thesis focuses on the study of OFDM-UWB systems. This part includes Chapters 4, 5 and 6. In Chapter 4, the signal model for the OFDM-UWB system is established. The signal model broadly covers transmission, propagation channel and receiving interference including PHN, CFO and SFO. Based on this signal model, in Chapter 5, we propose a novel channel estimator and analyze its performance and implementation complexity. In Chapter 6, going one step further, we propose and analyze an innovative phase error suppression technique which is simple yet highly effective for SFO estimation and compensation as well as CPE tracking and correction.

The third part of the thesis consisting of Chapters 7 to 9, deals with CFO estimation for SISO-OFDMA and MIMO–OFDMA uplink transmission. Chapter 7 presents the signal model of GCAS based OFDMA uplink transmission followed by a review and analysis of the existing APFE and AAPFE schemes. In Chapter 8, a new scheme, the DUFE, that overcomes the drawbacks of the existing solutions is proposed. Convergence behavior of the DUFE is discussed and its effectiveness is demonstrated through simulation and complexity comparison. In Chapter 9, we extend the use of the DUFE to deal with the more difficult problem of CFO estimation for MIMO–OFDMA uplink transmission.

Finally, Chapter 10 concludes the thesis with short remarks on the main message of each chapter. This chapter also presents some directions for future work in the problems addressed in the thesis. Further, detailed derivations of some of the analytical results used in the main chapters are provided in Appendices.
Chapter 2

ML Channel Estimation in OFDM Systems

The maximum likelihood (ML) channel estimation has proved to be very useful in OFDM systems from both analytical and practical perspectives. As described in Chapter 1, ML channel estimators used in OFDM systems are known to be of low complexity, yet with performance comparable to that of MMSE estimators. In this chapter, after a brief description of the signal model that will be used in our study, we give a review of the existing ML channel estimator (MLE). We present both analytical and simulation results to show that the mean-squared error (MSE) of MLE is linearly related to the effective length of channel impulse response (ELCIR). This is surely a drawback of the conventional MLE, which motivated us to develop a modified MLE as the remedy as will be seen in Chapter 3.

2.1 OFDM System Model

We consider a typical wireless spectral shaping OFDM system employing $N$ subcarriers for the transmission of $P$ ($P < N$) parallel data symbols. The DC subcarrier and $N - P - 1$ subcarriers (virtual carriers) at the edges of the spectrum are not used for simplifying the receiver design. The stream of data $\{c_i\}$ (from phase shift keying (PSK) or quadrature
2.1 OFDM System Model

amplitude modulation (QAM) constellations) is partitioned into adjacent blocks of length $P$. After insertion of $N - P$ zeros, the $m^{th}$ OFDM block

$$
\mathbf{s}^{(m)} := [s^{(m)}(0), s^{(m)}(1), s^{(m)}(2), \ldots, s^{(m)}(N - 1)] \\
= [0, c_1^{(m)}, \ldots, c_Q^{(m)}, 0, \ldots, 0, c_{-Q}^{(m)}, \ldots, c_{-1}^{(m)}]
$$

with $Q = P/2$, is fed to a $N$-point inverse discrete Fourier transform (IDFT) that generates a $N$-dimensional vector of time-domain samples. To maintain orthogonality among subcarriers and to eliminate inter-symbol interference (ISI) resulting from time dispersive channels, a $N_g$-point cyclic prefix (CP) is appended as guard interval to each time-domain vector, thus forming an OFDM symbol.

Prior to transmission, the time-domain OFDM symbols are usually formed into frames, with each frame consisting of a number of consecutive OFDM symbols. The time dispersive channel is assumed to be invariant over at least the duration of one OFDM symbol. The channel is modelled as a $N_h$-tap finite impulse response (FIR) filter whose impulse response during transmission of the $m^{th}$ OFDM symbol is given as

$$
\mathbf{h}^{(m)} = [h^{(m)}(0), h^{(m)}(1), \ldots, h^{(m)}(N_h - 1)]^T. \tag{2.1}
$$

The DFT of $\mathbf{h}^{(m)}$ is given by\(^1\)

$$
g^{(m)}(n) = \sum_{k=0}^{N_h-1} h^{(m)}(k) e^{-j2\pi nk/N}, \quad n \in \mathbb{Z}_{N-1}^Q. \tag{2.2}
$$

At the receiver end, after the $N_g$-point CP of each OFDM symbol is removed, the received samples are passed to a $N$-point discrete Fourier transform (DFT) processor. With the assumption of $N_h \leq N_g$ and dropping the OFDM symbol index, $m$, the useful output of the DFT processor becomes

$$
x(n) = s(n)g(n) + v(n), \quad n \in \mathbb{Z}_1^Q \cup \mathbb{Z}_{N-1}^{N-Q} \tag{2.3}
$$

\(^1\)Note that notations $x(t)$ and $x[l]$ ($x[l] = x(lT)$ at sampling interval $T$) are commonly used to denote continuous-time and discrete-time signals, respectively. In this thesis, by a slight abuse of notation convention, we also use $x(l)$ ($x(l) = x(lT)$) to denote discrete-time signals.
where $s(n)$ represents the afore-mentioned transmitted data symbol, and $v(n)$ is the channel additive noise, which is modelled in frequency domain as a zero-mean white complex Gaussian process with variance $\sigma^2 = \mathbb{E}[|v(n)|^2]$.

Let $S := \text{diag}\{s(1), s(2), \ldots, s(Q), s(N-Q), s(N-Q+1), \ldots, s(N-1)\}$ be a $P \times P$ diagonal matrix, $x := [x(1), x(2), \ldots, x(Q), x(N-Q), x(N-Q+1), \ldots, x(N-1)]^T$ be a $P \times 1$ vector, $g := [g(1), g(2), \ldots, g(Q), g(N-Q), g(N-Q+1), \ldots, g(N-1)]^T$ be a $P \times 1$ vector, and $v$ be a $P \times 1$ vector that is complex Gaussian distributed with mean zero and covariance matrix $C_v = \sigma^2 I_P$. Then, (2.3) can be rewritten as

$$x = Sg + v.$$ (2.4)

Let $M$ be an integer in the range $N_h \leq M \leq N_g$. Define a $M \times 1$ vector, $h_M$, whose first $N_h$ elements are same as the $h$ defined in (2.1) and the rest are all zeros, i.e.,

$$h_M = [h^T, 0, 0, \ldots, 0]^T$$ (2.5)

and, a $P \times M$ matrix, $D_M$, with entries

$$[D_M]_{n,k} = \begin{cases} 
  e^{-j2\pi k(n+1)/N}, & n \in \mathbb{Z}_Q^{Q-1} \\
  e^{-j2\pi k(n-2Q)/N}, & n \in \mathbb{Z}_Q^{P-1}
\end{cases}$$ (2.6)

for $k \in \mathbb{Z}_0^{M-1}$. By combining (2.2), (2.5) and (2.6), we can rewrite (2.4) as

$$x = SD_M h_M + v.$$ (2.7)

## 2.2 ML Channel Estimator and Performance

From (2.7), it can be seen that

$$(D_M^H D_M)^{-1} D_M^H S^{-1} x = h_M + (D_M^H D_M)^{-1} D_M^H S^{-1} v.$$ (2.8)

The second term on the right-hand side (RHS) of (2.8) is Gaussian distributed with zero mean. The conventional MLE of channel impulse response (CIR) is thus obtained as

$$\hat{h}_M = (D_M^H D_M)^{-1} D_M^H S^{-1} x.$$ (2.9)
2.2 ML Channel Estimator and Performance

Obviously, the DC subcarrier and \( N - P - 1 \) virtual subcarriers are not involved in obtaining the MLE of CIR\(^2\). The corresponding MLE of the channel frequency response (CFR) is obtained as [90]

\[
\hat{g}_M = D_M (D_M^H D_M)^{-1} D_M^H S^{-1} x. \tag{2.10}
\]

Let \( W_M \) be a \( P \times P \) matrix defined as

\[
W_M = D_M (D_M^H D_M)^{-1} D_M^H. \tag{2.11}
\]

Then, from (2.10), the MLE of CFR can be expressed as

\[
\hat{g}_M = W_M S^{-1} x. \tag{2.12}
\]

Clearly, the MLE given by (2.12) can be interpreted as a two-stage processing. The first stage, \( S^{-1} x \), performs the well-known least-square (LS) estimation of the CFR. The second stage, \( W_M \), first converts the LS estimate into the time-domain, then performs a linear transformation on the resulting CIR, and finally converts it back to the frequency-domain [90]. Since an indoor wireless multipath channel typically experiences a finite delay spread which is far less than \( N \) in practice, the use of the linear transformation, \( (D_M^H D_M)^{-1} \), forces the purely noisy tail portion (from \( M \) to \( N \)) of the estimated CIR to be zero. Using this, the residual error in the initial LS estimate can be further reduced in time-domain as long as \( M \) is selected to satisfy \( N_h \leq M \leq N_g \).

2.2.1 MSE of the ML Estimator

It is clear now that the accuracy of MLE depends on the selection of \( M \). From (2.8), (2.11) and (2.12), we have

\[
\hat{g}_M = W_M S^{-1} x = g + W_M S^{-1} v. \tag{2.13}
\]

\(^2\)One may envisage that the virtual subcarriers can also be used for obtaining enhanced channel estimation. However, the improved performance is achieved at the expense of significantly increased difficulty in the receiver implementation. Thus, we opt not to use the virtual subcarriers for channel estimation in this thesis.
The covariance matrix of $\hat{g}_M$ becomes
\[
C_g = E\{(\hat{g}_M - g)(\hat{g}_M - g)^H\} = E\{W_M S^{-1} vv^H (S^{-1})^H W_M^H\}.
\]
Since it can be assumed that $E\{S^{-1}(S^{-1})^H\} = I_P / P_{\text{avg}}$ for PSK or QAM modulated signal with average power $P_{\text{avg}}$, and $W_M$ is Hermitian and idempotent, i.e., $W_M = W_M^H$ and $(W_M)^k = W_M$ with $k$ being a finite positive integer, we have
\[
C_g = (\sigma^2 W_M W_M^H) / P_{\text{avg}} = (\sigma^2 / P_{\text{avg}}) W_M = W_M / \text{SNR}.
\]
Thus, MSE of the existing MLE can be obtained as
\[
\text{MSE}_1(M) = \text{Trace}(C_g) = M / \text{SNR}.
\] (2.14)
That is, MSE of the conventional MLE is linearly related to $M$.

### 2.2.2 Experimental Results

We shall now illustrate the above $M$-dependent property of conventional MLE using an experimental example. In the experiment, the MLE is applied to an IEEE 802.11a WLAN baseband system with $N = 64$, $P = 52$, $Q = 26$, and $N_g = 16$, and the initial estimate is achieved by applying LS on the two long training sequences [9].

In the simulations, we consider an indoor multipath fading channel, which is usually modelled by a tapped delay line with exponentially decaying weights on the taps and independent Rayleigh distributed fades on each tap. Mathematically, the exponential channel model is given as
\[
h(k) \sim \mathcal{N}(0, \sigma_k^2) + j\mathcal{N}(0, \sigma_k^2)
\]
where $\sigma_k^2 = \sigma_0^2 e^{-k T_S / T_{DS}}$ with $\sigma_0^2 = 1 - e^{-T_S / T_{DS}}$, $\sigma_0^2$ is chosen to ensure the same average received power for each channel realization, $T_S$ is the sampling interval and equals 50 ns in this case (with sampling rate = 20 MHz), and $T_{DS}$ is the delay spread and was set to a typical value of 50 ns in our simulations.
The channel estimation performance is evaluated in terms of normalized MSE (NMSE) defined by

\[
NMSE = \frac{E\{\sum_{n \in Z_1^Q \cup Z_{N-Q}^{N-1}} |g(n) - \hat{g}_M(n)|^2\}}{E\{\sum_{n \in Z_1^Q \cup Z_{N-Q}^{N-1}} |g(n)|^2\}}.
\]

Here, \(g(n)\) and \(\hat{g}_M(n)\) denote the actual and estimated channels, respectively.

Fig. 2.1 shows the NMSE behavior as \(M\) is varied from 1 to \(N_g\) (\(N_g = 16\) in WLAN). Note that \(M\) can be interpreted as the assumed ELCIR. Obviously, for each individual SNR, there exists an optimum value for \(M\). When \(M\) is selected to be equal to the ELCIR, the NMSE is minimized. For example, the optimum \(M\) is 5 when SNR = 5 dB. Observe that the estimation performance degrades as \(M\) deviates from its optimum value, and the degradation becomes significant for large deviations in \(M\). For example, if \(M\) is taken as 16 when SNR = 5 dB, the resulting NMSE is similar to that obtained by optimally setting \(M = 4\) in the case of SNR = 0 dB, thereby implying a SNR degradation of 5 dB.

![Fig. 2.1: NMSE performance for different values assumed for ELCIR.](image)

From the above analysis and discussion, we find that the ELCIR, which should be used as the optimum value of \(M\) in channel estimation, is related to both delay spread...
and SNR\textsuperscript{3}. Thus, ideally, \( M \) should be chosen adaptively such that it is always close to its optimum value. This may be done by incorporating certain means to dynamically detect the channel variations. However, in practice, this may not be desirable or even feasible for the following reason. The linear transformation, \( (D_M^H D_M)^{-1} \), in (2.11) is a \( M \times M \) matrix with its entries dependent on \( M \). Any change in \( M \) will require a real-time recalculation of a matrix inversion. This is undesirable and should be avoided in practice, if possible.

### 2.3 Concluding Remarks

In this chapter, we have reviewed the existing MLE for OFDM applications. Our analytical and numerical results show that the MSE of MLE is linearly related to the ELCIR. Thus, the existing MLE requires knowledge of the ELCIR, which includes delay spread and SNR as well as timing errors, for delivering good performance. Incorporating into MLE the adaptivity required to acquire and use this knowledge results in substantial computational complexity. In addition, the estimation error in delay spread and/or SNR contributes significantly to the system performance degradation. These drawbacks can be resolved by modifying conventional MLE as will be introduced in Chapter 3.

\textsuperscript{3}In case of non-perfect timing synchronization, the timing error contributes to ELCIR in a way similar to delay spread.
Chapter 3

Modified ML Channel Estimators

The discussion in Chapter 2 has shown that the application of existing ML channel estimator (MLE) in OFDM systems requires knowledge of the effective length of channel impulse response (ELCIR) for achieving optimum performance. In fact, in addition to channel environment variation, the ELCIR (i.e., $M$) may also be affected by symbol timing errors as the result of a non-perfect timing synchronization process. Thus, tracking the variation in ELCIR is very important for conventional MLE. But, incorporating a real-time update of ELCIR into the ML estimator turns out to be computationally expensive as we have seen from the discussion in Chapter 2.

In this chapter, we propose a modified ML channel estimator, which combines ML estimation with a frequency-domain smoothing technique. The modified estimator is presented in two forms. We call the first one optimum-smooth MLE (OMLE) and the second one iterative-smooth MLE (IMLE). For notational convenience, in the following, we call the conventional MLE as CMLE. Both OMLE and IMLE introduce no extra complexity when compared with CMLE, and their performances have been proved using both theoretical analysis and Monte-Carlo simulations to be robust to variation in ELCIR. Numerical results are provided to show the effectiveness of the proposed estimators under time-invariant and time-variant channel conditions.
3.1 Modified ML Channel Estimator I - OMLE

Generally speaking, one may assume that channel coherent bandwidth is larger than subcarrier spacing in OFDM systems. This property motivated us to smooth the channel frequency response (CFR) estimate on each subcarrier using the estimates from adjacent subcarriers such that the residual errors resulting from CMLE can be reduced. In particular, in this section, we assume that the CFR on one subcarrier is very close to those of its two adjacent subcarriers.

3.1.1 Smoothing Matrix for OMLE

We begin with the development of our first modified MLE - OMLE. We proceed by incorporating a frequency-domain smoothing in the CMLE. For this, we define a $P \times P$ matrix, $G_k$, with entries

$$[G_k]_{i,j} = \begin{cases} 
\alpha_i(k), & (i \in \mathbb{Z}^P_0) \cap (i = j); \\
1-\alpha_i(k), & [i, j] \in \{(0, 1], [Q-1, Q], [Q, Q+2], [P-1, P-2]\}; \\
\frac{1-\alpha_i(k)}{2}, & (i \in \mathbb{Z}^P_0) \cap (i \notin \{0, Q-1, Q, P\}) \cap (i = j \pm 1); \\
0, & \text{otherwise.}
\end{cases} \quad (3.1)$$

Here, $k \in \mathbb{Z}^K_1$ and $K$ is a design parameter as will be seen in the following. The OMLE of CFR is then given by

$$\hat{g}_G = W_G S^{-1}x = \prod_{k=1}^{K} (G_k W_M) S^{-1}x. \quad (3.2)$$

Careful examination of (3.1) shows that the role of $G_k$ in $W_G$ is to smooth the frequency-domain channel estimate on each subcarrier based on the estimates from its adjacent subcarriers. Thus, the proposed OMLE method can be illustratively interpreted as a process with the following steps:

**Step 1**: Perform least-square (LS) estimation by $S^{-1}x$.

**Step 2**: Compute the ML estimate based on the LS estimate obtained in Step 1.

**Step 3**: Apply the first smoothing matrix $G_1$ to the ML estimate obtained in Step 2.
Step 4: Refine the estimate obtained in Step 3 by again applying MLE followed by the second smoothing matrix \( G_2 \). Repeat this \( K \) times.

Denote by \( B_c \) the channel coherence bandwidth and \( B_s \) the subcarrier spacing. Usually, in a well-defined OFDM system, we have \( B_s \ll B_c/2 \). This leads to \( G_1g \approx g \).

Under these circumstances, one may find that introduction of the above simple smoother is undoubtedly effective in reducing the channel estimation error. For \( K = 1 \), in view of (2.4), we can rewrite (3.2) as

\[
\hat{g}_G = W_G S^{-1} x = G_1g + G_1 W_M S^{-1} v \approx g + G_1 W_M S^{-1} v.
\]  

(3.3)

Thus, the modified estimator can be treated as approximately unbiased. Following similar lines as in Section 2.2.1, the mean-squared error (MSE) of the modified MLE can be obtained as

\[
\text{MSE}_2(M) = \text{Trace}(W_G W_H^T_G)/\text{SNR}.
\]  

(3.4)

By properly selecting the parameters \( \alpha_i(k) \)'s and \( K \) in (3.1) and (3.2), the value of \( \text{Trace}(W_G W_H^T_G) \) can be reduced to be far less than \( M \). This is illustrated in Fig. 3.1 using the same example as described in Section 2.2.2. As shown in Fig. 3.1, the normalized MSE (NMSE) of the OMLE method remains almost flat as \( M \) increases from its optimum values. Thus, by systematically combining the MLE (time-domain) with smoothing (frequency-domain), as described above, we can potentially achieve a performance that is less sensitive to the choice of \( M \), as long as \( M \) is selected to be not less than the ELCIR.

The number of iterations, \( K \), in (3.2) can be generally selected to be a small value. The ideal value of \( K \) is dependent on the targeted system and application. In fact, while maintaining \( B_s \ll B_c/2K \), one may choose to use a larger \( K \) for reducing the estimation error further. In the IEEE 802.11a WLAN system, for example, our experiments showed that \( K = 2 \) is appropriate.

### 3.1.2 Derivation of Optimum \( \alpha_i(k) \)

We shall now discuss on the choice of the smoothing parameters \( \alpha_i(k) \)'s. In general, \( \alpha_i(k) \)'s can be selected to be small positive scalars which are less than 1. For different
k, $\alpha_i(k)$ can be selected to be either same or different. In fact, when the frequency selectivity of the channel is very low, it is possible to obtain unbiased channel estimation with optimum values of $\alpha_i(k)$’s, in the sense that $\text{MSE}_2(M)$ in (3.4) is minimized.

From (3.4), we know that minimizing $\text{MSE}_2(M)$ with respect to $\alpha_i(k)$ is equivalent to minimizing $J = \text{Trace}(W_GW^H_G)$. Let $K = 1$. Denote $\alpha_i(k)$ with $\alpha_i$ and $G_k$ with $G$ for notational convenience. Noting that $W_M$ is Hermitian and idempotent and assuming that all $\alpha_i$’s are real and with $U = G^T$, we have

$$J = \text{Trace}(G_W G^H_M G^T) = \text{Trace}(G_W G^T)$$

$$= \text{Trace}(U^T W_M U) = \sum_{r=1}^{P} u_r^T W_M u_r$$

where $u_r$ is the $r$th column of $U$. Next, setting to zero the gradients of $J$ with respect to $\alpha_i$’s, we obtain

$$\frac{\partial J}{\partial \alpha_i} = \sum_{r=1}^{P} (R u_r)^T \frac{\partial u_r}{\partial \alpha_i} = u_r^T R \frac{\partial u_r}{\partial \alpha_i} = 0$$

(3.5)
for $i = 0$ to $P - 1$, where $\mathbf{R} = \mathbf{W}_M + \mathbf{W}_M^*$. The second equality in (3.5) holds true because $\mathbf{u}_i$ is only related to $\alpha_i$'s that $\frac{\partial \mathbf{u}_i}{\partial \alpha_i} = 0$, if $r \neq i$. We further define $\Phi_i$ as a $3 \times 3$ matrix whose entries are copied from the $(i - 1)$th to $(i + 1)$th rows and columns of $\mathbf{R}$, $\Lambda = (-0.5, 1, 0.5)^T$, and let $\Psi_i = (\psi_{1i}, \psi_{2i}, \psi_{3i})^T = \Phi_i^T \Lambda$. From (3.1) and (3.5), it is straightforward to obtain closed-form expressions for $\alpha_i(k)$ as

$$
\alpha_i^{opt} = \begin{cases} 
\frac{|\mathbf{R}_{i+1,i+1} - |\mathbf{R}_{i,i+1}|}{|\mathbf{R}_{i,i+1}| + |\mathbf{R}_{i+1,i+1} - 2|\mathbf{R}_{i,i+1}|}, & i \in \{0, Q\} \\
\frac{|\mathbf{R}_{i-1,i-1} - |\mathbf{R}_{i,i-1}|}{|\mathbf{R}_{i,i-1}| + |\mathbf{R}_{i-1,i-1} - 2|\mathbf{R}_{i,i-1}|}, & i \in \{Q - 1, P - 1\} \\
\psi_{1i} + \psi_{3i}, & \text{otherwise}.
\end{cases}
$$

The simulations given in this chapter for OMLE are based on the optimum values of $\alpha_i(k)$ computed using (3.6).

It should be noted that, although the derivation of the optimum values of $\alpha_i(k)$ in (3.6) is based on the assumption $K = 1$, similar derivations can be applied for larger values of $K$ provided that the assumption $B_s \ll B_c / 2K$ maintains valid.

### 3.2 Modified ML Channel Estimator II - IMLE

In the last section, we have mentioned that the optimum $\alpha_i(k)$'s are achieved based on the assumption that the frequency selectivity of the channel is very low, i.e., $B_s \ll B_c / 2$. This condition may be too strict for some wireless applications. In this section, we introduce another modification to CMLE so that the frequency selectivity condition can be relaxed to some extent, e.g., $B_s < B_c / 2$. This method is based on an iterative smoothing process without using optimum $\alpha_i(k)$'s.

#### 3.2.1 Smoothing Matrix for IMLE

Similar to the development of OMLE, we obtain IMLE by incorporating a frequency-domain smoothing in CMLE. For this, we define a $P \times P$ matrix, $\mathbf{G}_{k,p}$, with
\[ [G_{k,p}]_{i,j} = \begin{cases} 
\alpha_1(k), & (i \in \mathbb{Z}_0^{P-1}) \cap (i \notin \{0, Q-1, Q, P-1\}) \cap (i = j = p); \\
\alpha_2(k), & (i = p) \cap ([i, j] \in \{[0, 0], [Q-1, Q-1], [Q, Q], [P-1, P-1]\}); \\
1, & (i \in \mathbb{Z}_0^{P-1}) \cap (i \neq p); \\
\frac{1 - \alpha_1(k)}{2}, & (i \in \mathbb{Z}_0^{P-1}) \cap (i \notin \{0, Q-1, Q, P-1\}) \cap (i = p) \cap (i = j \pm 1); \\
1 - \alpha_2(k), & (i = p) \cap ([i, j] \in \{[0, 1], [Q-1, Q-2], [Q, Q+1], [P-1, P-2]\}); \\
0, & \text{otherwise} 
\end{cases} \]  

(3.7)

where \( k \in \mathbb{Z}_1^K \), \( p \) is the subcarrier index with \( p \in \mathbb{Z}_0^{P-1} \), and \( \alpha_1(k) \) and \( \alpha_2(k) \) are small positive scalars which are less than 1. Here, \( K \) is the number of iterations involved in the iterative estimation as will be seen in the following. The proposed IMLE of CFR is then given by

\[ \hat{g}_I = W_I S^{-1} x = W_M \left[ \prod_{k=1}^{K} \prod_{p=0}^{P-1} (G_{k,p} W_M) \right] S^{-1} x. \]  

(3.8)

By examining (3.7), one may find that, again, the role of \( G_{k,p} \) is to smooth the frequency-domain channel estimation on \( p^\text{th} \) subcarrier by using the estimations from its adjacent subcarriers. For ease of understanding and with reference to Fig. 3.2, the proposed IMLE given in (3.8) can be illustratively decomposed into the following steps:

**Step 1**: Perform LS estimation by \( S^{-1} x \).

**Step 2**: Compute the ML estimate based on the LS estimate obtained in Step 1.

**Step 3**: In the \( k^{\text{th}} \) iteration, perform the following:

1. Apply the smoothing Matrix \( G_{k,0} \) to smooth the \( 0^\text{th} \) subcarrier;
2. Compute the ML estimate;
3. Repeat 1) and 2) \( P \) times, each time with matrix \( G_{k,p} \) for smoothing the \( p^\text{th} \) subcarrier \( (p = 0 \text{ to } P - 1) \).

**Step 4**: Refine the estimate obtained in Step 3 by repeating this \( K \) times.
3.2 Modified ML Channel Estimator II - IMLE

Clearly, in IMLE, each subcarrier is individually smoothed in the frequency-domain for reducing the estimation error caused by channel noise. Depending on the channel condition variation, it is possible that the smoothing itself may introduce some new estimation errors. In this case, the subsequent application of MLE can help to suppress such errors to some extent in time-domain. Clearly, the proposed estimator iteratively reduces the estimation error by a joint effort in time-domain and frequency-domain.

Again, following similar lines as in Section 2.2.1, the MSE of the IMLE can be obtained as

$$\text{MSE}_3(M) = \text{Trace}(W^*W^H)/\text{SNR}. \quad (3.9)$$

Numerical results show that, with a proper selection of parameters $\alpha_i(k)$’s and $K$ in (3.7) and (3.8), $\text{Trace}(W^*W^H)$ can be reduced to be far less than $M$. This has also been observed in an experiment similar to what we have done for OMLE, as shown in Fig. 3.3.
3.2 Modified ML Channel Estimator II - IMLE

Figure 3.3: NMSE performance comparison of CMLE and IMLE with $K = 5$. IMLE is far less sensitive to variation of $M$ than CMLE.

3.2.2 Parameter Selection in IMLE

The number of iterations, $K$, in (3.8) can be generally selected to be a small value. The ideal value of $K$ is dependent on the targeted system and application. While maintaining $B_s < B_c/2$, if the system needs to cope with the propagation channels with relatively lower frequency-selectivity, $K$ is expected to be larger, and vice versa. In the WLAN system, for example, our experiments showed that $K = 5$ is sufficiently large. The construction of $G_{k,p}$’s only depends on the selected values for $\alpha_1(k)$ and $\alpha_2(k)$. Although it is found that the final performance is not very sensitive to the selection of these values as long as they are small positive values less than 1, we may set a guideline for selection based on the following considerations. First, $\alpha_2(k)$ should be chosen slightly greater than $\alpha_1(k)$ since only a single neighbouring subcarrier is available for smoothing the estimation on the outer-most subcarrier. Second, $\alpha_i(k + 1)$ should be chosen slightly less than $\alpha_i(k)$ as the smoothers at higher iterations are supposed to have less noisy input. Following this guideline, in the simulations presented in this chapter, which are related to
IMLE, we have set $\alpha_1(k) = 0.5 - 0.05k$, and $\alpha_2(k) = \alpha_1(k) + 0.05$ for $k = 1$ to $K$.

### 3.3 Advantages of Modified Estimators

The fact that the performance of OMLE and IMLE is robust to the choice of $M$ becomes quite advantageous in actual implementation. Without any effort required to track the changes in the optimum value of $M$ caused by channel variations and/or timing estimation errors, we can now simply and safely pre-select $M$ to be $N_g$ when using the OMLE or IMLE method. Thereafter, $\alpha_i(k)$’s and $K$ can be properly selected, following the procedures presented above. In this way, the $P \times P$ matrix, $W_G$ in (3.2) or $W_I$ in (3.8), can be pre-computed and stored for real-time use. As a result, no extra implementation complexity or computational complexity is introduced by the smoothing matrices $G_k$’s or $G_{k,p}$’s as well as the extra MLE’s involved in the implementation of the modified MLEs given by (3.2) and (3.8). More importantly, the annoying real-time matrix inversion becomes unnecessary.

It should be noted that, with reference to (2.10), the CMLE can be efficiently implemented by the cascade of two $N$-point DFT operations ($D_H^M$ and $D_M$), weighted by a $M \times M$ matrix $D_C = (D_H^M D_M)^{-1}$ [90]. A similar arrangement can be made for OMLE and IMLE. Let $D_I = (D_H^M D_M)^{-1} D_H^M \prod_{k=1}^{K} \prod_{p=0}^{P-1} (G_{k,p} W_M) D_M (D_H^M D_M)^{-1}$. Since $W_M$ is Hermitian and idempotent, we can rewrite (3.8) as

$$\hat{g}_I = D_M D_I D_H^M S^{-1} x.$$  \hspace{1cm} (3.10)

Hence, the IMLE can also be efficiently implemented by the cascade of two $N$-point DFT operations ($D_H^M$ and $D_M$), weighted by a $M \times M$ matrix $D_I$. Again, the weighting matrix $D_I$ in the case of IMLE can be pre-computed by setting $M = N_g$ whereas the weighting matrix $D_C$ in the case of CMLE varies with different values of $M$. Clearly, our modified MLE is more efficient than the CMLE in practice.

\[1\] In the case of OMLE, by applying one more MLE to $\hat{g}_G$ in (3.2), one may obtain a similar implementation.
An additional advantage of the proposed IMLE scheme which should be emphasized here is its robustness to channel variations with less rigid frequency selectivity requirement when compared with the OMLE. It can be seen that the IMLE requires only an upper-bound of ELCIR for setting $M$ together with the assumption that the subcarrier spacing is less than half the channel coherence bandwidth, i.e., $B_s < B_c/2$. These conditions can be easily satisfied in the design of an indoor wireless OFDM system.

### 3.4 System Simulation Results

In the simulations, we consider the WLAN system described in Section 2.2.2. Two simulation scenarios are considered. In the first experiment, the channel is assumed to be time-invariant over a frame’s duration (with 1000 bytes payload). In this case, the channel is estimated using the two long training sequences at the beginning of each frame and is used by all subsequent OFDM symbols within that frame. Fig. 3.4 shows the NMSE performance versus SNR for different channel estimation schemes. As expected, both OMLE and IMLE perform close to the CMLE with optimum $^2 M$, but much better than the CMLE with $M = N_g = 16$. For comparison, Fig. 3.4 also shows the performances of the simplest LS method and the most sophisticated MMSE method. The frame error rate (FER) performance is shown in Fig. 3.5. It can be seen that, again, both OMLE and IMLE achieve similar FER performance as the CMLE method with optimum $M$, but they outperform the CMLE with $M = N_g = 16$. It is important to note that both modified MLEs are able to perform comparable to MMSE and much superior to LS.

In the second experiment, we assume that the channel is time-varying over a frame’s duration. In this case, instead of following the conventional way to model the time-varying nature of the channel by introducing the Doppler effect, we turn to an alternative way for simplifying the simulation effort. We continue to use the previous static channel model, in which channel realization remains unchanged over one frame and keeps varying (randomly) over different frames. However, over the entire period of

---

$^2$With reference to Fig. 2.1, “optimum $M$” refers to an integer which, among all possible integer values of $M$, gives lowest possible NMSE for a given SNR.

28
receiving a frame, the channel estimation keeps updated for each received OFDM symbol. The update of channel estimation proceeds as follows: the current OFDM symbol is equalized using the CFR estimated on the previous OFDM symbol; the decision-directed (DD) detection is then applied and an initial CFR estimate using LS is obtained; the MLE method (conventional or modified) is then used to improve the estimated CFR. The channel estimate resulting from the current OFDM symbol will be used for the next OFDM symbol, and so on. Clearly, by doing so, we equivalently obtain channel realization that is static over the transmission period of two consecutive OFDM symbols and variant over the transmission period of one OFDM frame.

The simulation results (for time-varying channel) are shown in Fig. 3.6. The improvement in FER performance by using the modified MLEs in combination with the DD detection is obvious. It shows an improvement of about 2.5 dB at a FER of 10% (the prescribed performance checking point for WLAN) with the OMLE and IMLE ($K = 5$), when compared to the CMLE with $M = N_g = 16$. It is interesting to note that the proposed methods outperform the training sequences based LS algorithm.
3.4 System Simulation Results

Figure 3.5: FER performance comparison for various channel estimation methods. The channel is assumed to be static over a frame’s duration with data rate of 6Mbps, 1000 bytes payload, and delay spread of 50 ns.

wherein the latter requires the channel to be constant. Also, interestingly, about 0.8 dB improvement has been achieved when the number of iterations in IMLE increases from 1 to 5. Observe that the IMLE with $K = 5$ has similar FER performance as the OMLE with a smaller value of $K$, i.e., $K = 2$. This is because $B_s \ll B_c/2K = B_c/4$ is satisfied in indoor WLAN applications that the smoothing parameters in OMLE have been set to their optimum values.

It should also be pointed out that, in practice, by smoothing/filtering the channel estimates on several neighbouring OFDM symbols, further performance improvement is possible for the proposed methods, especially when the channel is slowly fading. This means that the gap between the performance curves of the proposed methods (DD+OMLE; DD+IMLE, $K=5$) and that with perfect channel knowledge can be further reduced. The DD algorithm itself is well known to be easily prone to divergence at low SNR due to the poor performance of symbol-by-symbol detection [84]. On the other hand, we see that the proposed methods have provided a very simple and yet robust solution to
mitigate this problem. The inclusion of the smoothing matrices $G_k$’s or $G_{k,p}$’s in the MLE algorithm helps not only to suppress the additive channel noise but also to reduce the DD detection errors. This, of course, is desirable for achieving reliable blind channel estimation under time-varying channel conditions.

### 3.5 Concluding Remarks

In this chapter, we have presented two modified MLE methods with improved performance for OFDM systems. The performance enhancement arises from two aspects. First, by introducing frequency-domain smoothing in MLE, the proposed methods perform similar to a conventional MLE with perfect knowledge of the effective length of the channel. The fact that the proposed methods can be blind to variation in the effective length of the channel makes their implementation very simple and robust. Second, the simple and yet systematic combination of frequency-domain smoothing and MLE has proved to be very effective for reducing both channel noise and decision-directed detection errors. This makes the proposed methods promising candidates for performing
robust, low-complexity and high-performance channel estimation under time-varying channel environments.
Chapter 4

Multi-Band OFDM-UWB System Model

As described in Chapter 1, wide-band and high-speed processing raise a lot of challenges to OFDM-UWB system design. Examples of the design challenges are development of robust and yet high-performance channel estimation algorithms as well as phase error mitigation mechanisms. To address these issues effectively, a well-defined system model becomes essential. In this chapter, we present the signal model for the multi-band (MB) OFDM-UWB system. In a broad sense, this model covers the transmitter model, the propagation channel model as well as the modeling of phase noise, carrier frequency offset and sampling frequency offset.

4.1 Transmitter Model

As shown in Fig. 4.1(a), each OFDM-UWB frame is composed of a preamble, a header, and a payload. As specified in [12], the preamble consists of 30 OFDM symbols, among which the last six symbols are dedicated for channel estimation. The header consists of 12 OFDM symbols that convey information about the frame configuration. The payload consists of $M_s$ OFDM data symbols where $M_s$ is an integer multiple of 6, i.e., $M_s = 6M_0$. For the convenience of discussion, in the sequel, we index the OFDM symbol with $n$,
4.1 Transmitter Model

\[ n \in \mathbb{Z}_{0}^{M_{s}+17}, \] and \( n = 0 \) indicates the first channel estimation OFDM symbol. We also divide the OFDM symbols into groups, each of which consists of six consecutive OFDM symbols and is indexed with \( m, m \in \mathbb{Z}_{0}^{M_{0}+2}, \) and \( m = 0 \) indicates the group of six OFDM symbols for channel estimation.

<table>
<thead>
<tr>
<th>Packet/Frame Sync Sequence</th>
<th>Channel Estimation Sequence</th>
<th>Frame Header</th>
<th>Frame Payload</th>
</tr>
</thead>
<tbody>
<tr>
<td>24 OFDM Symbols</td>
<td>6 OFDM Symbols</td>
<td>12 OFDM Symbols</td>
<td>12 OFDM Symbols</td>
</tr>
<tr>
<td>Preamble: 30 × 0.3125 = 9.375 ( \mu )s</td>
<td>12 × 0.3125 = 3.75 ( \mu )s</td>
<td>( M_{f} ) × 0.3125 ( \mu )s</td>
<td></td>
</tr>
</tbody>
</table>

![Figure 4.1](image)

**Figure 4.1:** (a) Illustration of the OFDM-UWB frame structure; (b) Example of TFC for the \( m \)th multi-band OFDM symbol group with TFC = 1.

The six OFDM symbols in a group may be transmitted in multiple bands. The center frequency for the transmission of each OFDM symbol is prescribed by a time–frequency code (TFC). Fig. 4.1(b) shows one realization of TFC (corresponding to TFC = 1 as defined in [12]), where the first OFDM symbol of the \( m \)th group is transmitted on Sub-band 1, the second OFDM symbol is transmitted on Sub-band 2, the third OFDM symbol is transmitted on Sub-band 3, the fourth OFDM symbol is transmitted on Sub-band 1, and so on. Without loss of generality, we focus on TFC = 1 in the following. In this case, there are three sub-bands and each sub-band involves \( M_{1} = 2 \) channel estimation OFDM symbols and \( M_{2} = 4 \) frame header OFDM symbols.

Each OFDM symbol employs \( N = 128 \) subcarriers, which include \( R = 112 \) actual tones carrying useful information, \( R_{1} = 10 \) guard tones, and \( R_{2} = 6 \) virtual (null) tones. Of the \( R \) actual tones, \( P = 12 \) are assigned as pilot tones. The subcarrier frequency
4.1 Transmitter Model

spacing is $\Delta f_{sp} = 4.125$ MHz. Let us consider the generation of the $n$th OFDM symbol ($n = 6m + i$, $i \in \mathbb{Z}_5^0$, $m \in \mathbb{Z}_0^{M+2}$), and let

$$s_m^{(i)} = [s_m^{(i)}(0), s_m^{(i)}(1), \cdots, s_m^{(i)}(N-1)]^T$$

be a vector of $N$ complex symbols, where $s_m^{(i)}(k)$, $k \in \mathbb{Z}_0^{N-1}$, denotes the symbol modulating the $k$th subcarrier. Define a $P \times 1$ vector, $p = [p(0), p(1), \cdots, p(P-1)]^T = [5, 15, 25, 35, 45, 55, 73, 83, 93, 103, 113, 123]^T$, and let $R_0 = (R + R_1)/2$. The symbols $s_m^{(i)}(k)$ are drawn from a quadrature phase-shift keying (QPSK) constellation as specified in the Wimedia standard, denoted as $\pm c \pm jc$ with $j = \sqrt{-1}$ and $c = \sqrt{2}/2$, if $k \in \mathbb{Z}_i^{R_00} \cup \mathbb{Z}_i^{N-R_00}$. In particular, $s_m^{(i)}(k)$ are known at the receiver if $k \in \{p(l)\}_{l=0}^{P-1}$ since these constitute the pilot tones. Also, $s_m^{(i)}(k) = 0$ if $(k = 0) \cup k \in \mathbb{Z}_i^{N-R_00-1}$. The symbol vector, $s_m^{(i)}$, is fed to a $N$-point inverse discrete Fourier transform (IDFT) that yields a $N \times 1$ time-domain vector denoted by $x_m^{(i)}$. To eliminate the intersymbol interference (ISI) resulting from time dispersive channels, a $N_g$-point zero-padded (ZP) suffix is appended to each time-domain vector, thus forming an OFDM symbol as shown in Fig. 4.1(b).

Moreover, within the header’s OFDM modulation process, time-domain spreading is used by transmitting the same information across two consecutive header OFDM symbols, i.e.,

$$s_m^{(2i)}(k) = s_m^{(2i+1)}(k), \quad k \in \mathbb{Z}_i^{N-1}, \quad i \in \mathbb{Z}_5^0, \quad m \in \mathbb{Z}_1^2.$$ (4.2)

It should be also noted that, within each OFDM symbol contained in the header, a frequency-domain spreading technique is applied. That is,

$$s_m^{(i)}(k) = [s_m^{(i)}(N-k)]^*, \quad k \in \mathbb{Z}_i^{R/2}, \quad i \in \mathbb{Z}_5^0, \quad m \in \mathbb{Z}_1^2.$$ (4.3)

Such a spreading maximizes the frequency-diversity by transmitting the same information on two separate subcarriers within the same OFDM symbol. This feature will be exploited in the development of our channel estimator as shall be seen in Chapter 5.
4.2 UWB Channel Model

UWB channels can be described by adopting the Saleh–Valenzuela (S–V) model [119]. The S–V model uses a statistical process to model the discrete arrivals of the multipath components in clusters, as well as rays within a cluster. Mathematically, the impulse response of the multipath model is given by [24]

\[ h(t) = X \sum_{l=0}^{L} \sum_{k=0}^{K} \alpha_{k,l} \delta(t - T_l - \tau_{k,l}) \]

where \( \delta(t) \) is the Dirac delta function, \( L \) is the number of clusters, \( K \) is the number of rays in each cluster, \( T_l \) is the delay of the \( l \)th cluster’s first path, \( \tau_{k,l} \) is the delay of the \( k \)th multipath component (ray) relative to the \( l \)th cluster arrival time, \( \alpha_{k,l} \) is the multipath gain coefficient, and \( X \) represents the shadowing factor of propagation channels.

Denote by \( \Lambda \) and \( \lambda \) the cluster arrival rate and the ray arrival rate, respectively. The cluster arrival time and ray arrival time are described by the independent inter-arrival exponential probability density functions

\[ p(T_l|T_{l-1}) = \Lambda e^{-\Lambda(T_l-T_{l-1})}, \quad l > 0 \]
\[ p(\tau_{k,l}|\tau_{k-1,l}) = \lambda e^{-\lambda(\tau_{k,l}-\tau_{k-1,l})}, \quad k > 0. \]

The power delay profile is given by

\[ E[|\alpha_{k,l}|^2] = \Psi_0 e^{-\frac{T_l}{\Gamma}} e^{-\frac{\tau_{k,l}}{\gamma}} \]

where \( \Psi_0 \) is the mean energy of the first path of the first cluster, \( \Gamma \) is the cluster decay factor and \( \gamma \) is the ray decay factor. Moreover, the shadowing term \( X \) is modeled as a log-normal random variable, i.e., \( 20 \log_{10} X \sim \mathcal{N}(0, \sigma^2_X) \), while the total energy contained in the terms \( \alpha_{k,l} \) is normalized to unity for each channel realization, i.e.,

\[ \sum_{l=0}^{L} \sum_{k=0}^{K} \alpha_{k,l}^2 = 1. \]

Based on the S–V model and the measurements of actual channel environments, four types of indoor multipath channels, namely CM1, CM2, CM3 and CM4, are defined by the WiMedia Alliance with different values for parameters \( \{\Lambda, \lambda, \Gamma, \gamma, \sigma^2_X\} \), and each of them has 100 realizations [24]. Thus, in this thesis, we model the UWB channel in the
discrete time domain as a $N_h$-tap finite impulse response filter whose impulse response on the $r$th sub-band is denoted by

$$h_r^{(t)} = [h_r^{(t)}(0), h_r^{(t)}(1), \ldots, h_r^{(t)}(N_h - 1)]^T, \quad r \in \mathbb{Z}_1^3 \quad (4.4)$$

where the superscript $^{(t)}$ indicates time-domain. The corresponding channel frequency response (CFR) $h_r := [h_r(0), h_r(1), \ldots, h_r(N - 1)]^T$ is given by $h_r = F_{N_h} h_r^{(t)}$, where $F_{N_h}$ is the first $N_h$ columns of the $N$-point DFT matrix.

### 4.3 Modeling of PHN, CFO and SFO at Receiver

OFDM is sensitive to synchronization errors, which is partially due to the phase noise (PHN) from oscillators [121–127]. The PHN, $\phi(t)$, at the receiver can be modeled as a continuous Brownian motion process, which, from the standpoint of spectrum, can be described as a Wiener process with finite power [100, 128]. Denote by $\beta$ the two-sided 3 dB linewidth of the PHN’s Lorentzian\(^1\) power density spectrum, and by $\phi_n(k)$ the PHN on the $k$th sample of the $n$th OFDM symbol. The discrete-time samples of $\phi(t)$ form a random-walk process, which can be expressed as [129–131]

$$\phi_n(k) = \phi_{n-1}(N - 1) + \sum_{l=-N_g}^{k} \xi[n(N + N_g) + l], \quad k \in \mathbb{Z}_0^{N-1} \quad (4.5)$$

where $\xi(l) \sim \mathcal{N}(0, \sigma_\xi^2)$ is a Gaussian random variable with mean zero and variance $\sigma_\xi^2 = 2\pi\beta T_s$ with $T_s$ being the sampling interval, and $\xi(l) = 0$ for $l < 0$. We also assume that $\phi_n(k) = 0$ for $n < 0$ such that $\phi_0(k) = \sum_{l=0}^{k} \xi(l)$.

Clearly, the PHN process is characterized by the value of parameter $\beta$. The analysis in [27] shows that the design of local oscillators (LO) for an OFDM-UWB system should satisfy $\beta \leq 2 \times 7.7 = 15.4$ KHz, which\(^2\) corresponds to the Lorentzian spectrum with a power value of $-86.5$ dBC/Hz at 1 MHz. In fact, recent silicon implementations show

---

\(^1\)The power spectrum shaped by the squared magnitude of a one-pole lowpass filter transfer function is Lorentzian.

\(^2\)In [27], the parameter $\beta$ is defined as the one-sided 3 dB linewidth whereas it is the two-sided 3 dB linewidth in our case.
that PHN with power less than $-100$ dBc/Hz at 1 MHz is achievable \[132\]. Thus, in the following, we assume that $\beta = 6$ KHz (corresponding to $-90$ dBc/Hz at 1 MHz).

We further consider that a carrier frequency offset (CFO) of $\epsilon \Delta f_{sp}$ and a sampling frequency offset (SFO) of $\delta T_s$ are also present, where $\epsilon$ denotes normalized CFO and $\delta$ denotes normalized SFO. At the receiver, the received samples pass through a $N$-point DFT processor after the $N_g$ zero-padded points of each OFDM symbol are removed by using an overlap-and-add method (for converting linear convolution to circular convolution in a ZP-OFDM system \[134\]). We assume $N_h \leq N_g$ and that perfect timing synchronization (frame detection and symbol timing) can be achieved initially by using the first 24 OFDM symbols of the received preamble\(^3\). The output samples of the DFT processor corresponding to the $n$th ($n = 6m + i$) received OFDM symbol, $y_m^{(i)} = [y_m^{(i)}(0), y_m^{(i)}(1), \ldots, y_m^{(i)}(N - 1)]$, are given by \[95, 104\]

$$y_m^{(i)}(k) = \frac{1}{N} s_m^{(i)}(k) h_r(k) \kappa(\pi \rho_{kk}) e^{j \pi \rho_{kk}[2n(N+N_g)+N-1]/N} \sum_{l=0}^{N-1} e^{j \phi_n(l)} + n_m^{(i)}(k) + v_m^{(i)}(k) \quad (4.6)$$

$i \in \mathbb{Z}_N^5$, $r = |i|_3 + 1$, and $k \in \mathbb{Z}_N^{N-1}$. Here, we have defined $\kappa(\pi \rho_{qq}) := \frac{\sin(\pi \rho_{qq})}{N \sin(\pi \rho_{qq}/N)}$ and $\rho_{qq} := \bar{q} \delta + \epsilon (1 + \delta) + q - k$ with $\bar{q}$ given by

$$\bar{q} = \begin{cases} q, & q \in \mathbb{Z}_N^{N/2-1} \\ q - N, & q \in \mathbb{Z}_N^{N-1} \end{cases} \quad (4.7)$$

In (4.6), the term $v_m^{(i)}(k)$ denotes the channel noise on the $r$th sub-band ($r = |i|_3 + 1$), which is modelled in frequency-domain as a zero mean complex Gaussian process with variance $\sigma_r^2$, and the term $n_m^{(i)}(k)$, representing inter-carrier interference (ICI), is given by

$$n_m^{(i)}(k) = \frac{1}{N} \sum_{q=0, q \neq k}^{N-1} \left\{ s_m^{(i)}(q) h_r(q) e^{j 2 \pi q (N+N_g) (\rho_{qq} + k-q) / N} \sum_{l=0}^{N-1} e^{j [2 \pi l (\rho_{qq} + \phi_n(l))] \right\}. \quad (4.8)$$

It can be shown that $n_m^{(i)}(k)$ is a zero mean random variable independent of $v_m^{(i)}(k)$ \[102\].

\(^3\)The assumption $N_h \leq N_g$ may not always be strictly correct especially when CM4 is considered. The resulting ISI may slightly affect the consistency between the related analytical and simulation results. However, as we will show in next chapter, ISI has no significant impairment to the effectiveness of the proposed channel estimator in this case.
Furthermore, we assume that an initial estimate of CFO is obtained using the first
24 OFDM symbols of the received preamble and it is used to compensate the CFO in all
the subsequent symbols [96, 108–110, 114]. In other words, the value of ε represents the
residual CFO (normalized by the subcarrier-spacing $\Delta f_{sp}$) after the initial compensation
and thus we can safely assume $|\epsilon| < 0.02$ [50, 99, 135]. We also assume that $\delta \ll 1$. Since
the SFO is estimated and compensated in a continuous manner, i.e., via a clock recovery
loop, as shall be clear from our later discussion, the contribution of residual SFO to $n^{(i)}_{m}(k)$
in (4.8) becomes negligible. Under this assumption of no residual SFO, the variance of
ICI noise in (4.6) is reduced to [104]

$$
\sigma_{ICI}^2(k) = \mathbb{E}\{|n^{(i)}_{m}(k)|^2\} = \sum_{q \in \mathbb{Z}_{R_0} \cup \mathbb{Z}_{N-R_0}^{N-1}} |h_r(q)|^2 \left\{ \frac{1}{N} + \frac{2}{N^2} \sum_{l=1}^{N-1} (N-l) \cos \left[ \frac{2\pi l(q-k+\epsilon)}{N} \right] e^{-\pi l\beta T_s} \right\}
$$

(4.9)

for $k \in \mathbb{Z}_{0}^{N-1}$. Denote by $\text{CIR}(k)$ the carrier-to-ICI ratio (CIR) and $\text{CNR}(k)$ the
carrier-to-noise ratio (CNR) at the $k$th subcarrier, and define

$$
\rho(k) = \frac{\text{CIR}(k)}{\text{CNR}(k)} = \frac{\sigma_r^2}{\sigma_{ICI}^2(k)}.
$$

(4.10)

Using (4.9) and (4.10), $\rho(k)$ can be evaluated for each individual channel realization. In
fact, our experimental results show that $\rho(k)$ can be approximately evaluated by simply
considering the flat fading channel environment, i.e., $|h_r(q_1)|^2 = |h_r(q_2)|^2$ for all $q_1 \in \mathbb{Z}_{1}^{R_0} \cup \mathbb{Z}_{N-R_0}^{N-1}$, $q_2 \in \mathbb{Z}_{1}^{R_0} \cup \mathbb{Z}_{N-R_0}^{N-1}$ and $q_1 \neq q_2$, in which $\rho(k)$ is given by

$$
\rho(k) = \frac{1}{\text{SNR}_r \sum_{q \in \mathbb{Z}_{1}^{R_0} \cup \mathbb{Z}_{N-R_0}^{N-1}} \frac{1}{N} + \frac{2}{N^2} \sum_{l=1}^{N-1} (N-l) \cos \left[ \frac{2\pi l(q-k+\epsilon)}{N} \right] e^{-\pi l\beta T_s}}
$$

(4.11)

where $\text{SNR}_r$ is the average SNR over all subcarriers at the receiver. Replacing $\beta$ and $\epsilon$
in (4.11) with the worst-case assumptions $\beta = 6$ KHz and $\epsilon = \pm 0.02$, respectively, and
noting that the operational SNR is typically less than 20 dB for an OFDM-UWB device,
we have $\min_{k \in \mathbb{Z}_{1}^{R_0} \cup \mathbb{Z}_{N-R_0}^{N-1}} \rho(k) = 5.5$ dB. Thus, when compared with channel noise, ICI
can be neglected in this case. Consequently, in the sequel, we ignore the ICI term, $n^{(i)}_{m}(k)$,
in (4.6).
4.4 Concluding Remarks

Let $\chi_m^{(i)} = (6m + i)(N + N_d) + (N - 1)/2$. Denote by $\eta_m^{(i)} = \chi_m^{(i)}$ the accumulated SFO (normalized by $T_s$, and is not compensated) when receiving the $n$th ($n = 6m + i$) OFDM symbol. Taking into account that $\epsilon, \delta$ and PHN are all small, we have $\kappa(\pi \rho_{kk}) \approx 1$. Thus, we can rewrite (4.6) as

$$y_m^{(i)}(k) \approx s_m^{(i)}(k) h_r(k) e^{j[\phi_m^{(i)} + \varphi_m^{(i)}(k)] + \psi_m^{(i)}(k)}$$

(4.12)

where

$$\phi_m^{(i)} \approx \arg \left( \frac{1}{N} \sum_{l=0}^{N-1} e^{j\phi_n(l)} \right), \quad (4.13)$$

$$\varphi_m^{(i)} \approx 2\pi/N \cdot \epsilon(1 + \delta)\chi_m^{(i)} \quad (4.14)$$

and

$$\psi_m^{(i)}(k) \approx \begin{cases} 2\pi k/N \cdot \eta_m^{(i)}, & k \in \mathbb{Z}_{0}^{N/2-1} \\ 2\pi(k - N)/N \cdot \eta_m^{(i)}, & k \in \mathbb{Z}_{N/2}^{N-1}. \end{cases} \quad (4.15)$$

Note that the approximation in (4.13) follows from the fact that, for a slow PHN process, the term inside the bracket on the RHS of (4.13) manifests as a phase rotation [128]. Obviously, the phase-shift, $\varphi_m^{(i)} = \phi_m^{(i)} + \varphi_m^{(i)}$, called common phase error (CPE), is mainly related to PHN and CFO and independent of the subcarrier index, $k$, whereas the phase-shift, $\psi_m^{(i)}(k)$, caused by SFO, is proportional to $k$. As a result, estimation and compensation of CPE and SFO can be easily decoupled as will be described in Chapter 6.

4.4 Concluding Remarks

In this chapter, we have presented a complete signal model for multi-band OFDM-UWB systems. This model is the basis for developing effective channel estimation and phase suppression algorithms in Chapters 5 and 6. To the best of our knowledge, so far, no similar joint modeling of PHN, CFO and SFO for OFDM-UWB devices has been reported in the literature. In particular, our analysis shows that the effect of ICI, which is relatively weak when compared to that of channel noise, can be neglected in this case. This has
4.4 Concluding Remarks

reduced the effort in developing the related algorithms for OFDM-UWB receivers, as we will see in Chapter 6.
Chapter 5

Channel Estimation for Multi-Band OFDM-UWB Systems

In this chapter, we propose an efficient channel estimation scheme for multi-band orthogonal frequency-division multiplexing (OFDM) based ultra-wideband (UWB) communication systems, and more specifically, for practical implementation of low-cost and high-speed UWB-based wireless Universal Serial Bus (USB) devices. The proposed channel estimator consists of two stages. The first stage employs a simple least-squares (LS) method together with a frequency-domain smoothing operation for estimating the channel using the available training sequence. During the second stage, this channel estimate is used for detecting the frame header, and then a refined channel estimate is obtained by using a decision-directed (DD) technique. The mean-squared error (MSE) performance and computational complexity of the proposed scheme are analyzed. Numerical examples show that the proposed scheme substantially outperforms the conventional LS approach and it performs comparably to the maximum-likelihood (ML) estimator, under various highly noisy multipath channel conditions.
5.1 Assumptions and Definitions

In this discussion, we assume that the UWB channel is invariant over the transmission period of one OFDM frame. In fact, the phase-shift related components of the first term on the RHS of (4.12) can be temporarily dropped in this case as they either are negligible due to the fact that $m$ is close to 0 in the context of channel estimation, or, can be treated as a part of the channel frequency response (CFR) itself. Thus, for channel estimation, (4.12) can be rearranged as

$$y_m^{(i)}(k) \approx s_m^{(i)}(k)h_r(k) + v_m^{(i)}(k). \quad (5.1)$$

We define the estimate of $h_r$ on the $r$th sub-band as

$$\hat{h}_r^q = [\hat{h}_r^q(0), \hat{h}_r^q(1), \cdots, \hat{h}_r^q(N-1)], \quad r \in \mathbb{Z}_3^2.$$ The proposed channel estimation consists of five steps, which are further grouped into two stages. The first stage includes the first two steps and the second stage the rest.

Let $\hat{h}_r^q = [\hat{h}_r^q(0), \hat{h}_r^q(1), \cdots, \hat{h}_r^q(N-1)], \quad r \in \mathbb{Z}_3^2,$ be the estimate of $h_r$ after the $q$th step. The normalized MSE (NMSE) of this estimation is defined by

$$\text{MSE}_r^q(k) = E\{|\hat{h}_r^q(k) - h_r(k)|^2\}/E_0, \quad k \in \mathbb{Z}_2^{R/2} \cup \mathbb{Z}_{N-R/2}^{N-1}, \quad r \in \mathbb{Z}_3^3, \quad q \in \mathbb{Z}_5^5$$

where $E_0 = E\{|h_r(k)|^2\} = e^{0.0265\sigma_x^2}$, which is same for all $k \in \mathbb{Z}_2^{R/2} \cup \mathbb{Z}_{N-R/2}^{N-1}$ [136, Eq. (8)]. Note that $\sigma_x^2$ is a UWB channel parameter described in Section 4.2. Also, $\text{MSE}_r^q(k)$ is found to be independent of the subcarrier index, $k$, as shall be clear from our later discussion. Thus, in the sequel, we omit the index $k$ in $\text{MSE}_r^q(k)$.

5.2 Stage 1 – Primary CFR Estimation

In the first step, since $|s_m^{(i)}(k)|^2 = 2c^2 = 1$, we can obtain $\hat{h}_r^{(1)}$ from (5.1) as

$$\hat{h}_r^{(1)}(k) = \frac{1}{M_1} \left( y_0^{(r-1)}(k)/s_0^{(r-1)}(k) + y_0^{(r+2)}(k)/s_0^{(r+2)}(k) \right)$$

$$= \frac{1}{M_1} \left\{ y_0^{(r-1)}(k)[s_0^{(r-1)}(k)]^* + y_0^{(r+2)}(k)[s_0^{(r+2)}(k)]^* \right\}, \quad (5.2)$$

for $k \in \mathbb{Z}_2^{R/2} \cup \mathbb{Z}_{N-R/2}^{N-1}, \quad r \in \mathbb{Z}_3^3$, where $M_1 = 2$ is the number of channel estimation OFDM symbols per sub-band. Clearly, this is a LS estimate using the dedicated channel
estimation sequence, with 3 dB gain in estimation accuracy resulting from averaging two results \((M_1 = 2)\) obtained from the same sub-band. In this case, we have 
\[
\mathbb{MSE}_r^{(1)} = \frac{1}{(M_1 \cdot \text{SNR}_r)},
\]
where \(\text{SNR}_r\) is the average SNR over all subcarriers at the receiver. Without loss of generality, hereafter, we assume that all three subbands have the same average SNR.

In the second step, we apply a simple frequency-domain smoothing operation to \(\hat{h}_r^{(1)}\) and obtain \(\hat{h}_r^{(2)}\) as
\[
\hat{h}_r^{(2)}(k) = \alpha_h [\hat{h}_r^{(1)}(k - 1) + \hat{h}_r^{(1)}(k + 1)] + (1 - 2\alpha_h) \hat{h}_r^{(1)}(k),
\]
for \(k \in \mathbb{Z}_{1}^{R/2} \cup \mathbb{Z}_{N-R/2}^{N-1}\), where \(0 < \alpha_h < 0.5\). By doing so, the CFR estimate on each subcarrier is smoothed using the estimates from adjacent subcarriers such that the residual error contained in the initial LS estimate can be reduced. Note that when \(k \in \{1, R/2, N-R/2, N-1\}\), the \(k\)th subcarrier has only one valid adjacent subcarrier. In this case, (5.3) can be modified such that only one adjacent subcarrier is used for smoothing.

The validity of using the above frequency-domain smoothing technique can be justified by examining the relationship between channel coherent bandwidth and subcarrier spacing. Denote by \(\text{corr}(\Delta k)\) the normalized cross-correlation of \(h_r(k)\), i.e.,
\[
\text{corr}(\Delta k) := \frac{E[h_r(k + \Delta k) h_r^*(k)]}{E[|h_r(k)|^2]}.
\]
Then, from (5.4), we have
\[
|\text{corr}(\Delta k)| \approx \left| \frac{1 + j 2\pi \Delta k \gamma / (NT_s)}{1 + \Lambda \Gamma} \right| \left| \frac{1 + j 2\pi \Delta k \gamma / (NT_s)}{1 + \lambda \gamma} \right|,
\]
where \(T_s\) is the sampling interval of received signals and \(\Delta k\) is a small integer which is close to 0. Applying the actual values of \(\Lambda, \lambda, \Gamma\) and \(\gamma\) (see [24, Table II]) to (5.4), we find that \(|\text{corr}(\Delta k)| = 0.99, 0.98, 0.94\) and 0.84 for CM1, CM2, CM3 and CM4, respectively, when \(|\Delta k| = 1\). Since the UWB channel coherent bandwidth is much larger than the subcarrier spacing, \(\hat{h}_r^{(2)}\) will be close to an unbiased estimate of \(h_r\) as long as the smoothing parameter \(\alpha_h\) is sufficiently small [26]. Therefore, the use of frequency-domain smoothing for channel estimation in this case is appropriate.

The actual choice of the smoothing parameter \(\alpha_h\) should also take into account the resulting \(\mathbb{MSE}_r^{(2)}\). From (5.2) and (5.3), we can obtain the value of \(\alpha_h\), which is optimal in
the sense of minimizing $\text{MSE}^{(2)}$ (see Appendix A for detailed derivation). The closed-form expression for $\alpha_h$ is given by

$$\alpha_h^{\text{opt}} = \frac{1}{3 + M_1(3 - \Re[4\text{corr}(1) - \text{corr}(2)])\text{SNR}_r}. \quad (5.5)$$

Obviously, setting $\alpha_h$ in (5.3) to be its optimum value requires the knowledge of channel statistics and SNR, which may not be available in practice. A practical yet sub-optimal solution can be achieved by evaluating $\text{MSE}^{(2)}$ over the entire SNR ranges of interest for the four types of UWB channels. Defining $R_{\text{mse}}^{1,2} := \text{MSE}^{(1)}_{\text{r}}/\text{MSE}^{(2)}_{\text{r}}$, from (A.5), we have

$$R_{\text{mse}}^{1,2} = \frac{1}{2M_1(3 - \Re[4\text{corr}(1) - \text{corr}(2)])\alpha_h^2\text{SNR}_r + 6\alpha_h^2 - 4\alpha_h + 1}. \quad (5.6)$$

Fig. 5.1 shows the NMSE ratio, $R_{\text{mse}}^{1,2}$, versus $\alpha_h$ and SNR$_r$ for the four different types of UWB channels. The SNR range is chosen to be $-5 \sim 20$ dB in the cases of CM1 and CM2 and $-5 \sim 9$ dB in the cases of CM3 and CM4 since the latter are only applicable to lower rate transmission with lower high-end operational SNR’s [24]. Since a positive $R_{\text{mse}}^{1,2}$ (in dB) indicates that $\hat{h}^{(2)}_r$ is more accurate than the initial LS estimate, $\hat{h}^{(1)}_r$, we can conclude from Fig. 5.1 that a good smoothing factor should satisfy $0 < \alpha_h \leq 0.1$ so that we can achieve an unbiased CFR estimate with $R_{\text{mse}}^{1,2} > 0$ dB in all scenarios\(^1\).

The channel estimate, $\hat{h}^{(2)}_r$, obtained from the second step will be used to process the frame header. The header processing in turn leads to the second stage of channel estimation which is described next.

### 5.3 Stage 2 – Enhanced CFR Estimation

In this stage, we first (i.e., in the third step) introduce an efficient DD detection based semiblind CFR estimation by exploiting the frequency-domain spreading property of frame header and the finite-alphabet feature of QPSK modulation. Denoting by $s^{(i)}_m(k)$, $k \in \mathbb{Z}_1^{R/2} \cup \mathbb{Z}_{N-R/2}^{N-1}$ and $k \notin \{p(l)\}_{l=0}^{P-1}$, the detected signals of a header OFDM symbol

\(^1\)Note that the actual propagation environments of UWB may be different from those described by CM1 to CM4. However, as long as the ranges of corr(1), corr(2) and SNR$_r$ are roughly available, a desirable choice of $\alpha_h$ can be easily made using (5.6).
5.3 Stage 2 – Enhanced CFR Estimation

Figure 5.1: NMSE ratio, \( R_{\text{mse}}^{1,2} \), versus smoothing parameter \( \alpha_h \) and SNR\(_r\) under various channel environments. (a) CM1, (b) CM2, (c) CM3, and (d) CM4.

(after the one-tap, frequency-domain equalization using \( \hat{h}_r^{(2)} \)) are usually obtained as

\[
\hat{s}_m^{(i)}(k) = \frac{y_m^{(i)}(k)}{\hat{h}_r^{(2)}(k)}, \quad i \in \mathbb{Z}_5, \ m \in \mathbb{Z}_2^1, \ r = |i|_3 + 1. \tag{5.7}
\]

Correspondingly, let

\[
u_m^{(i)} = [u_m^{(i)(0)}, u_m^{(i)(1)}, \ldots, u_m^{(i)(N - 1)}]^T
\]

and

\[
\nu_m^{(i)} = [v_m^{(i)(0)}, v_m^{(i)(1)}, \ldots, v_m^{(i)(N - 1)}]^T
\]

where

\[
u_m^{(2q)}(k) = \nu_m^{(2q+1)}(k) = \text{sgn}\left[ \text{Re}\left( z_m^{(2q)}(k) + z_m^{(2q)}(N - k) + z_m^{(2q+1)}(k) + z_m^{(2q+1)}(N - k) \right) \right]
\]

\[
u_m^{(2q)}(k) = \nu_m^{(2q+1)}(k) = \text{sgn}\left[ \text{Im}\left( z_m^{(2q)}(k) - z_m^{(2q)}(N - k) + z_m^{(2q+1)}(k) - z_m^{(2q+1)}(N - k) \right) \right] \tag{5.8}
\]

with

\[
z_m^{(i)}(k) = \left| \hat{h}_r^{(2)}(k) \right|^2 \hat{s}_m^{(i)}(k) = y_m^{(i)}(k) \left[ \hat{h}_r^{(2)}(k) \right]^* \tag{5.9}
\]
for \( k \in \mathbb{Z}_{1}^{R/2} \cup \mathbb{Z}_{N-R/2}^{N-1} \) and \( k \notin \{p(l)\}_{l=0}^{P-1}, i \in \mathbb{Z}_{0}^{5}, q \in \mathbb{Z}_{0}^{2} \) and \( m \in \mathbb{Z}_{1}^{2} \). It should be noted that multiplying \( \hat{s}_{m}^{(i)}(k) \) by \( |\hat{h}_{r}^{(2)}(k)|^{2} \) in (5.9) yields a weighted combination of two frequency-domain spread signals in (5.8) in a way similar to the maximum ratio combining (MRC) principle (see, for example, [92]), but with lower complexity as no division is required for obtaining \( z_{m}^{(i)}(k) \) and \( z_{m}^{(i)}(N-k) \). The weighting factor \( |\hat{h}_{r}^{(2)}(k)|^{2} \) is the CFR magnitude (squared) on the \( k \)-th subcarrier. The larger is \( |\hat{h}_{r}^{(2)}(k)|^{2} \), the more reliable will be the detected header symbol (i.e., \( \hat{s}_{m}^{(i)}(k) \)) on the \( k \)-th subcarrier, and vice versa. Hence, compared with the direct combination of \( \hat{s}_{m}^{(i)}(k) \) and \( \hat{s}_{m}^{(i)}(N-k) \) (i.e., replacing \( z_{m}^{(i)}(k) + z_{m}^{(i)}(N-k) \) and \( z_{m}^{(i)}(k) - z_{m}^{(i)}(N-k) \) with \( \hat{s}_{m}^{(i)}(k) + \hat{s}_{m}^{(i)}(N-k) \) and \( \hat{s}_{m}^{(i)}(k) - \hat{s}_{m}^{(i)}(N-k) \), respectively, in (5.8)), the weighted combination in (5.8) results in a CFR-weighted detection that is more noise-resilient and reliable.

Using the detected header symbols, as well as the pilots, we now obtain a decision-directed channel estimate, \( \hat{h}_{r}^{(3)} \), in the third step, as

\[
\hat{h}_{r}^{(3)}(k) = \left\{ \begin{array}{ll}
\frac{1}{M_{2}} \sum_{m=1}^{2} \sum_{q=r-2}^{q=r+2} \{ y_{m}^{(q)}(k) [r_{m}^{(q)}(k) - j i_{m}^{(q)}(k)] \}, & k \in \mathbb{Z}_{1}^{R/2} \cup \mathbb{Z}_{N-R/2}^{N-1} \text{ and } k \notin \{p(l)\}_{l=0}^{P-1} \\
\frac{1}{M_{2}} \sum_{m=1}^{2} \sum_{q=r-2}^{q=r+2} \{ y_{m}^{(q)}(k) [s_{m}^{(q)}(k)]^{*} \}, & k \in \{p(l)\}_{l=0}^{P-1}.
\end{array} \right.
\]

(5.10)

In the fourth step of channel estimation, we further apply the frequency-domain smoothing introduced in the second step to \( \hat{h}_{r}^{(3)} \). The resulting CFR estimation, \( \hat{h}_{r}^{(4)} \), is given by

\[
\hat{h}_{r}^{(4)}(k) = \beta_{h}[\hat{h}_{r}^{(3)}(k - 1) + \hat{h}_{r}^{(3)}(k + 1)] + (1 - 2\beta_{h})\hat{h}_{r}^{(3)}(k)
\]

(5.11)

for \( k \in \mathbb{Z}_{1}^{R/2} \cup \mathbb{Z}_{N-R/2}^{N-1} \), where \( \beta_{h} \) is a smoothing factor whose value can be determined following a procedure similar to that for choosing \( \alpha_{h} \).

Finally, in the fifth step, we obtain \( \hat{h}_{r} \) as a weighted average of \( \hat{h}_{r}^{(2)} \) and \( \hat{h}_{r}^{(4)} \) as

\[
\hat{h}_{r} = \hat{h}_{r}^{(5)} = (M_{1}\hat{h}_{r}^{(2)} + M_{2}\hat{h}_{r}^{(4)})/(M_{1} + M_{2}), \quad r \in \mathbb{Z}_{1}^{2}.
\]

(5.12)

The final CFR estimate \( \hat{h}_{r} \) is more accurate than \( \hat{h}_{r}^{(2)} \) obtained in the first stage and will be used for processing the payload OFDM symbols in the current frame. The above
Table 5.1: Summary of the proposed multi-stage channel estimation scheme. ($k \in \mathbb{Z}^{R/2} \cup \mathbb{Z}^{N-1-R/2}$ and $r \in \mathbb{Z}^3$.)

**BEGINSTAGE 1:** Receiving $M_1$ channel estimation OFDM symbols, $y_0^{(r-1)}$ and $y_0^{(r+2)}$, from the $r$th sub-band.

**Step 1:**
\[
\hat{h}_r^{(1)}(k) = \frac{1}{M_1} \left\{ y_0^{(r-1)}(k)[s_0^{(r-1)}(k)]^* + y_0^{(r+2)}(k)[s_0^{(r+2)}(k)]^* \right\}.
\]

**Step 2:**
\[
\hat{h}_r^{(2)}(k) = \alpha_h [\hat{h}_r^{(1)}(k-1) + \hat{h}_r^{(1)}(k+1)] + (1 - 2\alpha_h)\hat{h}_r^{(1)}(k).
\]

**ENDSTAGE 1:** Using $\hat{h}_r^{(2)}$ for frame header processing.

**BEGINSTAGE 2:** Receiving $M_2$ header OFDM symbols, $y_1^{(r-1)}$, $y_1^{(r+2)}$, $y_2^{(r-1)}$ and $y_2^{(r+2)}$, from the $r$th sub-band.

**Step 3:** (Note that $q_1 = q - 2|q|_2 + 1$ and $r_1 = |q_1|_3 + 1$)
\[
\hat{h}_r^{(3)}(k) = \begin{cases} 
\frac{c}{M_2} \sum_{m=1}^{2} \sum_{q=q_1}^{2} \left\{ y_m^{(q)}(k)[s_m^{(q)}(k)]^* \right\}, & k \notin \{p(l)\}_{l=0}^{P-1} \\
\frac{c}{M_2} \sum_{m=1}^{2} \sum_{q=q_1}^{2} \left\{ y_m^{(q)}(k)[s_m^{(q)}(k)]^* \right\}, & k \in \{p(l)\}_{l=0}^{P-1}.
\end{cases}
\]

**Step 4:**
\[
\hat{h}_r^{(4)}(k) = \beta_h [\hat{h}_r^{(3)}(k-1) + \hat{h}_r^{(3)}(k+1)] + (1 - 2\beta_h)\hat{h}_r^{(3)}(k).
\]

**Step 5:**
\[
\hat{h}_r^{(5)}(k) = \frac{M_1\hat{h}_r^{(2)}(k) + M_2\hat{h}_r^{(4)}(k)}{(M_1 + M_2)}.
\]

**ENDSTAGE 2:** Using $\hat{h}_r^{(5)}$ for payload processing.

two-stage channel estimation can be summarized as shown in Table 5.1.
5.4 Performance Analysis

We observe from Section 5.3 that the actual performance enhancement resulting from the third step depends on the CFR-weighted detection performance. Suppose that a header symbol, \( s_m^{(i)}(k) \), is transmitted on \( k \)-th subcarrier, and the corresponding received and detected symbols are \( y_m^{(i)}(k) \) and \( c[u_m^{(i)}(k) + jv_m^{(i)}(k)] \), respectively. Denoting by \( h^d_m(k) \) and \( h^r_m(k) \) the LS estimates of CFR on the \( k \)-th subcarrier obtained by using the detected header symbol and by assuming that the transmitted header symbol is known at the receiver, respectively, we have

\[
h^d_m(k) = c \cdot y_m^{(i)}(k)[u_m^{(i)}(k) - jv_m^{(i)}(k)], \quad h^r_m(k) = y_m^{(i)}(k)s_m^{(i)\ast}(k). \quad (5.13)
\]

Apparently, the NMSE of \( h^r_m(k) \) is given by \( \text{MSE}_s = 1/\text{SNR}_r \).

In order to derive the NMSE of \( h^d_m(k) \), \( \text{MSE}_d \), we first need to consider the header detection error. Denote by \( P_e \) the average bit error probability of the proposed CFR-weighted header OFDM symbol detector, i.e., \( P_e = \Pr \{ u_m^{(i)}(k)\Re[s_m^{(i)}(k)] < 0 \} = \Pr \{ v_m^{(i)}(k)\Im[s_m^{(i)}(k)] < 0 \} \). Thus, from (5.13), we can obtain

\[
\text{MSE}_d = \frac{\mathbb{E}\{|h^d_m(k) - h_r(k)|^2\}}{E_0}
= \frac{\mathbb{E}\{|c \cdot y_m^{(i)}(k)[u_m^{(i)}(k) - jv_m^{(i)}(k)] - h_r(k)|^2\}}{E_0}
= \frac{\mathbb{E}\{|c \cdot h_r(k)s_m^{(i)}(k)[u_m^{(i)}(k) - jv_m^{(i)}(k)] - h_r(k)|^2\} + \sigma_r^2}{E_0}
= \mathbb{E}\{2 - (u_m^{(i)}(k)\Re[s_m^{(i)}(k)] + v_m^{(i)}(k)\Im[s_m^{(i)}(k)])/c\} + 1/\text{SNR}_r
= 4P_e + 1/\text{SNR}_r. \quad (5.14)
\]

We further show in Appendix B that \( P_e \) is approximately upper-bounded by \( P_e^{\text{ub}} \) as

\[
P_e \leq P_e^{\text{ub}} = \frac{1}{\sqrt{2\pi}\sigma_x} \int_{-\infty}^{\infty} Q \left( 2 \sqrt{\frac{([M_1\text{SNR}_r + 1]\text{SNR}_r + 1)\text{SNR}_r(10\pi)}{E_0}} \cdot 10\pi \right) e^{-\frac{x^2}{2\sigma_x^2}} dx \quad (5.15)
\]

where \( Q(\cdot) \) denotes the complementary cumulative distribution function of the standard Gaussian distribution. Let \( \text{MSE}_b = 4P_e^{\text{ub}} + 1/\text{SNR}_r \). Obviously, \( \text{MSE}_b \) can be interpreted
as the approximate upper-bound of \( \text{MSE}_d \). We define two equivalent SNR’s, \( \text{SNR}_d \) and \( \text{SNR}_b \), for \( \text{MSE}_d \) and \( \text{MSE}_b \), respectively, such that \( \text{MSE}_d = 1/\text{SNR}_d \) and \( \text{MSE}_b = 1/\text{SNR}_b \). Consequently, from (5.14) and (5.15), we have

\[
\text{SNR}_d \geq \text{SNR}_b = \frac{\text{SNR}_r}{1 + \frac{4\text{SNR}_r}{\sqrt{2\pi} \sigma_x} \int_{-\infty}^{\infty} Q \left( 2 \sqrt{\frac{[M_1 R_1 \text{SNR}_r + 1] \text{SNR}_r}{[M_1 R_1 \text{SNR}_r + 1] \text{SNR}_r + 1} \cdot 10^\frac{x}{10} \right)} \cdot e^{-\frac{x^2}{2\sigma_x^2}} \, dx. \tag{5.16}
\]

Fig. 5.2 shows the comparison between \( \text{MSE}_b \) and \( \text{MSE}_s \) under various channel environments. Clearly, the analytically computed \( \text{MSE}_b \) curves are close to their respective \( \text{MSE}_s \) curves in the entire SNR region of practical interest. Thus, it can be concluded that, when applied for channel re-estimation, a detected header OFDM symbol obtained using the proposed CFR-weighted detector should perform similar to a known channel training OFDM symbol, under all channel environments. This is confirmed further via simulations as shown in Fig. 5.2, i.e., all the \( \text{MSE}_d \) curves are close to their respective \( \text{MSE}_s \) curves in the entire SNR region of practical interest. In particular, Fig. 5.2 also shows the NMSE performances of the channel estimate obtained with the simple average detector, which is formed by replacing \( \hat{z}_m^{(i)}(k) + \hat{z}_m^{(i)}(N - k) \) and \( \hat{z}_m^{(i)}(k) - \hat{z}_m^{(i)}(N - k) \) with the direct combinations \( \hat{s}_m^{(i)}(k) + \hat{s}_m^{(i)}(N - k) \) and \( \hat{s}_m^{(i)}(k) - \hat{s}_m^{(i)}(N - k) \) in (5.8), as described in Section 5.3. In comparison, the proposed CFR-weighted detector results in much better MSE performance.

From (5.16), we have \( \text{MSE}_b \geq \text{MSE}_d \). As expected, we further observe from Fig. 5.2 that all \( \text{MSE}_d \) curves lie slightly below their analytical counterparts \( \text{MSE}_b \) under all channel environments with two exceptions. The first exception occurs in the very low SNR regime, where the validity of \( \text{MSE}_b \geq \text{MSE}_d \) is slightly violated. This can be attributed to the fact that (B.4) and further (B.3) in Appendix B give only loose approximations in this case. The second exception occurs in the case of CM4, where the simulated \( \text{MSE}_d \) slightly exceeds the analytical \( \text{MSE}_b \) in the high SNR regime (very slightly in CM3 also). In fact, this abnormal phenomenon is due to the inter-symbol interference (ISI) effect which has not been considered in the analysis of \( \text{MSE}_b \) but becomes serious when CM4 is used for simulations (to obtain \( \text{MSE}_d \)). The ISI effect becomes more evident in the case of \( \text{MSE}_s \) where ISI is the only factor that affects the
consistency between the analytical $\text{MSE}_s$ curve and its simulated counterpart for CM4 in Fig. 5.2. But even with these two exceptions, we observe that, in the entire SNR region
of practical interest, the simulation results of $\text{MSE}_d$ are close to the analytical $\text{MSE}_b$ under all channel environments. We therefore use the assumption $\text{MSE}_d \approx \text{MSE}_b$, or equivalently, $\text{SNR}_d \approx \text{SNR}_b$, in the subsequent discussion.

An alternative way to reduce the detrimental effect of detection errors in the third step of our CFR estimator is to use decoded data instead of the output signal directly from the detector (equalizer). As described in [12], a convolutional channel encoder together with a Reed Solomon encoder are employed at the transmitter such that the frame header information at the receiver can be reliably decoded. The recovered frame header sequence, $\hat{s}_m^{(i)}(k), i \in \mathbb{Z}_0^5, m \in \mathbb{Z}_1^2$, should be same as the transmitted sequence, $s_m^{(i)}(k)$, and thus it can be reused as an additional channel estimation sequence. However, practical implementation of this scheme is difficult due to the high-speed processing requirement and the limited hardware resource in a UWB device. Therefore, our proposed CFR-weighted detection scheme constitutes a desirable practical solution for this step since it is simple while almost error-free.

As shown in Appendix A, the NMSE of the CFR estimation resulting from the first stage is given by

$$\text{MSE}_r^{(2)} = 2 \left( 3 - \Re[4\text{corr}(1) - \text{corr}(2)] \right) \alpha_h^2 + \frac{(6\alpha_h^2 - 4\alpha_h + 1)}{M_1 \cdot \text{SNR}_r}. \quad (5.17)$$

Similarly, following the above discussion that $\text{MSE}_d \approx \text{MSE}_b = 1/\text{SNR}_b$, we can obtain

$$\text{MSE}_r^{(4)} \approx 2 \left( 3 - \Re[4\text{corr}(1) - \text{corr}(2)] \right) \beta_h^2 + \frac{(6\beta_h^2 - 4\beta_h + 1)}{M_2 \cdot \text{SNR}_b}. \quad (5.18)$$

Under the assumption that all three sub-bands have similar $\text{SNR}_r$’s and/or $\text{SNR}_b$’s, from (5.12), the NMSE of the final channel estimate averaged over three sub-bands can be obtained as

$$\text{MSE} = \frac{\sum_{r=1}^{3} \mathbb{E}\{\|\hat{h}_r - h_r\|^2\}}{\sum_{r=1}^{3} \mathbb{E}\{\|h_r\|^2\}} = \frac{M_1^2 \cdot \text{MSE}_r^{(2)} + 2M_1M_2 \cdot C_{24} + M_2^2 \cdot \text{MSE}_r^{(4)}}{(M_1 + M_2)^2}. \quad (5.19)$$

where $C_{24} = \mathbb{E}\{(\hat{h}_r^{(4)} - h_r)(\hat{h}_r^{(2)} - h_r)^\dagger\}/\mathbb{E}\{\|h_r\|^2\}$. Following the derivation of $C_{24}$ in Appendix B, we obtain

$$C_{24} \approx \alpha_h (2 - 1/\text{SNR}_b + 1/\text{SNR}_r) (3\beta_h - 1 + \Re[(1 - 4\beta_h)\text{corr}(1) + \beta_h \cdot \text{corr}(2)])$$

$$+ 2\alpha_h (1 - \Re[\text{corr}(1)]). \quad (5.20)$$
5.4 Performance Analysis

Defining \( R_{\text{mse}} = \frac{\text{MSE}^{(1)}}{\text{MSE}} \) and using (5.17) to (5.20), we obtain

\[
R_{\text{mse}} \approx \frac{(M_1 + M_2)^2}{M_1} \cdot \{ 2(M_1^2 \alpha_h^2 + M_2^2 \beta_h^2)(3 - \Re[4\text{corr}(1) - \text{corr}(2)])\text{SNR}_r \\
+ M_1(6\alpha_h^2 - 4\alpha_h + 1) + M_2(6\beta_h^2 - 4\beta_h + 1)\text{SNR}_r/\text{SNR}_b \\
+ 2M_1M_2\alpha_h [(2\text{SNR}_r - \text{SNR}_r/\text{SNR}_b + 1) (3\beta_h - 1) \\
+ \Re[(1 - 4\beta_h)\text{corr}(1) + \beta_h\text{corr}(2))] + 2 (1 - \Re[\text{corr}(1)]) \text{SNR}_r \}^{-1}.
\]

Fig. 5.3 shows \( R_{\text{mse}} \) versus \( \text{SNR}_r \) under different channel environments with \( \alpha_h = 0.1 \) and \( \beta_h = 0.05 \). Although it is not critically necessary in an actual design, setting \( \beta_h < \alpha_h \) is justified in principle based on the discussion about the choice of \( \alpha_h \) in Section 5.2 and from comparing (5.17) and (5.18) since we generally have \( M_2 \cdot \text{SNR}_b > M_1 \cdot \text{SNR}_r \), or, equivalently, \( \text{MSE}_r^{(3)} < \text{MSE}_r^{(1)} \) with \( \text{MSE}_r^{(3)} = 1/(M_2 \cdot \text{SNR}_b) \) and \( \text{MSE}_r^{(1)} = 1/(M_1 \cdot \text{SNR}) \). This is to prevent the occurrence of \( \text{MSE}_r^{(3)}/\text{MSE}_r^{(4)} < 0 \) (in dB) in the high SNR regime, which is otherwise possible as can be envisaged from Fig. 5.1 especially in the case of CM4. It can be seen from Fig. 5.3 that the proposed channel estimation scheme using 18 OFDM symbols \( (M_1 + M_2 = 6 \text{ per subband}) \) can achieve about 4.3 – 5.8 dB NMSE performance gain over the conventional LS solution which uses 6 OFDM symbols \( (M_1 = 2 \text{ per subband}) \). In comparison, the ML estimator based on 6 OFDM symbols \( (M_1 = 2 \text{ per subband}) \) has about \( 10 \log_{10}(N/N_m) \) dB gain over the conventional LS estimator with \( N_m \) being the assumed length of channel impulse response. This is because MSE of the ML estimator is linearly related to \( N_m \) when \( N_m \geq N_h \) (see Eq. (2.14)). Thus, ideally, we can select \( N_m = N_h \). Since \( N_h \) is usually not perfectly known, a common practice is to set \( N_m = N_g \). However, due to the fact that the maximum excess delays of some realizations of CM3 and CM4 are actually larger than those of CM1 and CM2 and also non-negligibly larger than \( N_g \), we assume that \( N_m = N_g = 37 \) for CM1/CM2 and \( N_m = 64 \) for CM3/CM4 in the ML estimator used here. This leads to NMSE performance gain of ML to be 5.4 dB and 3 dB for CM1/CM2 and CM3/CM4, respectively, over conventional LS. Therefore, we can conclude that, in terms of the NMSE performance, the proposed scheme significantly outperforms the conventional LS estimator and is comparable to the more sophisticated ML estimator. We shall demonstrate this in our numerical examples.
5.4 Performance Analysis

![Graph 1]

**Figure 5.3:** Analytical NMSE ratio, $R_{mse}$, under various channel environments with $\alpha_h = 0.1$ and $\beta_h = 0.05$.

We next analyze the computational complexity of the proposed scheme. Although the proposed CFR estimator requires more OFDM symbols than the ML estimator, its advantage of implementation ease is evident. Table 5.2 lists the numbers of real multiplications and additions required for performing channel estimation on a subband with various estimators. We exploit the finite alphabet property of QPSK for saving computations. For example, Step 1 in Table 5.1 can be computed with 2 real multiplications and 6 real additions for each $k \in \mathbb{Z}_1^{R/2} \cup \mathbb{Z}_N^{N-1-R/2}$. Further, given the smoothing parameters $\alpha_h$ and $\beta_h$, one may always find integers $\rho_1$ and $\rho_2$ such that $\rho_1\alpha_h = 1$ and $\rho_2\beta_h = 1$. Based on this arrangement, the related multiplications in Steps 2 and 4 in the proposed method can be implemented simply by logic shifters. The scaling factors $\rho_1$ and $\rho_2$ can be incorporated into the multiplication operations required in Steps 1 and Step 3, respectively. Steps 2 and 4 require $2R$ and $4R$ real multiplications and additions (per subband), respectively. We assume that the ML estimator can be
implemented by the cascade of two \( N \)-point DFT operations\(^2\), weighted by a \( N_m \times N_m \) matrix [90], where the DFT is split-radix based requiring \( N \log_2 N - 3N + 4 \) and \( 3N \log_2 N - 3N + 4 \) real multiplications and additions, respectively [140]. As a result, the LS estimator given by (5.2) totally requires \( 2R \) and \( 6R \) real multiplications and additions (per subband), respectively. As shown in Table 5.2, compared with the conventional LS estimator, while the proposed scheme requires 3 times more real multiplications and about 5 times more real additions, the ML estimator requires about 28 (when \( N_m = 37 \)) or 77 (when \( N_m = 64 \)) times more real multiplications and 15 (when \( N_m = 37 \)) or 31 (when \( N_m = 64 \)) times more real additions. The drastically increased computational complexity of the ML scheme makes it unacceptable in practice.

We want to point out that the complexity of the ML scheme given in Table 5.2 is based on the assumption that a matrix of size \( N_m \times N_m \) is prestored. This matrix contains the coefficients required for performing ML estimation, which depend on the parameter \( N_m \). Since \( N_m \) may vary with the actual channel environment and that one can not afford to prestore several different ML matrices with limited hardware resource, the \( N_m \)-dependent property of the ML matrix is commonly considered as a serious drawback of the ML-based channel estimation scheme for implementation of OFDM systems in practice. In fact, this problem becomes even worse in the case of OFDM-UWB. Suppose that only a single matrix which amounts to \( 2N_m^2 \) real data elements requires to be stored in memory. Assuming that each real data is 8-bit long and three logic gates are required for implementing one bit memory, about 66K (\( N_m = 37 \)) to 197K (\( N_m = 64 \)) logic gates may be required for storing one ML matrix. This is prohibitive for implementation in a handheld UWB device as this amounts to a significant portion of the logic gates available for implementing the whole digital portion of the UWB physical layer [24]. One way to circumvent this is by computing the ML matrix in real-time. However, this calculation

\(^2\)In fact, due to the rigid timing requirement in OFDM-UWB processing, the ML estimator may not be able to share the use of a DFT processor with the normal OFDM symbol processing. Thus, practical implementation of the ML estimator may require an additional DFT processor with dramatically increased use of hardware resource. In other words, our assumption with the ML estimator here may result in serious underestimation of its complexity in practice.
5.5 Numerical Results

Involves matrix inversion, which is of high complexity and hence practically infeasible.

In comparison, our proposed scheme requires no matrix storage and maintains similar order of computational complexity as the simple LS estimator, which makes it feasible and attractive for practical implementation.

Table 5.2: Required computational complexity for CFR estimation per subband in a frame.

<table>
<thead>
<tr>
<th>Scheme</th>
<th>Real Multiplications</th>
<th>Real Additions</th>
</tr>
</thead>
<tbody>
<tr>
<td>LS (2 Symbols per Suband)</td>
<td>$2R (= 224)$</td>
<td>$6R (= 672)$</td>
</tr>
<tr>
<td>Proposed (6 Symbols per Subband)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Step 1</td>
<td>$2R (= 224)$</td>
<td>$6R (= 672)$</td>
</tr>
<tr>
<td>Step 2</td>
<td>$2R$</td>
<td>$4R$</td>
</tr>
<tr>
<td>Step 3a</td>
<td>$2R$</td>
<td>$20R - 6P$</td>
</tr>
<tr>
<td>Step 4</td>
<td>$2R$</td>
<td>$4R$</td>
</tr>
<tr>
<td>Step 5</td>
<td>0</td>
<td>$2R$</td>
</tr>
<tr>
<td>Total</td>
<td>$8R (= 896)$</td>
<td>$36R - 6P (= 3960)$</td>
</tr>
<tr>
<td>ML (2 Symbols per Suband)</td>
<td>$2(N \log_2 N - 3N + 4) + 4N_m^2$</td>
<td>$2(3N \log_2 N - 3N + 4) + 4N_m^2 - 2N_m + 6R$</td>
</tr>
<tr>
<td></td>
<td>$6508, \text{ if } N_m = 37;$</td>
<td>$10690, \text{ if } N_m = 37;$</td>
</tr>
<tr>
<td></td>
<td>$17416, \text{ if } N_m = 64.$</td>
<td>$21544, \text{ if } N_m = 64.$</td>
</tr>
</tbody>
</table>

*The operations required in (5.9) are excluded as they can be attributed to the equalization process for header symbols.

5.5 Numerical Results

In our simulations, we consider an OFDM-UWB system with the data transmission rate of 53.3 Mbps. The selection of the lowest data rate specified in [12] as example here is to illustrate the effectiveness of the proposed techniques under very low SNR conditions. We assume that TFC = 1 and the frame payload is 1024-byte long with perfect timing and frequency synchronization. We use the UWB channel models CM1, CM2, CM3 and CM4, and each of them has 100 realizations [24]. Following the convention of
OFDM-UWB system design, the worst 10 realizations of each channel model are ignored for all the NMSE and frame error rate (FER) related performance evaluation [24, 136]. This is due to the fact that the maximum excess delays of some channel realizations, in particular those of CM3 and CM4, are non-negligibly longer than $N_g$, as mentioned in Section 5.4. Also, we set $\alpha_h = 0.1$ and $\beta_h = 0.05$ in the simulations.

In the simulation, we first evaluate the NMSE performance of the proposed channel estimation scheme. Fig. 5.4 shows the NMSE performance versus the SNR for different channel estimation schemes. As expected, the proposed channel estimation using 18 OFDM symbols ($M_1 + M_2 = 6$ per sub-band) performs much better than the conventional LS estimator using 6 OFDM symbols ($M_1 = 2$ per sub-band). Observe that the simulated NMSE performance of the proposed scheme closely follows the analytical result under CM1 or CM2. When the channel is CM3 or CM4, on the other hand, a deviation between the simulation and analytical results is observed. This is because the multipath delay spreads of most of the CM3 and CM4 realizations are quite large that certain amount of ISI exists even after the worst 10% channel realizations are dropped. Nevertheless, it should be pointed out that, even without using any specific remedy for the residual ISI, the resulting system performance is still acceptable as we shall see in the FER performance evaluation of the overall system. For comparison, Fig. 5.4 also shows the performance of the ML estimator with $N_m = 37$ and $N_m = 64$ for CM1/CM2 and CM3/CM4, respectively. Clearly, the proposed scheme using 18 OFDM symbols and the ML estimator using 6 OFDM symbols ($M_1 = 2$ per subband) have similar NMSE performance for\(^3\) CM1/CM2, and the proposed scheme outperforms the ML for CM3/CM4.

The FER performance for the four different types of UWB channels are shown in Fig. 5.5. Observe that the proposed channel estimator performs slightly worse than ML for CM1/CM2 and slightly better than ML for CM3/CM4, while performing significantly better than LS under all channel conditions. It is interesting to note that the NMSE

---

\(^3\)Strictly speaking, in these cases, with CM1/CM2, compared with the ML estimator, the NMSE performance of the proposed scheme is slightly worse in the very low SNR regime and is slightly better otherwise.
Figure 5.4: NMSE performance comparison for various channel estimation methods under (a) CM1, (b) CM2, (c) CM3, and (d) CM4 (Sim, Ana, and PD are abbreviations for simulation, analytical and proposed, respectively.).
5.5 Numerical Results

Figure 5.5: FER performance comparison for various channel estimation methods under (a) CM1 & CM2, (b) CM3 & CM4.

The performance of the proposed estimator and the ML estimator observed in Fig. 5.4 do not seem to correlate well with the FER performance in Fig. 5.5. In other words, even though the proposed estimator and the ML estimator exhibit a significant difference in their NMSE performance for CM3/CM4, the difference in their FER performance is actually
5.6 Concluding Remarks

In this chapter, we have proposed a novel channel estimation scheme for the multi-band OFDM-based UWB systems. The solution has been verified to be effective and efficient in terms of performance and implementation complexity. The channel estimator is LS-based, but enhanced with a multi-stage procedure using a simplified, CFR-weighted decision-directed process as well as frequency-domain smoothing. Both analytical and numerical results show that the proposed scheme achieves performance similar to that of the more sophisticated but practically infeasible ML estimator, and it outperforms the conventional LS estimator with about 4.3 – 5.8 dB gain in terms of NMSE performance and about 1 dB gain in terms of FER performance under various highly noisy multipath channel environments.

---

4 As described in Section 1.3.1, the ML estimator links the finite delay spread of the channel to the frequency-domain channel correlation and thus achieves similar noise reduction capability as that of the LMMSE estimator with lower complexity. For this reason, the LMMSE estimator has not been used here for performance comparison.
Chapter 6

Phase Error Suppression for
Multi-Band OFDM-UWB Systems

In this chapter, we propose an efficient phase error suppression scheme for multi-band OFDM based UWB communication systems. As shown in Fig. 6.1, the proposed scheme consists of a clock recovery loop and a common phase error (CPE) tracking loop. The clock recovery loop performs estimation of sampling frequency offset (SFO) and its two-dimensional (time and frequency) compensation, while the CPE tracking loop estimates and corrects the phase errors caused by residual carrier frequency offset (CFO), residual SFO and phase noise (PHN). The SFO and CPE estimators employ pilot-tone based and channel frequency response weighted low-complexity approaches, each of which uses a robust error reduction scheme without using angle calculation and division. Analytical results and numerical examples show the effectiveness of the proposed scheme in different multi-path fading scenarios and SNR regimes.

6.1 SFO Estimation and Compensation

In this section, we focus on the development of a novel SFO estimation and compensation scheme which is presented in the form of a clock recovery loop, as described below.
6.1 SFO Estimation and Compensation

6.1.1 Basic Algorithm for SFO Estimation

Recall that \( \hat{h}_r = [\hat{h}_r(0), \hat{h}_r(1), \ldots, \hat{h}_r(N - 1)] \), \( r \in \mathbb{Z}_1^3 \) is the obtained estimate of channel frequency response (CFR) \( h_r \) on the \( r \)th sub-band\(^1\) using the channel estimation scheme proposed in Chapter 5. Suppose that the \( m \)th group of OFDM symbols are being processed. Our task is to estimate the accumulated normalized SFO (ANSFO), \( \eta_m^{(i)} \), which is related to the \((6m + i)\)th OFDM symbol. The SFO estimation is based on \( P \) pilot subcarriers embedded in each symbol and the CFR estimate, \( \hat{h}_r \). Suppose that an amount \( \hat{\eta}_m^{(i)} \) of ANSFO has been already compensated in the received \((6m + i)\)th OFDM symbol, resulting in \( \bar{y}_m^{(i)} = [\bar{y}_m^{(i)}(0), \bar{y}_m^{(i)}(1), \ldots, \bar{y}_m^{(i)}(N - 1)] \), before we use it to estimate the residual SFO denoted by \( \Delta \eta_m^{(i)} = \eta_m^{(i)} - \hat{\eta}_m^{(i)} \). Using (4.12), we obtain

\[
e^{j[\hat{\eta}_m^{(i)} + \Delta \psi_m^{(i)}(k)]} \approx \bar{y}_m^{(i)}(k)/[s_m^{(i)}(k)\hat{h}_r(k)], \quad k \in \{p(l)\}_{l=0}^{P-1} \tag{6.1}
\]

where

\[
\Delta \psi_m^{(i)}(k) = \begin{cases} 
2\pi k / N \cdot \Delta \eta_m^{(i)}, & k \in \{p(l)\}_{l=0}^{P/2-1} \\
2\pi (k - N) / N \cdot \Delta \eta_m^{(i)}, & k \in \{p(l)\}_{l=P/2}^{P-1}.
\end{cases} \tag{6.2}
\]

From the definition of pilot tones in Section 4.1, we further define

\[
J(l_1, l_2) := p(l_2) - N - p(l_1) = 10(l_2 - P - l_1), \quad l_1 \in \mathbb{Z}_0^{P-1} \text{ and } l_2 \in \mathbb{Z}_{P/2}^{P/2+l_1} \tag{6.3}
\]

\(^1\)Since the phase errors manifest themselves similarly in different sub-bands, the phase error mitigation scheme used in all sub-bands is same except that the CFR estimates used are different.
Noting that \(|s_m^{(i)}(p(l_2))|^2 = 1\) for any \(l_2 \in \mathbb{Z}^{P/2}_{P/2} \), from (6.1), (6.2), (4.13) and (4.14), we can derive
\[
e^{j2\pi J(l_1,l_2)\Delta\eta_m^{(i)}(l_1,l_2)/N} = \frac{e^{j\Delta\psi_m^{(i)}(p(l_2))}}{e^{j\Delta\psi_m^{(i)}(p(l_1))}} = \frac{a_m^{(i)}(l_1,l_2)}{d_m^{(i)}(l_1,l_2)}, \quad l_1 \in \mathbb{Z}^{P/2-1}_0 \text{ and } l_2 \in \mathbb{Z}^{P/2}_{P/2} \]  
(6.4)
where
\[
a_m^{(i)}(l_1,l_2) = \left[\hat{y}_m^{(i)}(p(l_1))\hat{r}_m^{(i)}(p(l_2))\hat{h}_r(p(l_1))\hat{h}_r(p(l_2))\right]^* s_m^{(i)}(p(l_1))s_m^{(i)}(p(l_2))^* \]  
(6.5)
and
\[
d_m^{(i)}(l_1,l_2) = \left|\hat{y}_m^{(i)}(p(l_1))\hat{h}_r(p(l_2))\right|^2. \]  
(6.6)

Here, \(\Delta\eta_m^{(i)}(l_1,l_2)\) emphasizes the relation of \(\Delta\eta_m^{(i)}\) to the pilot-tone pair \((l_1, l_2)\). Also, it can be seen that, in (6.3), we have chosen to use only \(B = \sum_{q=1}^{P/2} q = 21\) pilot-tone pairs which is less than the total number of possible combinations of any two different pilot tones. The reason behind this is that the two pilot subcarriers in any selected pair need to be sufficiently apart, i.e., \(|J(l_1, l_2)|\) is not small, such that in the presence of noise the result given by the RHS of the second equality in (6.4) is not channel noise dominant. In (6.3), \(|J(l_1, l_2)| \geq 5P = 60\) is selected.

From (6.4), we can obtain \(\Delta\eta_m^{(i)}(l_1,l_2)\) as
\[
\Delta\eta_m^{(i)}(l_1,l_2) = \frac{N}{2\pi J(l_1,l_2)} \arg \left\{ \frac{a_m^{(i)}(l_1,l_2)}{d_m^{(i)}(l_1,l_2)} \right\}, \quad l_1 \in \mathbb{Z}^{P/2-1}_0 \text{ and } l_2 \in \mathbb{Z}^{P/2}_{P/2}. \]  
(6.7)

By averaging, we obtain our basic algorithm for estimating the residual SFO, \(\Delta\hat{\eta}_m^{(i)}\), given by
\[
\Delta\hat{\eta}_m^{(i)} = \frac{1}{B} \sum_{l_1=0}^{P/2-1} \sum_{l_2=P/2}^{P/2+l_1} \Delta\eta_m^{(i)}(l_1,l_2)
= \frac{N}{2\pi B} \sum_{l_1=0}^{P/2-1} \sum_{l_2=P/2}^{P/2+l_1} \frac{1}{J(l_1,l_2)} \arg \left\{ \frac{a_m^{(i)}(l_1,l_2)}{d_m^{(i)}(l_1,l_2)} \right\}. \]  
(6.8)

This estimation algorithm has two shortcomings, which weaken its feasibility for practical implementation. First, it includes a phase argument function which is usually annoying
6.1 SFO Estimation and Compensation

and should be avoided in actual implementation, if possible. Secondly, in (6.8), the estimates obtained from all pilot-tone pairs are treated equally. Obviously, in a frequency selective fading environment, this is not optimum since it has ignored the fact that different pilot tones may experience different degrees of fading and thus have different carrier-to-noise ratios (CNR’s). Fortunately, this problem can be solved by weighting each individual estimation in an efficient manner, as introduced next.

6.1.2 Weighted SFO Estimation

Being aware of the fact that \( |2 \pi J(l_1, l_2) \Delta \eta_{m}^{(i)}(l_1, l_2)/N| \ll 1 \) and applying \( \sin(\varphi) \approx \varphi \) (when \( |\varphi| \ll 1 \)) to (6.4), we have

\[
\Delta \eta_{m}^{(i)}(l_1, l_2) \approx \Delta \hat{\eta}_{m}^{(i)}(l_1, l_2)
= \frac{N}{2 \pi J(l_1, l_2)} \Re \left[ a_{m}^{(i)}(l_1, l_2) \right], \quad l_1 \in \mathbb{Z}_{0}^{P/2-1} \text{ and } l_2 \in \mathbb{Z}_{P/2}^{P/2+l_1}.
\]

(6.9)

By virtue of this approximation, no actual trigonometric operation for angle calculation is required. Similar to that in (6.8), we use the full set of pilot-tone pairs to obtain the estimate of residual SFO, \( \Delta \hat{\eta}_{m}^{(i)} \), as

\[
\Delta \hat{\eta}_{m}^{(i)} = \sum_{l_1=0}^{P/2-1} \sum_{l_2=P/2}^{P/2+l_1} \Delta \hat{\eta}_{m}^{(i)}(l_1, l_2)
= \frac{N}{2 \pi B} \sum_{l_1=0}^{P/2-1} \sum_{l_2=P/2}^{P/2+l_1} b_{m}^{(i)}(l_1, l_2)
= \frac{N}{2 \pi} \sum_{l_1=0}^{P/2-1} \sum_{l_2=P/2}^{P/2+l_1} b_{m}^{(i)}(l_1, l_2)
\]

(6.10)

where

\[
b_{m}^{(i)}(l_1, l_2) := \Re \left[ a_{m}^{(i)}(l_1, l_2) \right]/J(l_1, l_2).
\]

(6.11)

The last equality in (6.10) holds in the absence of noise, as we have used the following proportion property:

\[
\frac{x(l)}{y(l)} = \frac{\sum_{l=0}^{B-1} x(l)}{\sum_{l=0}^{B-1} y(l)}, \quad \text{if } \frac{x(0)}{y(0)} = \frac{x(1)}{y(1)} = \cdots = \frac{x(B-1)}{y(B-1)}.
\]

In this way, division operations can be avoided in obtaining the final estimate, as shall be clear from our discussion in Section 6.1.3.
In fact, the use of the proportion property here has an important implication in the presence of noise. Observe from (6.1) that we have \(|\hat{y}^{(i)}_m(p(l_1))| = |\hat{h}_r(p(l_1))|\) since \(|s^{(i)}_m(p(l_1))| = 1\). Thus, from (6.6), we can obtain

\[
d^{(i)}_m(l_1, l_2) = |\hat{h}_r(p(l_1))|^2 |\hat{h}_r(p(l_2))|^2.
\]

Let

\[
D_r = \frac{1}{B} \sum_{q_1=0}^{P/2-1} \sum_{q_2=P/2}^{P/2+q_1} |\hat{h}_r(p(q_1))|^2 |\hat{h}_r(p(q_2))|^2.
\]

The last equality in (6.10) can be rewritten as

\[
\Delta \hat{\eta}^{(i)}_m = \frac{N}{2\pi B} \sum_{l_1=0}^{P/2-1} \sum_{l_2=P/2}^{P/2+l_1} \left( \frac{b^{(i)}_m(l_1, l_2)}{d^{(i)}_m(l_1, l_2)} \right) \underbrace{D_r}_{g(l_1, l_2)}. \tag{6.12}
\]

Thus, it can be seen that the use of the proportion property in (6.10) is actually equivalent to weighting the estimate of residual SFO obtained with the pilot-tone pair \((l_1, l_2)\) by a weighting factor, \(g(l_2, l_1)\), which is the normalized product of CFR magnitudes (squared) on that pilot-tone pair. In general, the larger \(g(l_2, l_1)\) is, the more reliable will be the estimate of residual SFO on the corresponding pilot-tone pair. Hence, the last equality in (6.10) performs a CFR-weighted combination of the estimates of residual SFO using all \(B\) pilot-tone pairs, which, of course, can reduce the estimation error caused by channel noise.

### 6.1.3 Combined SFO Estimation

Albeit effective in compensation of the noise-impairment effect on the estimate of residual SFO, in the presence of heavy noise, i.e., under very low SNR conditions, the estimate in (6.10) (last equality) is prone to causing errors which, in turn, may still make the SFO tracking unstable. This is mainly due to the limited number of pilot pairs available for use in (6.10). Because of this, we further devise a simple yet effective error suppression technique as described below.

We define from (6.10)

\[
\zeta^{(i)}_m = \sum_{l_1=0}^{P/2-1} \sum_{l_2=P/2}^{P/2+l_1} b^{(i)}_m(l_1, l_2), \quad \iota^{(i)}_m = \sum_{l_1=0}^{P/2-1} \sum_{l_2=P/2}^{P/2+l_1} d^{(i)}_m(l_1, l_2), \quad i \in \mathbb{Z}_5. \tag{6.13}
\]
Note that $i_m > 0$. We then find two values, $N^p_m$, the number of $\varsigma^{(i)}_m$’s (among 6) satisfying $\varsigma^{(i)}_m > 0$, and, $N^n_m$, the number of $\varsigma^{(i)}_m$’s satisfying $\varsigma^{(i)}_m < 0$. Clearly, we have $0 \leq |N^p_m - N^n_m| \leq 6$. Moreover, in the presence of noise, especially under low-SNR conditions, a large $|N^p_m - N^n_m|$ implies the existence of a relatively large residual SFO, and vice versa. Let $\mu > 0$ be a system design parameter defined as the micro-shift of SFO which can be fixed at a small value, and $\varrho = [\varrho(0), \varrho(1), \cdots, \varrho(6)]$ a predefined integer vector satisfying $\varrho(l + 1) \geq \varrho(l)$ and $\varrho(l) \in \mathbb{Z}_6$. Then we can replace the estimate of residual SFO, $\Delta \hat{\eta}^{(i)}_m$, in (6.10) with

$$\Delta \hat{\eta}_m = \text{sgn}(N^p_m - N^n_m) \varrho(|N^p_m - N^n_m|) \mu$$

which combines the six SFO estimates obtained independently by applying (6.10) (last equality) to each individual OFDM symbol of the $m$th group. The combined estimation is robust and noise resilient in the sense that it does not require using the exact value of $\Delta \hat{\eta}^{(i)}_m$ given by (6.10), but involves a unique approximation and average process for combining a group of estimates. The combination requires two parameters, $\mu$ and $\varrho$, whose setting relates to the maximum permissible SFO, maximum duration of a frame, and, in particular, tolerance to residual SFO, i.e., the maximum amount of SFO which has negligible effect on system performance. The first two factors are available from the OFDM-UWB specification [12] while the last factor can be evaluated based on a trial and error method via simulations. As an example, we have used $\mu = 1/32$ and $\varrho = [0, 1, 2, 2, 2, 3, 3]$ in our design which yield good SFO tracking performance under all system conditions as shall be demonstrated through our numerical simulations. It should be emphasized that the actual choice of $\mu$ and $\varrho$ may not necessarily follow this example exactly since the setting is found to be not very sensitive to the system performance and thus can be easily adjusted to cater for different practical designs.

Finally, we update the estimation of the ANSFO as

$$\hat{\eta}^{(i)}_{m+1} = \hat{\eta}^{(5)}_m + (i + 1) \Delta \hat{\eta}_m + \hat{\eta}^{(i)}_m / m$$

which shall be used to compensate the $i$th OFDM symbol in the $(m + 1)$th group. The last term on the RHS of (6.15) is used to bridge the time gap between estimation and
compensation due to the processing delay for one group of OFDM symbols.

In addition to its robustness in mitigating the estimation error even in the very low SNR regime, which will be validated via simulations in Section 6.3, the combined SFO estimation scheme presented above is of low complexity. As a matter of fact, the combination formulated in (6.14) has greatly simplified the SFO estimation since it is now unnecessary to compute \( \ell_m^{(i)} \) in (6.13). The major computational burden in SFO estimation lies in computing \( \varsigma_m^{(i)} \), which, as can be found from (6.5), (6.11) and (6.13), involves \( 10B = 210 \) real multiplications, \( 6B − 1 = 125 \) real additions and \( B = 21 \) real divisions per OFDM symbol. Since \( J(l_1, l_2) \)'s are predefined scalars with fixed values, the divisions involved in (6.11) can be easily converted to multiplications in actual implementation.

Further reduction in computational complexity of the proposed SFO estimator is possible when we reduce the number of pilot-tone pairs used. A special scenario occurs when we use \( \bar{B} = P/2 = 6 \) equal-distant pilot-tone pairs. In this case, (6.3) is simplified to

\[
J(l_1, l_2) = 10(l_2 - P - l_1) = -5P, \quad l_1 \in \mathbb{Z}_{P/2-1} \quad \text{and} \quad l_2 = P/2 + l_1. \tag{6.16}
\]

Modifying (6.5), (6.11) and (6.13) accordingly yields

\[
\varsigma_m^{(i)} = \sum_{l=0}^{P/2-1} \Im \left[ \hat{a}_m^{(i)}(l) \right], \quad i \in \mathbb{Z}_5^0 \tag{6.17}
\]

where

\[
\hat{a}_m^{(i)}(l) = \left[ \hat{y}_m^{(i)}(p(l)) \right]^* \hat{y}_m^{(i)}(p(l + P/2)) \hat{h}_r(p(l)) \times \left[ \hat{h}_r(p(l + P/2)) \right]^* \varsigma_m^{(i)}(p(l)) \left[ s_m^{(i)}(p(l + P/2)) \right]^* . \tag{6.18}
\]

The operations required for computing \( \varsigma_m^{(i)} \) are finally reduced to \( 10\bar{B} = 60 \) real multiplications and \( 6\bar{B} − 1 = 35 \) real additions per OFDM symbol. No division is involved in this case. Such a low computational load makes the proposed SFO estimator suitable for achieving low-complexity and low-power implementation for high-speed applications. This additional complexity reduction is obtained at the expense of slightly degraded system performance as we will see in the simulations in Section 6.3.
6.1.4 Two-Dimensional SFO Compensation

With the above derived ANSFO estimate for the $(6m + i)^{th}$ OFDM symbol, $\hat{\eta}_m^{(i)}$, the SFO compensation becomes straightforward. We next propose a new approach that performs SFO compensation in both time and frequency domains. Compared with the conventional approach that uses a time-domain interpolator [52, 64, 65, 95, 96], this approach is more suitable in high-speed processing environments. Denote by $I(\hat{\eta}_m^{(i)})$ and $F(\hat{\eta}_m^{(i)})$ the integral and fractional portions of $\hat{\eta}_m^{(i)}$, respectively, and define

$$
\varphi_m^{(i)}(k) = \begin{cases} 
2\pi k/N \cdot F(\hat{\eta}_m^{(i)}), & k \in \mathbb{Z}_{N/2}^{N/2-1} \\
2\pi (k - N)/N \cdot F(\hat{\eta}_m^{(i)}), & k \in \mathbb{Z}_{N/2}^{N-1}.
\end{cases}
$$

(6.19)

With reference to (4.12), (4.15) and Fig. 6.1, the SFO compensation is accomplished by first shifting the sample timing (forward or backward depending on the sign of $I(\hat{\eta}_m^{(i)})$ — forward if $I(\hat{\eta}_m^{(i)}) < 0$; backward if $I(\hat{\eta}_m^{(i)}) > 0$) by $|I(\hat{\eta}_m^{(i)})|$ sampling intervals (integer part) in time-domain and then correcting the remaining SFO (fractional part) in frequency-domain as

$$
\hat{y}_m^{(i)}(k) = \bar{y}_m^{(i)}(k)e^{-j\varphi_m^{(i)}(k)}, \quad k \in \mathbb{Z}_N^{N-1}.
$$

(6.20)

Here, $\bar{y}_m^{(i)}(k)$ is used in place of $y_m^{(i)}(k)$ to reflect the above mentioned sample timing adjustment (integer part). With the compensated $\hat{y}_m^{(i)}(k)$ obtained in (6.20), the proposed clock recovery loop is closed and the above SFO estimation and compensation procedure shall be repeated for processing the subsequent OFDM symbols.

We shall now use a numerical example to demonstrate the effectiveness of the proposed clock recovery loop. Fig. 6.2 shows the ANSFO recovery trajectory in a multipath channel environment (CM1) with $\text{SNR}_r = 0$ dB. The actual SFO introduced is 40 parts per million (ppm) and we use $\bar{B} = 6$ pilot-tone pairs for SFO estimation. Clearly, the proposed clock recovery loop can closely track both the integral and fractional portions of the actual SFO.
6.2 CPE Estimation and Correction

Recall from the discussion in Section 6.1 that, after the SFO compensation, the residual SFO remaining in each OFDM symbol is negligible. Using this, from (4.12), we obtain

$$ e^{j\hat{\theta}_{m}^{(i)}(t)} = \frac{\tilde{y}_{m}^{(i)}(p(l))}{s_{m}^{(i)}(p(l))} \frac{c_{m}^{(i)}(l)}{|\hat{h}_{r}(p(l))|^{2}}, \quad l \in \mathbb{Z}_{0}^{P-1} $$  \hspace{1cm} (6.21)

where $\hat{\theta}_{m}^{(i)}(l)$ is the CPE estimate on pilot tone $l$, and

$$ c_{m}^{(i)}(l) = \tilde{y}_{m}^{(i)}(p(l)) [s_{m}^{(i)}(p(l))]^{*} [\hat{h}_{r}(p(l))]^{*}. $$  \hspace{1cm} (6.22)

### 6.2.1 Weighted CPE Estimation

By using the full set of pilot tones of the $i$th OFDM symbol in the $m$th group, from (6.22), the average CPE estimate, denoted as $e^{j\tilde{\theta}_{m}^{(i)}}$, can be obtained as

$$ e^{j\tilde{\theta}_{m}^{(i)}} = \frac{1}{P} \sum_{l=0}^{P-1} e^{j\hat{\theta}_{m}^{(i)}(t)} = \frac{1}{P} \sum_{l=0}^{P-1} \frac{c_{m}^{(i)}(l)}{|\hat{h}_{r}(p(l))|^{2}} \approx \frac{\sum_{l=0}^{P-1} c_{m}^{(i)}(l)}{G_{r}} $$  \hspace{1cm} (6.23)
6.2 CPE Estimation and Correction

where \( G_r = \sum_{l=0}^{P-1} |\hat{h}_r(p(l))|^2 \). As in Section 6.1.2, the last approximation in (6.23) holds due to the proportion property in the absence of noise. The use of the proportion property in this case also has an important implication in the presence of noise. In fact, the last approximation in (6.23) can be rewritten as

\[
e^{j\hat{\theta}_m^{(i)}} \approx \frac{1}{P} \sum_{l=0}^{P-1} \left( \frac{c_m^{(i)}(l)}{|\hat{h}_r(p(l))|^2} \cdot \frac{|\hat{h}_r(p(l))|^2}{\frac{1}{P} \sum_{q=0}^{P-1} |\hat{h}_r(p(q))|^2} \right).
\]

(6.24)

It can be seen that the use of the proportion property in (6.23) is equivalent to weighting the CPE estimate on pilot tone \( l \) with a factor, \( z(l) \), which is the normalized CFR magnitude (squared) on that pilot tone. The larger \( z(l) \) is, the more reliable will be the CPE estimate on the respective pilot tone. Hence, the last approximation in (6.23) performs a CFR-weighted combination of the CPE estimates on all pilot tones, which can reduce the estimation error caused by channel noise. Interestingly, this estimator coincides with the near-ML estimator described in [104], and thus it is expected to perform close to that of the more sophisticated, exact ML solution in the low or moderate SNR regime. In particular, we observe from (6.21)–(6.23) that \( e^{j\hat{\theta}_m^{(i)}} \) can be expressed as

\[
e^{j\hat{\theta}_m^{(i)}} = e^{j\theta_m^{(i)}} + w_m^{(i)}
\]

(6.25)

where the estimation error, \( w_m^{(i)} \), can be modeled as a zero-mean complex Gaussian random variable which is uncorrelated with \( e^{j\theta_m^{(i)}} \). The approximation of the variance of estimation error on the \( r \)th sub-band is given by [104, Eq. (17)]

\[
\sigma^2_{w_r}(r) \approx \frac{\sigma_r^2}{\sum_{q=0}^{P-1} |\hat{h}_r(p(q))|^2}, \quad r \in \mathbb{Z}_3^1
\]

(6.26)

where \( \sigma_r^2 \) is the variance of channel noise as described in Section 4.3.

6.2.2 Smoothed CPE Estimation

Clearly, the CPE estimator described in Section 6.2.1 is intra-OFDM-symbol based because it uses the pilot tones of each individual OFDM symbol independently. Due to the limited number of pilot tones available in each OFDM symbol, the estimate based
on (6.23) may not be able to achieve sufficient accuracy, especially in the low SNR regime. It has been shown in [102] that a decision feedback process can be used for further enhancing the CPE estimation by using data tones. The solution has proved to be fairly effective for applications with low or moderate processing speed (e.g., OFDM-based WLANs), but the extra computational load incurred may prevent its use in UWB devices that require very high-speed processing and low power consumption. To further improve the CPE estimation accuracy while maintaining the virtues of simplicity and ease of implementation, we propose to use the following inter-OFDM-symbol smoothing scheme

\[ e^{j\hat{\theta}_m^{(i)}} = (1 - \alpha_c)e^{j\tilde{\theta}_m^{(i)}} + \alpha_c\nu e^{j\hat{\theta}_{m_1}^{(i)}} \]  

(6.27)

where \( m_1 = m \) and \( i_1 = i - 1 \), if \( i \in \mathbb{Z}_1^5 \), and, \( m_1 = m - 1 \) and \( i_1 = 5 \), if \( i = 0 \). In (6.27), the inter-OFDM-symbol smoothing is performed by a 1st-order low-pass filter with the smoothing factor \( \alpha_c \). Specifically, \( \nu \) is used to compensate the CPE difference between two consecutive OFDM symbols, which is explained below.

In the absence of PHN or in the case that the contribution from PHN to CPE is negligible, i.e., the term \( \phi_m^{(i)} \) in (4.12) can be ignored, we have \( \theta_m^{(i)} = \psi_m^{(i)} \). From (4.14), one may find that \( \nu \) can be ideally set to

\[ \nu = e^{j2\pi\epsilon(1+\delta)(N+N_g)/N}. \]  

(6.28)

With this setting, the CPE can be perfectly tracked. However, due to the unavailability of \( \epsilon \) (the effect of \( \delta \) is negligible in this case since \( \epsilon \delta \) is close to 0) and the difficulty in obtaining an accurate estimate of \( \epsilon \) under low SNR conditions, if not impossible, an approximate value of \( \nu \) is expected to be used in practice. Thanks to the fact that \( \epsilon \) is small, the value of \( \nu \) can be approximated in two ways. We first simply assume that \( \nu \approx 1 \). Thus, we can rewrite (6.27) as

\[ e^{j\hat{\theta}_m^{(i)}} = (1 - \alpha_c)e^{j\tilde{\theta}_m^{(i)}} + \alpha_c\nu e^{j\hat{\theta}_{m_1}^{(i)}} \]  

(6.29)

which is still a 1st-order low-pass filter. Secondly, we assume that \( \Re(\nu) \approx 1 \). Based on this approximation, it is easy to obtain

\[ e^{j\theta_m^{(i)}} = \nu e^{j\theta_{m_1}^{(i)}} \approx 2e^{j\theta_{m_1}^{(i)}} - e^{j\theta_{m_2}^{(i)}} \]  

(6.30)
where $m_2 = m$ and $i_2 = i - 2$, if $i \in \mathbb{Z}_2$, and, $m_2 = m - 1$ and $i_2 = i + 4$, if $i \in \mathbb{Z}_1$. In this case, (6.27) can be rewritten as

$$e^{j\hat{\theta}_m^{(i)}} = (1 - \alpha_c)e^{j\tilde{\theta}_m^{(i)}} + \alpha_c[2e^{j\hat{\theta}_m^{(i_1)}} - e^{j\hat{\theta}_m^{(i_2)}}]$$

which forms a 2nd-order low-pass filter. Combining (6.29) and (6.31) yields a generalized yet approximate form of (6.27) as follows

$$f(e^{j\hat{\theta}_m^{(i)}}) = (1 - \alpha_c)e^{j\tilde{\theta}_m^{(i)}} + \alpha_1 f(e^{j\hat{\theta}_m^{(i_1)}}) + \alpha_2 f(e^{j\hat{\theta}_m^{(i_2)}})$$

where $\alpha_1 = \alpha_c$ and $\alpha_2 = 0$, if 1st-order low-pass filtering is used, and $\alpha_1 = 2\alpha_c$ and $\alpha_2 = -\alpha_c$, if 2nd-order low-pass filtering is used. Note that, hereafter, we use $f(e^{j\hat{\theta}_m^{(i)}})$ instead of $e^{j\hat{\theta}_m^{(i)}}$ to denote the CPE estimate to highlight that $|f(e^{j\hat{\theta}_m^{(i)}})|$ may deviate slightly from unity.

We next show that (6.32) is effective even when PHN is also present in CPE. We consider the deviation between $E\{f(e^{j\hat{\theta}_m^{(i)}})\}$ and $E\{e^{j\theta_m^{(i)}}\}$. Let

$$\sigma_E = \left| \frac{E\{f(e^{j\hat{\theta}_m^{(i)}})\} - E\{e^{j\theta_m^{(i)}}\}}{E\{e^{j\theta_m^{(i)}}\}} \right|$$

be the normalized deviation of CPE estimate. Following the derivation of $E\{f(e^{j\hat{\theta}_m^{(i)}})\}$ in Appendix C, we have

$$\sigma_E = \left| \frac{1 - \alpha_c}{1 - \alpha_1e^{\pi(N+N_g)|\beta T_s-j2c(1+\delta)/N|} - \alpha_2e^{2\pi(N+N_g)|\beta T_s-j2c(1+\delta)/N|} - 1} \right|,$$

for $n = 6m + i \gg 1$. Fig. 6.3 shows the variation of $\sigma_E$ versus $\alpha_c$, which is evaluated at various values of $\epsilon$ (0.005, 0.01 and 0.015 represent the small, medium and large values of $\epsilon$, respectively). It can be seen that the scheme with 2nd-order low-pass filtering performs better than the scheme with 1st-order filtering in terms of the resulting deviation of CPE estimate. Nevertheless, it is important to note that the deviation of CPE estimate becomes almost negligible (within 5%) in both cases of 1st-order and 2nd-order low-pass filtering when $\alpha_c \leq 0.3$.

### 6.2.3 Analysis of MSE Reduction Performance

Effectiveness of the smoothed CPE estimation can be evaluated by analyzing the MSE-reduction ratio resulting from the introduction of the low-pass filtering approach,
6.2 CPE Estimation and Correction

Figure 6.3: Normalized deviation of CPE estimation, $\sigma_E$, when two types of smoothing filters are used at various values of $\epsilon$ with $\beta = 6$ KHz.

which is given by

$$\eta_{mse} = 1 - \frac{E\{|f(e^{i\hat{\theta}_n}) - e^{i\theta_n}|^2\}}{E\{|e^{i\theta_n} - e^{i\theta_n}|^2\}} = \left(1 - \frac{\text{MSE}_{\text{cpe}}}{\sigma_w^2}\right) \times 100\%$$  \hspace{1cm} (6.34)

where $\sigma_w^2 = E\{|e^{i\hat{\theta}_n} - e^{i\theta_n}|^2\} = \sigma_w^2(r)$ with the assumption that the error variances $\sigma_w^2(r)$ are same in all three sub-bands, and $\text{MSE}_{\text{cpe}} = E\{|f(e^{i\hat{\theta}_n}) - e^{i\theta_n}|^2\}$ has a closed-form expression for $n = 6m + i \gg 1$, as shown in Appendix C. Thus, we have

$$\eta_{mse} = 1 - \frac{A_0}{\alpha_0 \sigma_w^2} \left(2(1-\alpha_c)\left[\Re(\Omega_0) - (1-\alpha_2)(1-\alpha_c)+\alpha_0\right] + \frac{2 - e^{\pi N\beta T_s} - e^{-\pi N\beta T_s}}{N^2(2 - e^{\pi \beta T_s} - e^{-\pi \beta T_s})}\right)$$

(6.35)

where

$$\alpha_0 = 1 - \alpha_2 - \alpha_1^2 - \alpha_2^2 - \alpha_2 \alpha_1^2 + \alpha_2^3$$

$$A_0 = \alpha_0(2\alpha_c - 1) + (1 - \alpha_2)(1 - \alpha_c)^2(1 + \sigma_w^2)$$
and
\[ \Omega_0 = \frac{(1-\alpha_2)(1-\alpha_c) - \alpha_0 + [\alpha_1(1-\alpha_c-\alpha_0) - (1-\alpha_2)(1-\alpha_c)+\alpha_0] e^{-\pi(N+N_g)\beta T_s+j2\epsilon(1+\delta)/N}}{1 - \alpha_1 e^{-\pi(N+N_g)\beta T_s+j2\epsilon(1+\delta)/N} - \alpha_2 e^{-2\pi(N+N_g)\beta T_s+j2\epsilon(1+\delta)/N}}. \]

Fig. 6.4 and Fig. 6.5 show the variation of \( \eta_{mse} \) as a function of \( \alpha_c, \epsilon \) and \( \sigma_w^2 \), when 1st-order and 2nd-order low-pass filtering are used, respectively. It can be seen that, in general, 1st-order low-pass filtering yields more MSE-reduction than 2nd-order. Taking into account the fact that the CPE estimation scheme with 1st-order low-pass filtering is simpler and the resulting estimation deviation is negligible when \( \alpha_c \) is properly selected, we can conclude that 1st-order filtering is preferable over the 2nd-order one in practice.

Clearly, in the case of 1st-order filtering, MSE-reduction is significant (more than 30%) under both low and medium SNR conditions, which are represented by \( \sigma_w^2 = 0.1 \) and \( \sigma_w^2 = 0.025 \), respectively, in Fig. 6.4. This is achieved when \( 0.25 \leq \alpha_c \leq 0.4 \). However, at high SNR (represented by \( \sigma_w^2 = 0.01 \) in Fig. 6.4), MSE-reduction is less since the CPE estimate given by (6.23) is already sufficiently accurate in this case, and hence a small value should be selected for the smoothing factor \( \alpha_c \). From Fig. 6.4, we note that \( \alpha_c \leq 0.3 \) should be appropriate so that at least 10% MSE reduction in CPE estimation is achievable under all circumstances. This requirement happens to match the analytical result on \( \sigma_E \). Thus, we set \( \alpha_c = 0.3 \) in our design for achieving balanced MSE-reduction over all SNR conditions.

### 6.2.4 CPE Correction

With the obtained CPE estimate, \( e^{j\hat{\theta}_m(i)} \), the CPE involved in \( \hat{y}_m(i)(k) \) can be corrected in frequency-domain as
\[ \hat{y}_m(i)(k) = \hat{y}_m(i)(k) f^*(e^{j\hat{\theta}_m(i)}), \quad k \in \mathbb{Z}_{N-1}^N. \] (6.36)

The resulting signals \( \hat{y}_m(i)(k) \) are now almost free of phase-error and are ready to be applied to the frequency-domain equalizer for further signal detection processing, as shown in Fig. 6.1.

It is important to note that, in all the above derivations, no trigonometric operation for angle calculation is required. In addition, the factor, \( 1/G_r \), required in (6.23), is
Figure 6.4: MSE-reduction ratio, $\eta_{mse}$, varies as a function of $\alpha_c$, $\epsilon$ and $\sigma_w^2$ with $\beta = 6$ KHz, when 1st-order low-pass filtering is used.

Figure 6.5: MSE-reduction ratio, $\eta_{mse}$, varies as a function of $\alpha_c$, $\epsilon$ and $\sigma_w^2$ with $\beta = 6$ KHz, when 2nd-order low-pass filtering is used.
only related to the CFR’s on pilot tones and thus need to be computed only once at the beginning of a frame. In this sense, the proposed CPE estimation and correction scheme can be also regarded as being free of division operations, and thus it is of low complexity.

## 6.3 Numerical Results

Similar to that in Section 5.5, in the simulation examples, we consider an OFDM-UWB system with the data transmission rate of 53.3 Mbps. We assume that TFC = 1, and the frame payload is 1024-byte long with perfect frame timing and initial symbol timing. We use the UWB channel models CM1, CM2, CM3 and CM4, each of which has 100 realizations with the worst 10 realizations of each channel model being dropped from all the MSE and frame error rate (FER) related performance evaluation. This is due to the fact that the maximum excess delays of some channel realizations, in particular those of CM3 and CM4, are non-negligibly longer than $N_g$. The channel estimation scheme proposed in Chapter 5 is used for obtaining CFR estimates.

**Example 6.1 (SFO Estimation)**: In the first example, we evaluate the effectiveness of the proposed SFO estimator in terms of the MSE of the ANSFO estimate. The SFO estimation is performed based on $B = 21$ pilot-tone pairs and the MSE of the ANSFO estimate is evaluated at those OFDM symbols which have the same location (fixed at the 30th symbol in this case) across all the frames involved in the Monte Carlo tests. In the experiment, we assume that CPE is absent. As shown in Fig. 6.6 and Fig. 6.7, different amounts of SFO, including 0 ppm (low), 20 ppm (medium) and 40 ppm (high), are intentionally used in the simulations to demonstrate the robustness of the proposed SFO mitigation scheme over all possible scenarios. As expected, under all channel conditions, the weighted SFO estimator performs much better than the basic solution using conventional angle-based algorithm. Furthermore, the proposed combined SFO estimator performs either comparably to or better than the weighted estimator under low and medium SNR conditions, whereas the latter outperforms the former when the SNR becomes high. In particular, the combined estimator which involves certain
approximations, manifests an error floor phenomenon at the high SNR end, as can be seen in Fig. 6.6. However, when SNR is high, the SFO estimates achieved by both the weighted estimator and the combined estimator are already sufficiently accurate. Therefore, the error floor phenomenon in the case of the proposed combined SFO estimator does not cause any significant system performance degradation as confirmed through our system performance simulations.

![MSE performance comparison for various SFO estimation methods under CM1 and CM2.](image)

**Figure 6.6:** MSE performance comparison for various SFO estimation methods under CM1 and CM2.

**Example 6.2 (CPE Estimation and Correction):** In the second example, we evaluate the performance of the proposed CPE tracking scheme. We assume that SFO is absent, and both CFO and PHN are present. In this case, \( \beta = 6 \) KHz and \( \epsilon \) is fixed at 0.01. Fig. 6.8 shows the MSE reduction ratio, \( \eta_{\text{mse}} \), for various error variances of the intra-OFDM-symbol based estimation, \( \sigma^2_\omega \), with known CFR. Clearly, when the channel is known, the simulation results obtained with both 1st-order and 2nd-order inter-OFDM-symbol smoothing (filtering) schemes match well with the analytical results. A similar observation has been made in the scenario where the CFR is actually estimated.
as shown in Fig. 6.9. As expected, in all scenarios, the 1st-order filtering achieves more MSE reduction than the 2nd-order one when $\alpha_c = 0.3$. This is confirmed further through observing their respective FER performance in Fig. 6.10 and Fig. 6.11. It can be seen that the FER performance achieved by 1st-order filtering is similar to that achieved with the assumption that the value of $\nu$ is known and slightly worse than that with the assumption that no phase error is involved. Moreover, 1st-order filtering performs slightly better than 2nd-order filtering, and both of them clearly outperform the scheme without inter-OFDM-symbol smoothing (i.e., with $\alpha_c = 0$). As a result, the 1st-order inter-OFDM-symbol filtering with $\alpha_c = 0.3$ is selected for CPE estimation in subsequent simulations.

**Example 6.3 (Overall System Performance)**: In the third example, we evaluate the overall performance of the system employing the proposed phase error mitigation scheme. We assume 40 ppm CFO and 40 ppm SFO, which correspond to the worst-case values as defined by the OFDM-UWB specification [12]. We suppose that the CFO is initially

![Figure 6.7: MSE performance comparison for various SFO estimation methods under CM3 and CM4.](image)
6.3 Numerical Results

Figure 6.8: CPE tracking performance comparison for 1st-order and 2nd-order inter-OFDM-symbol smoothing methods with known CFR under CM1.

Figure 6.9: CPE tracking performance comparison for 1st-order and 2nd-order inter-OFDM-symbol smoothing methods with estimated CFR under CM1.
6.3 Numerical Results

Figure 6.10: FER performance comparison for various CPE tracking methods with $\epsilon = 0.01$ and $\beta = 6$ KHz under CM1 and CM3.

Figure 6.11: FER performance comparison for various CPE tracking methods with $\epsilon = 0.01$ and $\beta = 6$ KHz under CM2 and CM4.
6.3 Numerical Results

estimated using the preamble and corrected in time-domain in the subsequent OFDM symbols. In this example, we first focus on the case where SFO is absent but CFO is present, and then turn our attention to the case where both SFO and CFO are present. In both cases, PHN with $\beta = 6$ KHz is present and the CPE’s are tracked using the proposed CPE estimator with 1st-order filtering ($\alpha_c = 0.3$ and $\nu = 1$). Fig. 6.12 shows their FER performance curves. For comparison, the FER performance when no phase error is present is also shown. As can be seen from Fig. 6.12, the proposed CPE tracking scheme experiences only a small FER performance loss, e.g., about 0.3 dB at FER = 0.08 – the performance comparison point specified in [12], in all channel environments.

Moreover, it can be observed from Fig. 6.12 that, compared to the FER performance in the absence of phase error, our proposed phase error suppression solution experiences a performance loss of only about 0.5 dB at FER = 0.08 in the presence of the worst-case SFO, CFO and PHN. Fig. 6.12 further shows the FER performance in an ‘all-are-ideal’ case where perfect channel knowledge and no phase error are assumed. In comparison, the proposed solution experiences about 2.2 $\sim$ 2.8 dB FER performance loss at FER = 0.08 in all channel environments. Taking into account that these results are obtained under very low SNR conditions, it suffices to say that the proposed scheme is efficient and effective for phase error suppression in multi-band OFDM-UWB systems.

Finally, in the example, we also evaluate the performance of the SFO estimator with reduced number of pilot-tone pairs as described in Section 6.1.3. Fig. 6.13 compares the FER performance of the overall system when the SFO estimator employs 21 or 6 pilot-tone pairs. Observe that the simplified SFO estimator (with $\overline{B} = 6$ pilot-tone pairs) experiences a performance loss of only about 0.3 dB at FER = 0.08 when compared with the more sophisticated estimator (with $B = 21$ pilot-tone pairs). Taking into account that the simplified SFO estimator offers significant reduction in computational complexity, one may find it more suitable, compared to others, for implementation of OFDM-UWB devices in practice.
6.3 Numerical Results

Figure 6.12: FER performance of the overall system under various assumptions on phase error and channel condition.

Figure 6.13: FER performance comparison for SCO tracking using 21 and 6 pilot-tone pairs.
6.4 Concluding Remarks

In this chapter, we have proposed a novel phase error mitigation solution which is highly suitable for multi-band OFDM-based UWB systems. The solution has been verified to be effective and efficient in terms of performance and implementation complexity. The proposed scheme decomposes the phase error into the related SFO and CPE, which are suppressed separately. The approaches proposed for estimation of SFO and CPE are pilot-tone based and CFR-weighted with the virtues of simplicity and ease of implementation. Each of them is improved by using a robust and yet simple error suppression technique. In particular, we have shown that, in the presence of both residual CFO and PHN, the intra-OFDM-symbol CPE estimator can be significantly improved by employing an inter-OFDM-symbol smoothing technique for achieving further suppression of estimation error. Our analysis shows that the inter-OFDM-symbol smoothing with 1st-order filtering is preferable over that with 2nd-order filtering in practical implementation of OFDM-UWB devices. In addition, SFO compensation is performed jointly in time-domain and frequency-domain. This avoids the need for a time-domain interpolator which becomes expensive in silicon implementation of UWB devices. It has been shown that, with the proposed phase-error suppression technique, the overall system can achieve FER performance close to that of the system with no phase error even in the very low SNR regime.
Chapter 7

ML Estimation in OFDMA Systems

As discussed in Chapter 1, the generalized carrier-assignment scheme (GCAS) based OFDMA system, where each user can select the best available subcarriers, is becoming more and more attractive as the GCAS allows dynamic resource allocation and provides more flexibility than subband-based or interleaved schemes. The problem of estimating the synchronization parameters and channel responses of all active users in the uplink of such an OFDMA system is known to be challenging. Fortunately, recently, Pun et al. [39] have investigated this issue and proposed a training sequence based, thus semi-blind, ML joint estimation of carrier frequency (CFO), timing error, and channel response. The exact ML solution to this problem is prohibitively complex since it requires exhaustive search over a multi-dimensional space. Noting this, Pun et al. resorted to the alternating-projection (AP) algorithm [117] and proposed a simpler scheme, in which the CFOs are estimated sequentially instead of being jointly estimated. The resulting scheme, called AP frequency estimator (APFE) can reduce the multi-dimensional search to a sequence of simple 1-D searches.

In this chapter, after describing the signal model for OFDMA uplink transmission under discussion, we give a review of the APFE and its simplified version, the so called approximate APFE (AAPFE).
7.1 Signal Model for Generalized OFDMA Uplink

Suppose that \( K \) active users simultaneously communicate with the base station (BS) in the uplink of an OFDMA system using \( N \) subcarriers. Let \( N_k \) denote the number of the subcarriers assigned to the \( k \)th user, \( k = 1, \ldots, K \), and let \( i_{k,j}, j = 1, \ldots, N_k \) be the \( j \)th subcarrier of the \( k \)th user. Without loss of generality, we assume \( i_{k,1} < \cdots < i_{k,N_k} \), \( \bigcup_{k=1}^{K} \{i_{k,p}\}_{p=1}^{N_k} \subseteq \{0, \ldots, N-1\} \), and \( \bigcap_{k=1}^{K} \{i_{k,p}\}_{p=1}^{N_k} = \emptyset \).

As depicted in Fig. 7.1, the information stream of the \( k \)th user is parsed into blocks of length \( N_k \). Let \( s_k(n) \) be the \( n \)th transmission block of the \( k \)th user and let \( s_k(nN_k + j) \) denote its \( j \)th data symbol, i.e., \( s_k(n) := [s_k(nN_k), s_k(nN_k+1), \ldots, s_k(nN_k+N_k-1)]^T \).

After the serial-to-parallel (S/P) conversion, \( (N - N_k) \) zeros are inserted into \( s_k(n) \) to form a \( N \times 1 \) vector, \( q_k(n) \), whose \( i \)th entry is \( s_k(nN_k + p) \) if \( i = i_{k,p} \) for some \( p \), and is zero otherwise. The corresponding time-domain vector of \( q_k(n) \) is obtained as \( c_k(n) = F_N^{H} q_k(n) \), where the superscript ‘\( H \)’ denotes Hermitian transpose and \( F_N \) denotes the DFT matrix of size \( N \).

A cyclic prefix (CP) of length \( N_g \) is appended in front of \( c_k(n) \) for the purpose of eliminating inter-block interference (IBI). The resulting vector \( u_k(n) \) of length \( N_p = N + N_g \), after parallel-to-serial (P/S) conversion, is transmitted over the channel.

Consider \( g_k^{(TX)}(t) \) as the pulse shaping filter at the \( k \)th user’s transmitter, \( h_k^{(PR)}(t) \) as the impulse response of the propagation medium, and \( g_k^{(RX)}(t) \) as the shaping filter at the BS receiver, where \( t \) denotes continuous-time. The discrete-time baseband channel
impulse response (CIR) vector of the kth user to the BS can be expressed as \( h_k := [h_k(0), h_k(1), \cdots, h_k(N_h)]^T \), where \( h_k(l) = g_k^{(TX)}(t) * h_k^{(PR)}(t) * g_k^{(RX)}(t) \) at \( t = T - \tau_k \) and \( \ast \) denotes linear convolution. Taking both timing offset and propagation delay into account, the parameter \( \tau_k \) can be expressed as \( \tau_k := d_k T + \hat{\tau}_k \), where \( d_k \) denotes an integer number of the sampling period \( T \) and \( \hat{\tau}_k \) is the resulting fractional part. Note that the first \( d_k \) elements of \( h_k \) are zeros, and \( \hat{\tau}_k \) can be incorporated into the CIR of the kth user. Thus, the estimation of \( \tau_k \) can be indirectly obtained through the estimation of the CIR \( h_k \).

We assume the same channel order, \( N_h \), for all users. In practice, even though each user may have its own channel order, this assumption is still valid as zero taps can be padded to the end of shorter CIR vectors to make their length be \( N_h \). Also, the condition, \( N_h < N_g \), can be satisfied such that IBI can be effectively eliminated. As a result, in the sequel, we use \( \xi_k \) to represent the augmented CIR vector of the kth user of length \( N_g \), i.e.,

\[
\xi_k := [h_k^T, 0_{(N_g - N_h - 1) \times 1}]^T.
\]

We consider a quasi-synchronous scenario in which each user achieves timing acquisition via a downlink synchronization channel before initiating the uplink transmissions [37, 39]. Taking CFOs into account, the signal samples at the BS receiver filter output can be expressed as

\[
x(i) = \sum_{k=1}^{K} \{ e^{j\omega_k i} \sum_{l=0}^{N_g} \xi_k(l) u_k(i - l) \} + v(i) \tag{7.1}
\]

where \( \xi_k(l) := [\xi_k]_l \), \( u_k(nN_p + m) \) is the serialized version of the nth block \( u_k(n) \) with \( u_k(nN_p + m) = [u_k(n)]_m \), \( v(i) \) is the receiver noise modelled as a circularly symmetric white Gaussian process with variance \( \sigma_v^2 \), and \( \omega_k := 2\pi \Delta f_k / N \) with \( \Delta f_k \) being the kth user’s CFO normalized by the subcarrier spacing. In practice, we assume \( |\Delta f_k| < 0.5 \), i.e., \( |\omega_k| < \pi / N \).

With the BS timing as a reference, the received samples \( x(i) \) are segmented into consecutive OFDMA blocks of length \( N_p \). Through a S/P converter, the samples in the block are converted into a \( N_p \times 1 \) vector, denoted by \( \mathbf{x}(n) \) with \( [\mathbf{x}(n)]_m := x(nN_p + m) \). After removing the CP, i.e., discarding the first \( N_g \) elements of \( \mathbf{x}(n) \), a \( N \times 1 \) vector,
where $D(\omega_k)$ is defined as $\text{diag}\{e^{j\omega_k N_g}, e^{j\omega_k(N_g+1)}, \ldots, e^{j\omega_k(N_g+N-1)}\}$, $U_k(n)$ is a $N \times N_g$ column circulant matrix with the first column being $c_k(n)$, and $v(n)$ is a $N \times 1$ zero-mean Gaussian noise vector with covariance matrix $\sigma^2_v I_N$. Typically, the OFDMA blocks are organized in frames and the CFO of each user is estimated during a training block at the beginning of each frame. Hence, without loss of generality, we consider a single transmission block corresponding to $n = 0$ and omit the block index $n$ for notational simplicity. Defining $\omega = [\omega_1, \omega_2, \ldots, \omega_K]^T$, $\xi = [\xi_1^T, \xi_2^T, \ldots, \xi_K^T]^T$, and $Z(\omega) = [D(\omega_1)U_1, D(\omega_2)U_2, \ldots, D(\omega_K)U_K]$, we can rewrite (7.2) as

$$y = Z(\omega)\xi + v.$$  

Further, defining $\Phi(\omega) := [Z^H(\omega)Z(\omega)]^{-1}$ and $\hat{Z}(\omega) := Z(\omega)\Phi(\omega)Z^H(\omega)$, we can readily obtain the following properties of $\Phi(\omega)$ and $\hat{Z}(\omega)$ for later use:

$$\Phi(\omega + \Delta\omega i_K) = \Phi(\omega)$$  

$$\hat{Z}(\omega) = \hat{Z}^H(\omega)\hat{Z}(\omega)$$  

and

$$\hat{Z}(\omega + \Delta\omega i_K) = D_N(\Delta\omega)\hat{Z}(\omega)D_N^H(\Delta\omega)$$

where $\Delta\omega$ is a scalar variable.

### 7.2 Existing ML Based CFO Estimators

Under the assumption that the elements of $v$ are independent and identically distributed Gaussian random variables, the ML estimate of the CFO vector $\omega$, $\hat{\omega} := [\hat{\omega}_1, \hat{\omega}_2, \ldots, \hat{\omega}_K]^T$, and the ML estimate of the CIR vector $\xi$, $\hat{\xi} := [\hat{\xi}_1^T, \hat{\xi}_2^T, \ldots, \hat{\xi}_K^T]^T$, are jointly given by [39, Eq. (14)],

$$[\hat{\xi}; \hat{\omega}] = \arg\min_{\omega, \xi} \{\|y - Z(\omega)\hat{\xi}\|^2\}$$
where $\tilde{\omega} := [\tilde{\omega}_1, \tilde{\omega}_2, \cdots, \tilde{\omega}_K]^T$ and $\tilde{\xi} := [\tilde{\xi}_1^T \xi_2^T \cdots \xi_K^T]^T$ are the trial variables representing the possible values of $\omega$ and $\xi$, respectively, during search. Further, the ML estimate of $\omega$ can be obtained as

$$\hat{\omega} = \arg \max_\omega \{ \| \hat{Z}(\tilde{\omega}) y \|^2 \}. \quad (7.8)$$

The maximization problem (7.8) can be directly solved by using an exhaustive grid-search over the multi-dimensional space spanned by $\tilde{\omega}$, which requires prohibitively large computational complexity and is clearly not feasible in practice. To reduce computational complexity, an APFE scheme based on the AP technique was proposed in [39]. The APFE is an iterative approach that consists of cycles and steps. Each cycle comprises of $K$ steps, and in each step one user’s CFO estimate is updated while the CFO estimates of the other users remain unchanged. Without loss of generality, the natural ordering $k = 1, 2, \cdots, K$ is followed in updating the users’ CFO estimates.

Let $\hat{\omega}_k^{(i)}$ be the estimate of $\omega_k$ at the $i$th cycle and $\tilde{\omega}_k$ be a possible value of $\omega_k$. We define a $K \times 1$ vector $\tilde{\omega}_k^{(i)}$ as $\tilde{\omega}_k^{(i)} := [\hat{\omega}_1^{(i+1)}, \cdots, \hat{\omega}_{k-1}^{(i+1)}, \tilde{\omega}_k, \hat{\omega}_{k+1}^{(i+1)}, \cdots, \hat{\omega}_K^{(i+1)}]^T$. In the $k$th step of the $(i + 1)$th cycle, the AP algorithm updates the estimate of $\omega_k$ by solving the following 1-D maximization problem:

$$\hat{\omega}_k^{(i+1)} = \arg \max_{\tilde{\omega}_k} \{ \| \hat{Z}(\tilde{\omega}_k^{(i)}) y \|^2 \}. \quad (7.9)$$

The estimation proceeds in this manner until the completion of the $(i + 1)$th cycle where $\hat{\omega}_K^{(i+1)}$ is computed. Multiple cycles, say $N_c$, are performed until the CFO estimates converge.

The maximization problem in (7.9) is solved via a dense grid-search over the interval spanned by $\tilde{\omega}_k$. However, computation of $\hat{Z}(\tilde{\omega}_k^{(i)})$ requires inverting a $KN_g \times KN_g$ matrix for each value of $\tilde{\omega}_k$, which is computationally costly. The high complexity can be reduced to some extent by splitting $Z(\tilde{\omega}_k^{(i)})$ into two parts: 1) $A_k^{(i)}(\tilde{\omega}_k)$, containing all columns related to $\omega_k$, and 2) $\tilde{A}_k^{(i)}(\tilde{\omega}_k)$, containing all of the remaining columns of $Z(\tilde{\omega}_k^{(i)})$. Using $\tilde{\omega}_k(q)$, for $q = 1, \ldots, N_\omega$, to denote the $N_\omega$ possible values of $\tilde{\omega}_k$ over which the grid-search evaluation is performed, the complete APFE described in [39] can be summarized as shown in Table 7.1.
### 7.2 Existing ML Based CFO Estimators

#### Table 7.1: Summary of the APFE scheme.

**INITIALIZE** $\hat{\omega}$ with $\hat{\omega}^{(0)}$;

**BEGIN Cycle** for $i = 0$ to $N_c - 1$

**Begin Step** for $k = 1$ to $K$

$$
\begin{align*}
\hat{R}_k^{(i)} &= \hat{A}_k^{(i)}(\hat{\omega}_k)[(\hat{A}_k^{(i)}(\hat{\omega}_k))^\mathsf{H} \hat{A}_k^{(i)}(\hat{\omega}_k)]^{-1}(\hat{A}_k^{(i)}(\hat{\omega}_k))^\mathsf{H} \\
&= (I_N - \hat{R}_k^{(i)}) \hat{A}_k^{(i)}(\hat{\omega}_k(q))
\end{align*}
(7.10)
$$

**Begin Grid Search** for $q = 1$ to $N_\omega$

$$
\begin{align*}
\hat{R}_k^{(i)}(\hat{\omega}(q)) &= (I_N - \hat{R}_k^{(i)}(\hat{\omega}_k(q))) \\
Q_k^{(i)}(\hat{\omega}(q)) &= \hat{R}_k^{(i)}(\hat{\omega}(q)) \times \\
&\left[(\hat{R}_k^{(i)}(\hat{\omega}(q)))^\mathsf{H} \hat{R}_k^{(i)}(\hat{\omega}(q))\right]^{-1}(\hat{R}_k^{(i)}(\hat{\omega}(q)))^\mathsf{H} \\
&= \left[(\hat{R}_k^{(i)}(\hat{\omega}(q)))^\mathsf{H} \hat{R}_k^{(i)}(\hat{\omega}(q))\right]^{-1}(\hat{R}_k^{(i)}(\hat{\omega}(q)))^\mathsf{H}
(7.12)
\end{align*}
$$

**End Grid Search**

$$
\hat{\omega}_k^{(i+1)} = \arg \max_{\hat{\omega}_k(q)} \{\|Q_k^{(i)}(\hat{\omega}_k(q))y\|^2\}
(7.13)
$$

**End Step**

**End Cycle**

The grid search in APFE requires inverting a $N_g \times N_g$ matrix for each value of $\hat{\omega}_k$. This can be avoided by introducing the so-called Approximate APFE (AAPFE). Following the description in [39], the AAPFE can be summarized as shown in Table 7.2.

Clearly, the AAPFE is much simpler than the APFE since $\|Q_k^{(i)}(\hat{\omega}(q))y\|^2$ has been approximated in (7.17) by using a truncated von Neumann series of order $L$ [142] such that the $N_g \times N_g$ matrix inversion required for each $\hat{\omega}_k$ in the APFE (see (7.12)) is avoided. The required inversion of $U_l^lU_l$, i.e., $\Lambda(l)$, is independent of $\hat{\omega}_k$ and thus can be computed and/or pre-stored before entering the grid search loop. In this case, computational complexity is reduced at the expense of estimation accuracy. In particular, the AAPFE’s accuracy is $L$-dependent as we will show in Chapter 8.
7.3 Cramér–Rao Bound (CRB)

Referring to [39] and [143], in this section, we derive the CRB for joint estimation of \( \omega \) and \( \xi \) based on the model in (7.3). For the convenience of discussion, (7.3) can be rewritten as

\[
y = \sum_{k=1}^{K} D(\omega_k)U_k \xi_k + v
\]

where \( v \) is a \( N \times 1 \) zero-mean Gaussian noise vector with covariance matrix \( \sigma_v^2 I_N \). Let \( \xi_R \) and \( \xi_I \) be the real and imaginary parts, respectively, of \( \xi \) and \( \kappa := [\omega^T, \xi_R^T, \xi_I^T]^T \) be the set of the unknown parameters. The components of the Fisher Information Matrix
(FIM) $\Theta$ are given by

$$[\Theta]_{m,l} = -\mathbb{E} \left\{ \frac{\partial^2 \ln p(y; \kappa)}{\partial \kappa_m \partial \kappa_l} \right\}, \quad m \in \mathbb{Z}_{1}^{K(2N_g+1)}, \quad l \in \mathbb{Z}_{1}^{K(2N_g+1)}$$

(7.20)

where $\kappa_i = [\kappa]_i$ and $p(y; \kappa)$ is the probability density function (PDF) of $y$, which is given by

$$p(y; \kappa) = \frac{1}{(\pi \sigma_v^2)^N} \exp \left\{ -\frac{1}{\sigma_v^2} \| y - \sum_{k=1}^{K} D(\omega_k)U_k \xi_k \|^2 \right\}.$$  

(7.21)

Substituting (7.21) into (7.20) yields

$$[\Theta] = \frac{2}{\sigma_v^2} \left[ \begin{array}{ccc} \Re[B^H(\omega)B(\omega)] & \Im[B^H(\omega)Z(\omega)] & \Re[B^H(\omega)Z(\omega)] \\ -\Im[Z^H(\omega)B(\omega)] & \Re[Z^H(\omega)Z(\omega)] & -\Im[Z^H(\omega)Z(\omega)] \\ \Re[Z^H(\omega)B(\omega)] & \Im[Z^H(\omega)Z(\omega)] & \Re[Z^H(\omega)Z(\omega)] \end{array} \right]$$

(7.22)

where we have defined the following quantities:

$$B(\omega) = \begin{bmatrix} b_1 & b_2 & \cdots & b_K \end{bmatrix}$$  

(7.23)

$$b_k = (D_N + N_g I_N)D(\omega_k)U_k \xi_k$$  

(7.24)

$$D_N = \text{diag}\{0, 1, \cdots, N-1\}.$$  

(7.25)

Finally, the CRB for the estimation of $\omega_k$ is given by

$$\text{CRB}(\omega_k) = [\Theta^{-1}]_{k,k}.$$  

(7.26)

The RHS of (7.26) can be rewritten in a more convenient form using arguments similar to those employed in [143]. Skipping the details, it can be shown that [39, 143]

$$\text{CRB}(\omega_k) = \frac{\sigma_v^2}{2} \left[ \left( \Re \left\{ C^H(\omega)(I_N - \hat{Z}(\omega))C(\omega) \right\} \right) \right]^{-1}_{k,k}$$  

(7.27)

where $C(\omega) = \begin{bmatrix} c_1 & c_2 & \cdots & c_K \end{bmatrix}$ and $c_k = D_N D(\omega_k)U_k \xi_k$, $k \in \mathbb{Z}_1^K$. Finally, since $\omega_k = 2\pi \Delta f_k / N$, we have

$$\text{CRB}(\Delta f_k) = \frac{N^2 \sigma_v^2}{8\pi^2} \left[ \left( \Re \left\{ C^H(\omega)(I_N - \hat{Z}(\omega))C(\omega) \right\} \right) \right]^{-1}_{k,k}.$$  

(7.28)

As indicated by (7.28), CRB($\Delta f_k$) depends on the specific channel realization. Since we are interested in the average performance of the CFO estimators, the RHS of (7.28) is numerically averaged over all channel realizations, and the result is taken as a baseline in the discussion in Chapter 8.
7.4 Convergence Property of ML Estimation

Before we proceed to introduce a new method for CFO estimation in the next chapter, some remarks on the convergence of the AP algorithm are essential. In [39], without an explicit analysis, both the APFE and the AAPFE were shown, by extensive simulations, to converge to the true CFOs in a few cycles. Similar convergence behaviors are also observed from our simulation examples. In particular, we find that the convergence performance of the AP algorithms can actually be explained by studying the \( \tilde{\omega} \)-dependent behavior of the Euclidean norm \( \| \hat{Z}(\tilde{\omega})y \| \) in (7.8). We illustrate it by using a numerical example as shown in Fig. 7.2, where the variation in \( \| \hat{Z}(\tilde{\omega})y \| \) versus the variation in Euclidean distance, \( \| \tilde{\omega} - \omega \| \), is plotted. Observe from Fig. 7.2 that \( \| \hat{Z}(\tilde{\omega})y \| \) does not monotonically increase as \( \| \tilde{\omega} - \omega \| \) decreases, except when \( \tilde{\omega} \) is close to \( \omega \). Basically, this means that the AP algorithms can not achieve convergence in a single cycle. However, \textit{globally}, \( \| \hat{Z}(\tilde{\omega})y \| \) versus \( \| \tilde{\omega} - \omega \| \) maintains a monotonic-decreasing property whereby the fast and steady convergence of the AP algorithms can be expected, that is, most likely \( \| \tilde{\omega} - \omega \| \) becomes much smaller after each cycle.

7.5 Concluding Remarks

In this chapter, we have presented a signal model for the GCAS based OFDMA uplink transmission. The ML CFO estimation for this application is investigated with focus on its CRB, convergence behavior and implementation complexity. Two AP based ML CFO estimation schemes, namely, APFE and AAPFE, are reviewed and analyzed. These schemes are shown to be of fairly high computational complexity that may prevent their use in practice. This weakness of the existing solutions motivated us to develop a new scheme with reduced implementation complexity, as we will show in Chapter 8.
Figure 7.2: Illustration of convergence of the exact ML estimate of $\omega$. The Euclidean norm, $\| \hat{Z}(\tilde{\omega})y \|$ (denoted as $\Omega$), versus the Euclidean distance, $\| \tilde{\omega} - \omega \|$, with $K = 4$ active users, $N = 128$ subcarriers and the true CFOs $\omega = 2\pi/N \times [0.245, 0.26, 0.09, 0.05]$. 
Chapter 8

New Approach for OFDMA Uplink

CFO Estimation

In Chapter 7, we have observed that alternating-projection frequency estimator (APFE) and approximate APFE (AAPFE) can be employed to significantly reduce the complexity associated with the maximization of the likelihood function of carrier frequency offset (CFO) estimation over a multi-dimensional space. However, careful examination of these algorithms shows that APFE is still too complicated for its use in practical systems and the reduced computational complexity of AAPFE is obtained at the expense of degraded system performance as can be seen in [39] as well as our discussion in this chapter. Therefore, in this chapter, we propose a novel iterative CFO estimator, which not only is more accurate than AAPFE but also is comparable to APFE in performance with complexity that is much less than that of APFE.

8.1 Divide-and-Update Frequency Estimator (DUFE)

The proposed estimation approach consists of a number of iterations, each of which involves two steps, namely, a primitive CFO estimation followed by a divide-and-update CFO adjustment.
8.1 Divide-and-Update Frequency Estimator (DUFE)

Define a $K \times 1$ vector $\hat{\omega}^{(i)}$ as

$$
\hat{\omega}^{(i)} := [\hat{\omega}_1^{(i)}, \ldots, \hat{\omega}_k^{(i)}, \ldots, \hat{\omega}_K^{(i)}]^T \quad (8.1)
$$

where $i$ denotes the iteration index. Initially, we select $\hat{\omega}^{(0)} = 0_{K \times 1}$ being a $K \times 1$ vector whose elements are all zeros, and accordingly $\Phi(\hat{\omega}^{(0)}) = [Z^H(\hat{\omega}^{(0)})Z(\hat{\omega}^{(0)})]^{-1}$, which can be computed and pre-stored for later use.

8.1.1 Step 1 – Primitive CFO Estimation

In the $(i + 1)$th iteration, ML CFO estimation is first performed as if all users in the OFDMA system had the same residual CFO (i.e., difference between actual CFO and CFO estimate in the $i$th iteration), resulting in a same residual CFO estimate for all users. We term this as primitive CFO estimation.

In a way similar to that in single-user OFDM systems [107], the primitive CFO estimate, $\hat{\delta}_\omega^{(i)}$, and the normalized estimation error, $\varepsilon^{(i)}$, are defined as

$$
\hat{\delta}_\omega^{(i)} = \arg \max_{\delta_\omega} \{ y^H D(\delta_\omega) \tilde{Z}(\hat{\omega}^{(i)}) D^H(\delta_\omega) y \} \quad (8.2)
$$

$$
\varepsilon^{(i)} = \frac{\| y - D(\hat{\delta}_\omega^{(i)}) \tilde{Z}(\hat{\omega}^{(i)}) D^H(\hat{\delta}_\omega^{(i)}) y \|^2}{\| y \|^2} \quad (8.3)
$$

where $\tilde{Z}(\hat{\omega}^{(i)}) := Z(\hat{\omega}^{(i)})\Phi(\hat{\omega}^{(i)})Z^H(\hat{\omega}^{(i)})$.

Adding $\hat{\delta}_\omega^{(i)}$ to each component of $\hat{\omega}^{(i)}$, we obtain an updated CFO estimate $\phi^{(i)}$ as

$$
\phi^{(i)} = \hat{\omega}^{(i)} + \hat{\delta}_\omega^{(i)} i_K \quad (8.4)
$$

where $i_K$ denotes a $K \times 1$ vector whose entries are all ones. Thus, since the primitive CFO estimation step adjusts all the CFO estimates equally (either increases all $\hat{\omega}_k^{(i)}$ or decreases all $\hat{\omega}_k^{(i)}$ by the same amount $\hat{\delta}_\omega^{(i)}$), it is only used as a starting point for estimating the residual CFO of each individual user in the current iteration. In other words, because different users may have different CFO’s in an OFDMA system, the CFO estimate $\phi^{(i)}$ in (8.4) cannot be used to directly update $\hat{\omega}^{(i)}$, and needs to be incorporated with a local-search based CFO adjustment step, which will be described next.
### 8.1.2 Step 2 – Divide-and-Update CFO Adjustment

Consider the $k$th user ($k = 1, 2, \cdots, K$) in the system. Let $\hat{\phi}^{(i)}_k$ be the $k$th entry of $\hat{\phi}^{(i)}$ and $\bar{\phi}^{(i)}_k(q)$ denote the values varying in a limited range centered around $\hat{\phi}^{(i)}_k$, given by

$$
\bar{\phi}^{(i)}_k(q) = \hat{\phi}^{(i)}_k + (q - \frac{N_s}{2})\Delta \varphi, \quad q = 0, 1, \ldots, N_s
$$

(8.5)

where $N_s$ and $\Delta \varphi > 0$ are used to control the search range and step-size, respectively. Note that $\bar{\phi}^{(i)}_k(q) \in (-\pi/N, \pi/N)$. Let $\bar{\phi}^{(i)}_k(q) := [\bar{\phi}^{(i)}_1(q), \bar{\phi}^{(i)}_{k-1}(q), \bar{\phi}^{(i)}_k(q), \bar{\phi}^{(i)}_{k+1}(q), \cdots, \bar{\phi}^{(i)}_K]^T$ and $\bar{Z}(\bar{\phi}^{(i)}_k(q)) := Z(\bar{\phi}^{(i)}_k(q)) \Phi(\bar{\phi}^{(i)}_k(q)) \bar{Z}^H(\bar{\phi}^{(i)}_k(q))$, for each possible value of $\bar{\phi}^{(i)}_k(q)$. Define $\gamma_k(q)$ as the normalized estimation error associated with $\bar{\phi}^{(i)}_k(q)$, i.e.,

$$
\gamma_k(q) = \frac{\|y - \bar{Z}(\bar{\phi}^{(i)}_k(q))y\|^2}{\|y\|^2} \approx \frac{\|y - \bar{Z}(\bar{\phi}^{(i)}_k(q))y\|^2}{\|y\|^2}
$$

(8.6)

where

$$
\bar{Z}(\bar{\phi}^{(i)}_k(q)) := Z(\bar{\phi}^{(i)}_k(q)) \Phi(\bar{\phi}^{(i)}_k(q)) \bar{Z}^H(\bar{\phi}^{(i)}_k(q)) = Z(\bar{\phi}^{(i)}_k(q)) \Phi(\bar{\omega}^{(i)}) \bar{Z}^H(\bar{\phi}^{(i)}_k(q)).
$$

(8.7)

The approximation in (8.6) follows from $\Phi(\bar{\phi}^{(i)}) \approx \Phi(\bar{\phi}^{(i)}_k(q))$, which holds when the local search range is small, i.e., $|\bar{\phi}^{(i)}_k(q) - \hat{\phi}^{(i)}_k|$ is small. The second equality in (8.7) follows from the definition of $\Phi(\bar{\omega}^{(i)})$ and (8.4) that $\Phi(\bar{\phi}^{(i)}) = \Phi(\bar{\omega}^{(i)})$. It is important to note that $\Phi(\bar{\omega}^{(i)})$ remains unchanged so that no matrix inversion is involved in this step. Note that, when $q = N_s/2$, $\gamma_k(q) = \varepsilon^{(i)}$ according to (8.3), and thus the computation of $\gamma_k(N_s/2)$ is not necessary.

Let $\eta_k^{(i)}$ be the minimum value of $\{\gamma_k(q)\}_{q=0}^{N_s}$, i.e., $\eta_k^{(i)} = \gamma_k(q_k)$, with $q_k := \arg\min_{q} \gamma_k(q)$. The corresponding variation of $\hat{\phi}^{(i)}_k$ is obtained as $\Delta \hat{\phi}^{(i)}_k = (q_k - N_s/2)\Delta \varphi$. Depending on the sign of $\Delta \hat{\phi}^{(i)}_k$, the CFO estimates are updated as

$$
\hat{\omega}^{(i+1)}_k = \begin{cases} 
\hat{\omega}^{(i)}_k + \delta^{\omega}_i \cdot \text{primitive adjustment} \\
\hat{\omega}^{(i)}_k + \bar{G}(\varepsilon^{(i)}, \eta^{(i)}) \Delta \hat{\phi}^{(i)}_k \cdot \text{local-search based adjustment}
\end{cases}
$$

if $\Delta \hat{\phi}^{(i)}_k \delta^{\omega}_i \geq 0, (8.8a)$

(8.8b)

otherwise,

where $\eta^{(i)} := [\eta_1^{(i)}, \eta_2^{(i)}, \cdots, \eta_K^{(i)}]^T$, and $\bar{G}(\varepsilon^{(i)}, \eta^{(i)})$ denotes a weighting factor that is a function of $\varepsilon^{(i)}$ and $\eta^{(i)}$ for achieving fast adaptation.
8.1 Divide-and-Update Frequency Estimator (DUF E)

Very likely, compared with the CFO estimates in the \(i\)th iteration, we will obtain more accurate CFO estimates for some users and less accurate estimates for the other users, after the primitive estimation in the \((i + 1)\)th iteration. Clearly, the less accurate estimates should not be accepted. We assess the quality and/or reliability of the updates by examining the sign of the quantity \(\Delta \hat{\varphi}_k^{(i)} \hat{\delta}_\omega^{(i)}\). If \(\Delta \hat{\varphi}_k^{(i)} \hat{\delta}_\omega^{(i)} \geq 0\), the primitive CFO estimate \(\hat{\omega}_k^{(i)}\) and the local-search based CFO adjustment \(\Delta \hat{\varphi}_k^{(i)}\) suggest to adjust \(\hat{\omega}_k^{(i)}\) in the same direction, and the corresponding updated CFO estimate is given by (8.8a). On the other hand, if \(\Delta \hat{\varphi}_k^{(i)} \hat{\delta}_\omega^{(i)} < 0\), the primitive estimation and local-search steps suggest opposite adjustments, and we simply maintain the CFO estimate obtained in the previous \((i)\)th iteration. Fig. 8.1 illustrates this CFO adjustment step for the case of \(K = 4\).

It should be noted that, under the condition \(\Delta \hat{\varphi}_k^{(i)} \hat{\delta}_\omega^{(i)} \geq 0\) in (8.8a), there may exist two special cases, which, most likely, are under the scenario when a local maximum is present in the vicinity of the current CFO estimates, \(\hat{\omega}^{(i)}\). First, we may have \(\hat{\delta}_\omega^{(i)} \neq 0\), and, \(\Delta \hat{\varphi}_{k_1}^{(i)} = 0, \forall k_1 \in \mathcal{S}\) and \(\Delta \hat{\varphi}_{k_2}^{(i)} \hat{\delta}_\omega^{(i)} < 0, \forall k_2 \in \bar{\mathcal{S}}\), where \(\mathcal{S}\) is a subset of \(\{k\}_1^K\) and \(\bar{\mathcal{S}}\) is the complement of \(\mathcal{S}\). Secondly, we may have \(\hat{\delta}_\omega^{(i)} = 0\) and \(\Delta \hat{\varphi}_{k_1}^{(i)} = 0, \forall k_1 \in \{k\}_1^K\).

For all \(k_0 \in \mathcal{S}\) in the first case and for \(k_0 = \arg \max_k \{\eta_k^{(i)}\}\) in the second case, the CFO estimates at the \((i + 1)\)th iteration are given by

\[
\hat{\omega}_k^{(i+1)} = \begin{cases} 
\hat{\omega}_k^{(i)} + \alpha_0 \cdot \text{sgn}(\hat{\omega}_k^{(i)} - \hat{\omega}_k^{(i-m)}) \cdot \mathcal{G}(\varepsilon^{(i)}, \eta^{(i)}), & k = k_0 \\
\hat{\omega}_k^{(i)}, & \text{otherwise} 
\end{cases} (8.9a)
\]

where \(\alpha_0\) is a small constant that depends on the system design, and, \(\text{sgn}(\hat{\omega}_k^{(i)} - \hat{\omega}_k^{(i-m)})\) is the sign of \(\hat{\omega}_k^{(i)} - \hat{\omega}_k^{(i-m)}\) with \(m\) being the smallest positive integer which satisfies \(\hat{\omega}_k^{(i)} \neq \hat{\omega}_k^{(i-m)}\).

Although it is highly unlikely, we may have \(\Delta \hat{\varphi}_k^{(i)} \hat{\delta}_\omega^{(i)} < 0\) for all users. In such a case, we simply update the CFO estimate for a single user, say the \(k_1\)th user, who has the smallest \(\eta_k^{(i)}\) value, i.e., \(k_1 = \arg \min_k \{\eta_k^{(i)}\}\), and hold the CFO estimates of other users. Accordingly, the CFO estimates at the \((i + 1)\)th iteration are given by

\[
\hat{\omega}_k^{(i+1)} = \begin{cases} 
\hat{\omega}_k^{(i)} + \hat{\delta}_\omega^{(i)}, & k = k_1 \\
\hat{\omega}_k^{(i)}, & \text{otherwise}. 
\end{cases} (8.10b)
\]

Succinctly speaking, the users in the OFDMA systems are divided into two groups,
8.1 Divide-and-Update Frequency Estimator (DUFE)

After \((i+1)\)th iteration

\[-\pi/N \quad \hat{\omega}_1^{(i+1)} \quad 0 \quad \hat{\omega}_2^{(i+1)} \quad \hat{\omega}_3^{(i+1)} \quad \hat{\omega}_4^{(i+1)} \quad \pi/N\]

After \(i\)th iteration

\[-\pi/N \quad \hat{\omega}_1^{(i)} \quad 0 \quad \hat{\omega}_2^{(i)} \quad \hat{\omega}_3^{(i)} \quad \hat{\omega}_4^{(i)} \quad \pi/N\]

True CFO’s

\[-\pi/N \quad \omega_1 \quad 0 \quad \omega_2 \quad \omega_3 \quad \omega_4 \quad \pi/N\]

Figure 8.1: Illustration of the user grouping in the \((i+1)\)th iteration. Users 1 and 2 with
\(\Delta \hat{\phi}_k^{(i)} \delta_{\omega}^{(i)} < 0\), and Users 3 and 4 with \(\Delta \hat{\phi}_k^{(i)} \delta_{\omega}^{(i)} \geq 0\).

and their CFO estimates are updated differently according to either (8.8), (8.9) or (8.10). The users that satisfy the conditions attached to (8.8a), (8.9a) or (8.10a) belong to the first group and the rest belong to the second group. Therefore, we call this adjustment process ‘divide-and-update adjustment’.

At the end of the divide-and-update adjustment process, we compute \(\Phi(\hat{\omega}^{(i+1)})\) using the updated CFO estimates, \(\hat{\omega}^{(i+1)}\). The computation of \(\Phi(\hat{\omega}^{(i+1)})\) can be performed recursively as will be described next.

8.1.3 Computation of \(\Phi(\hat{\omega}^{(i+1)})\)

Without loss of generality, suppose that in Step 2 of the \((i+1)\)th iteration, the CFO estimates of \(K_i\) users with indices \(1, \ldots, K_i\) have been updated. Define a \(K \times 1\) vector \(\hat{\omega}_m^{(i+1)}\) as

\[
\hat{\omega}_m^{(i+1)} := [\hat{\omega}_{m,1}^{(i+1)}, \hat{\omega}_{m,2}^{(i+1)}, \ldots, \hat{\omega}_{m,K}^{(i+1)}]^T
= [\hat{\omega}_1^{(i+1)}, \ldots, \hat{\omega}_m^{(i+1)}, \ldots, \hat{\omega}_{K_i}^{(i)}, \hat{\omega}_{K_i+1}^{(i)}, \ldots, \hat{\omega}_K^{(i)}]^T
\]

(8.11)

for \(m = 1, \ldots, K_i\). We use \(\omega_{m,k}^{(i+1)}\) to denote the \(k\)th element of \(\hat{\omega}_m^{(i+1)}\), i.e., \(\hat{\omega}_{m,k}^{(i+1)} = [\hat{\omega}_m^{(i+1)}]_k\). Furthermore, we have \(\omega_0^{(i+1)} = \hat{\omega}^{(i)}\) and \(\hat{\omega}_{K_i+1}^{(i+1)} = \hat{\omega}^{(i)}\). Recall that we have defined \(\Phi(\hat{\omega}^{(i)})\) as \(\Phi(\hat{\omega}^{(i)}) = [Z^H(\hat{\omega}^{(i)})Z(\hat{\omega}^{(i)})]^{-1}\). In the following, we show that \(\Phi(\hat{\omega}^{(i+1)})\) can be computed recursively in \(K_i\) steps. In the \(m\)th recursion, \(\Phi(\hat{\omega}_m^{(i+1)})\) is
obtained by updating \( \Phi(\hat{\omega}_{m-1}^{(i+1)}) \) with \( \hat{\omega}_m^{(i+1)} \). Denote \( \hat{Z}_{m,k}^{(i+1)} = D(\hat{\omega}_{m,k}^{(i+1)}) U_k \), for \( k = 1, \ldots, K \) and \( m = 1, \ldots, K \). From (8.11) and the definition of \( Z(\omega) \), we have

\[
Z(\hat{\omega}_{m-1}^{(i+1)}) = [Z_{m-1,1}^{(i+1)}, Z_{m-1,2}^{(i+1)}, \ldots, Z_{m-1,K}^{(i+1)}]
\]

and

\[
Z(\hat{\omega}_m^{(i+1)}) = [Z_{m,1}^{(i+1)}, Z_{m,2}^{(i+1)}, \ldots, Z_{m,K}^{(i+1)}].
\]

Further examination shows that only the \( m \)th block of \( Z(\hat{\omega}_m^{(i+1)}) \) is different from that of \( Z(\hat{\omega}_{m-1}^{(i+1)}) \), i.e., \( Z_{m,m}^{(i+1)} \neq Z_{m-1,m}^{(i+1)} \). Defining

\[
\Delta Z_m^{(i+1)} := Z_{m,m}^{(i+1)} - Z_{m-1,m}^{(i+1)}
\]

and

\[
B_k^{(i+1)} := (\Delta Z_m^{(i+1)})^H Z_{m-1,k}^{(i+1)}
\]

for \( k = 1, \ldots, K \), we can further derive that

\[
Z^H(\hat{\omega}_m^{(i+1)}) Z(\hat{\omega}_m^{(i+1)}) = Z^H(\hat{\omega}_{m-1}^{(i+1)}) Z(\hat{\omega}_{m-1}^{(i+1)}) + C_m^{(i+1)} \bar{C}_m^{(i+1)}
\]  

(8.12)

where \( C_m^{(i+1)} \) is a \( KN_g \times 2N_g \) matrix defined as

\[
C_m^{(i+1)} := \begin{bmatrix}
0_{N_g} & \cdots & 0_{N_g} & I_{N_g} & 0_{N_g} & \cdots & 0_{N_g} \\
B_1^{(i+1)} & \cdots & B_{m-1}^{(i+1)} & 0_{N_g} & B_{m+1}^{(i+1)} & \cdots & B_K^{(i+1)}
\end{bmatrix}^H
\]

and \( \bar{C}_m^{(i+1)} \) is a \( 2N_g \times KN_g \) matrix defined as

\[
\bar{C}_m^{(i+1)} := \begin{bmatrix}
B_1^{(i+1)} & \cdots & B_{m-1}^{(i+1)} & 0_{N_g} & B_{m+1}^{(i+1)} & \cdots & B_K^{(i+1)} \\
0_{N_g} & \cdots & 0_{N_g} & I_{N_g} & 0_{N_g} & \cdots & 0_{N_g}
\end{bmatrix}.
\]

Define a \( 2N_g \times 2N_g \) matrix \( G_m^{(i+1)} \) as

\[
G_m^{(i+1)} := I_{2N_g} + C_m^{(i+1)} \Phi(\hat{\omega}_{m-1}^{(i+1)}) C_m^{(i+1)}
\]

Using (8.12) and the Woodbury matrix identity [144, p.50], we obtain

\[
\Phi(\hat{\omega}_m^{(i+1)}) = \Phi(\hat{\omega}_{m-1}^{(i+1)}) - \Phi(\hat{\omega}_{m-1}^{(i+1)}) C_m^{(i+1)} \left( G_m^{(i+1)} \right)^{-1} \bar{C}_m^{(i+1)} \Phi(\hat{\omega}_{m-1}^{(i+1)})
\]

(8.13)
where a matrix inversion of order $2N_g$ is required. Thus, the overall recursive computation of $\Phi(\hat{\omega}^{(i+1)})$ from $\Phi(\hat{\omega}^{(i)})$ requires inverting $K_i$ matrices of dimension $2N_g \times 2N_g$. Since inverting a $KN_g \times KN_g$ matrix (for direct computation of $\Phi(\hat{\omega}^{(i+1)})$) typically requires $O(K^3N_g^3)$ operations while inverting a $2N_g \times 2N_g$ matrix requires only $O(8N_g^3)$ operations, the saving in computational complexity is significant for relatively large $K$.

### 8.2 Further Discussion on DUFE

As mentioned in Section 8.1.2, the choice of $G(\cdot)$ is crucial for achieving fast convergence of the proposed DUFE algorithm. In this section, we discuss how to select an appropriate $G(\cdot)$, and use numerical examples to illustrate the convergence behavior of the proposed algorithm.

#### 8.2.1 Choices of $G(\cdot)$

In principle, the function $G(\cdot)$ should be selected to achieve fast convergence of the algorithm. Due to the nature of the iterative algorithm, it is difficult to quantitatively relate the choice of $G(\cdot)$ with the convergence behavior of the algorithm. The actual choice of $G(\cdot)$ depends on system design. As an example, a specific choice of $G(\cdot)$, which empirically works well and is adopted in our simulations, is given by

$$G(\varepsilon^{(i)}, \eta^{(i)}) = \alpha \varepsilon^{(i)}/\mathcal{F}(\eta^{(i)})$$

where $\alpha$ is a constant that depends on each individual system design, and $\mathcal{F}(\eta^{(i)}) = \sum_{m=1}^{K} \eta^{(i)}_m / K$ is the average error resulting from the local-search process. Obviously, we have $\varepsilon^{(i)}/\mathcal{F}(\eta^{(i)}) \geq 1$. Thus, as a general design guideline, $G(\cdot)$ can be selected to be positively proportional to $\varepsilon^{(i)}/\mathcal{F}(\eta^{(i)})$. This setting is based on the consideration that a relatively large $\varepsilon^{(i)}/\mathcal{F}(\eta^{(i)})$ implies the existence of one or more users whose CFO estimates obtained in the primitive CFO estimation (i.e., $\hat{\phi}^{(i)}_l$ for the $l$th user) are too far from their actual CFO’s, $\omega_l$. Under this circumstance, for example, if $k$th user belongs to the first group, it is highly likely that $k$th user is among those users whose CFO estimates
are far from their actual CFO’s. Thus, taking into account the fact that the local search range is quite small, a relatively large $\varepsilon^{(i)}/F(\eta^{(i)})$ means that a contribution larger than $\Delta\hat{\phi}_k^{(i)}$ is expected for filling the gap between $\omega_k$ and $\hat{\phi}_k^{(i)}$. In this case, the use of a large value of $G(\cdot)$ is justified and vice versa.

### 8.2.2 An Example Illustrating the Convergence Behavior of DUF E

The foregoing description shows that the proposed DUF E algorithm estimates CFO’s adaptively. In each iteration, the adaptation of CFO estimates is done selectively among users and is accomplished by a two-step estimation process. To better understand the adaptation mechanism involved in the DUF E algorithm, we use an illustrative example to qualitatively analyze the convergence behavior of DUF E and its dependence on the two-step adjustment.

We consider an OFDMA system with $N = 512$ subcarriers and $K = 8$ users. Fig. 8.2 shows a 2-D footprint plot of the two-step adjustments of eight users. The true CFO’s are selected as

$$\omega = \left\{\frac{2\pi}{N} \times \Delta f_k\right\}_{k=1}^{K} = \frac{2\pi}{N} \times [-0.44, -0.11, -0.07, 0.085, 0.1, 0.345, 0.4, 0.475]^T,$$

and are indicated on the $X$-axis. The grid lines parallel to the $X$-axis indicate the completion of each iteration. The black marks on the first line just above the $X$-axis correspond to the estimation results, $\hat{\omega}^{(1)}$, obtained in the first iteration, those on the second line are obtained in the second iteration, and so on. Between the $i$th and the $(i + 1)$th horizontal lines, the white marks (White$^{(i)}$) represent the CFO estimates after the primitive adjustment (Black$^{(i)}$ $\rightarrow$ White$^{(i)}$) during the $(i + 1)$th iteration, i.e., $\hat{\phi}^{(i)}$, and the gray marks (Gray$^{(i)}$) show the results right after the local search (White$^{(i)}$ $\rightarrow$ Gray$^{(i)}$, $G(\cdot) = 1$), i.e., $(\hat{\phi}_k^{(i)} + \Delta\hat{\phi}_k^{(i)})$ for the $k$th user. When a user belongs to the first group during the $(i+1)$th iteration, the corresponding adjustment Gray$^{(i)}$ $\rightarrow$ Black$^{(i+1)}$ (the black marks on the $(i + 1)$th line) is due to the function $G(\cdot)$ in (8.8a), and White$^{(i)}$ $\rightarrow$ Black$^{(i+1)}$ corresponds to the second adjustment step.

Clearly, the task in the $(i + 1)$th iteration is to perform the Black$^{(i)}$ $\rightarrow$ Black$^{(i+1)}$ adjustment. It can be observed from Fig. 8.2 that some pairs of Black$^{(i)}$ and Black$^{(i+1)}$
8.2 Further Discussion on DUFE

Figure 8.2: Illustration of the two-step adjustment of CFO estimation in each iteration of the DUFE algorithm. The system has $K = 8$ active users, $N = 512$ subcarriers, and true normalized CFO’s: $[-0.44, -0.11, -0.07, 0.085, 0.1, 0.345, 0.4, 0.475]$.

position themselves on lines parallel to the $Y$-axis. This corresponds to the case when the related users have $\Delta \hat{\phi}_k^{(i)} \hat{\omega}^{(i)} < 0$, i.e., in the second group, as we have described in Section 8.1.2. In this case, those users’ CFO estimates will not be updated. In fact, the user grouping is achieved by tracking the adjustment directions of Black$^{(i)}$ → White$^{(i)}$ and White$^{(i)}$ → Gray$^{(i)}$. Take the first user as an example. In the seventh iteration, the primitive estimation moves the estimate from Black$^{(6)}$ right to White$^{(6)}$, whereas the local search moves the estimate from White$^{(6)}$ left to Gray$^{(6)}$. As these two adjustments are in opposite directions, we can conclude that, for the first user, the primitive adjustment in this iteration is in wrong direction and is thus invalid. Indeed, this is true, as we can see from Fig. 8.2 that White$^{(6)}$ actually goes further away from the true CFO than Black$^{(6)}$ does. On the other hand, in the eighth iteration, the adjustments of Black$^{(7)}$ → White$^{(7)}$ and White$^{(7)}$ → Gray$^{(7)}$ are in the same direction (to left). In this case, the first user
8.2 Further Discussion on DUFE

is marked to be in the first group, and thus its CFO can be updated. It can be seen that both adjustments shift the estimate closer towards the true value. Thus, it is clear from this example that effectively dividing the users into two groups is key to achieving fast convergence of DUFE.

Observe from Fig. 8.2 that the adjustment from a white mark to a gray mark is always in small amounts. This is due to the fact that the local search is limited to a small range around the white mark. Such an arrangement brings in some benefits to the convergence of the DUFE algorithm as this enables us to determine if the primitive adjustment is in the correct direction. Otherwise, if the search range is large, the required approximation assumption in (8.6), \( \tilde{Z}(\tilde{\varphi}_k^{(i)}(q)) \approx \hat{Z}(\hat{\varphi}_k^{(i)}(q)) \), may become invalid. Thus, the local search is another important factor affecting the convergence of the algorithm. In addition, as we explained while describing (8.7), the local search itself saves numerous computations, and in particular, avoids the need for inversion of large matrices.

Moreover, we observe from Fig. 8.2 that, when a user is marked as being in the first group, the second adjustment (White\(^{(i)} \rightarrow\) Black\(^{(i+1)}\)) is likely to be in large amounts if the white mark is not close to the true CFO (e.g., first user in eighth iteration). This demonstrates the effectiveness of the weighting function \( G(\cdot) \), which turns out to be a key factor for achieving fast convergence. It should be pointed out that the adaptation may diverge when the second adjustment is over-weighted under certain extreme cases. In practice, this can be easily prevented by setting limits on the quantities \( G(\varepsilon^{(i)}, \eta^{(i)}) \Delta \hat{\varphi}_k^{(i)} \) in (8.8a) and \( \alpha_0 G(\varepsilon^{(i)}, \eta^{(i)}) \) in (8.9a).

It should be emphasized that, in order to prevent premature convergence to local optima in DUFE, the adaptation of estimates in (8.8) has been complemented with two special adjustments as shown in (8.9) and (8.10). By doing so, we observe steady convergence of DUFE to global optima in our extensive simulations.

8.2.3 Remarks on Joint CFO and Channel Estimation

Before ending the discussion in this section, a remark on joint CFO and channel estimation is in order. After obtaining the estimation of CFO’s, \( \hat{\omega} \), using (7.7) and (7.8),
we can obtain the ML estimate of the channel impulse response (CIR) vector $\xi$ as

$$
\hat{\xi} = [Z^H(\hat{\omega})Z(\hat{\omega})]^{-1} Z^H(\hat{\omega})y = \Phi(\hat{\omega})Z^H(\hat{\omega})y.
$$

(8.15)

Thus, the computation of $\hat{\xi}$ involves a matrix inversion. In the case of DUF$E$, the matrix inverse, $\Phi(\hat{\omega})$, is obtained in an iterative manner, and thus it is readily available for use once DUF$E$ completes its iterative estimation of CFO’s. This makes the computation of $\hat{\xi}$ simple and easy. On the contrary, carefully examining Table 7.1 and Table 7.2 shows that neither AP$F$E nor AAP$F$E can provide $\Phi(\hat{\omega})$ straightforwardly, and hence, inversion of a $KN_g \times KN_g$ matrix becomes necessary. This becomes an additional advantage of the proposed DUF$E$.

### 8.3 Performance and Complexity Comparison

In this section, we will compare by simulations the proposed DUF$E$ scheme with the AP$F$E and AAP$F$E schemes in terms of performance and computational complexity.

#### 8.3.1 Performance Evaluation

We simulate an OFDMA system which has $N$ subcarriers shared by $K$ users, i.e., the number of subcarriers per user is $N_k = N/K$. The subcarriers are randomly assigned to each user in all the simulations. We assume that the CFO of each user is uniformly distributed over the interval $[-\Delta f_{\text{max}}, \Delta f_{\text{max}}]$, where $\Delta f_{\text{max}}$ denotes the maximum permitted frequency offset normalized by the subcarrier spacing, i.e., $|\Delta f_k| \leq \Delta f_{\text{max}} < 0.5$.

The channel response of each user is generated according to the IEEE WLAN channel model with nine paths, corresponding to a delay spread of 40 ns with the signal bandwidth is 20 MHz. In particular, the channel coefficients are modelled as independent complex Gaussian random variables with zero mean and an exponential power delay
8.3 Performance and Complexity Comparison

profile

\[ E \{ |h_k(l)|^2 \} = \begin{cases} 
0, & l = 0, 1, \ldots, d_k - 1 \\
\lambda_k \lambda_0 \cdot \exp\{-l + d_k\}, & l = d_k, d_k + 1, \ldots, d_k + 8 
\end{cases} \]

where \( d_k \) is the timing error of the \( k \)th user modelled as a small random positive integer. The constant \( \lambda_0 \) is chosen such that the signal power of the \( k \)th user is normalized to unity when \( \lambda_k = 1 \). The parameters \( \lambda_k \), for \( k = 1, \ldots, K \), affect the signal power received from each user. We select various \( \lambda_k \) in the simulation in order to assess their impact on the system performance.

In our simulations, SNR is defined as \( \text{SNR(dB)} = 10 \log_{10}(P_r/\sigma^2) \), where \( P_r \) is the average signal power at the BS receiver. The length, \( N_p \), of cyclic prefix is selected to be 28, and \( N_I \) denotes the number of iterations in the case of DUFE. To quantify the performance of CFO estimation, we define the mean-square error (MSE) in the estimation of normalized CFO’s as

\[ \text{MSE} = \frac{N^2}{4\pi^2 K \chi} \sum_{\rho=1}^{\chi} \| \hat{\omega}_\rho - \omega_\rho \|^2 \]

where \( \chi \) is the total number of Monte Carlo tests and \( \hat{\omega}_\rho \) is the estimate of \( \omega_\rho \). We adopt the Cramér–Rao bound (CRB) in (7.28) as benchmark performance. As mentioned in derivation of (7.28), the CRB for the estimation of normalized CFO’s is averaged over a large number of channel realizations. In particular, we study the CRB behavior of OFDMA systems for various values of the number of subcarriers, \( N \), and the number of users, \( K \), and the results are shown in Table 8.1. Observe from Table 8.1 that, for a given SNR, the CRBs remain in the same order for different configurations of \( N \) and \( K \) as long

<table>
<thead>
<tr>
<th>CRB</th>
<th>( N = 128 )</th>
<th>( N = 256 )</th>
<th>( N = 384 )</th>
<th>( N = 512 )</th>
<th>( N = 640 )</th>
<th>( N = 768 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( N_k = 32 )</td>
<td>2.371e-4</td>
<td>2.300e-4</td>
<td>2.506e-4</td>
<td>2.522e-4</td>
<td>2.472e-4</td>
<td>2.448e-4</td>
</tr>
<tr>
<td>( N_k = 64 )</td>
<td>2.259e-5</td>
<td>2.489e-5</td>
<td>2.454e-5</td>
<td>2.495e-5</td>
<td>2.527e-5</td>
<td>2.493e-5</td>
</tr>
</tbody>
</table>
as the number of subcarriers per user, $N_k$, remains the same. When $N_k$ increases, a lower CRB can be reached and thus higher estimation accuracy can be expected.

**Example 8.1 (Convergence Performance – $K = 4$)**: In this example, we choose $N = 128$, $K = 4$, and $\Delta f_{\text{max}} = 0.32$. An important design parameter in the proposed DUF is the number of iterations needed to achieve convergence. Fig. 8.3 shows the MSE provided by DUF, APFE and AAPFE as a function of $J$, where $J = N_c$ is the number of cycles in the cases of APFE and AAPFE, and $J = N_I/K$ is the number of iterations per user in the case of DUF. The use of $J$, instead of $N_I$, in DUF is for fair comparison of convergence among various methods based on similar order of computational complexity as shall be clear from our later discussion. All users have equal power ($\lambda_k = 1$) with SNR = 22 dB. The approximation of $\|Q_k^{(i)}(\tilde{\omega}_k(q))y\|^2$ with various orders has been assumed in the case of AAPFE by setting $L = 1$ to 4 in (7.17). We observe from Fig. 8.3 that the proposed DUF requires about nine iterations ($J = 2.25$, $N_I = KJ = 9$) to converge, while both APFE and AAPFE achieve convergence in two cycles ($N_c = J = 2$). Further,

**Figure 8.3**: Convergence performance of various CFO estimators ($N = 128$, $K = 4$, $\Delta f_{\text{max}} = 0.32$, and SNR = 22 dB).
the estimation accuracies achieved by DUFE and APFE are close to the CRB. It can be also observed that the estimation accuracy of AAPFE decreases when a lower order approximation is used. In particular, even with a rather high order approximation ($L = 4$), the performance degradation of AAPFE is significant compared to DUFE and APFE.

**Example 8.2 (Convergence Performance – $K = 8$)**: In this example, we consider two scenarios. In the first scenario, we assume $N = 256$, $K = 8$, $\Delta f_{\text{max}} = 0.32$ and SNR = 22 dB. As shown in Fig. 8.4, both DUFE and APFE achieve similar estimation accuracy that is close to CRB, as in Fig. 8.3. Note that the CRBs in Fig. 8.3 and Fig. 8.4 are in the same order as we have $N_k = 32$ in both cases. We further observe that DUFE converges in about $J = 1.5$ iterations per user ($N_I = 12$), which is smaller than that in Test Example 8.1. This relatively faster convergence is due to the involvement of more users, because the function $\mathcal{F}(\eta(0))$ and the weighting factor in (8.14) that affect the convergence rate become more reliable with increase in the number of users, $K$. In the second scenario, we choose $K = 8$ and SNR = 22 dB, but increase the total number of subcarriers from $N = 256$ to $N = 512$. In addition, we assume $\Delta f_{\text{max}} = 0.48$ that spans almost the full range of possible CFO variation. As shown in Fig. 8.5, the proposed DUFE converges in about $J = 2.125$ iterations per user ($N_I = 17$). Again, both DUFE and APFE are able to achieve estimation accuracy close to the CRB (around $2.5e-5$ for $N_k = 64$) which, as expected, is one order lower than that in Figs. 8.3 and 8.4 (around $2.4e-4$ for $N_k = 32$). It should be noted that, in this case, the AAPFE, with a large value of $L$ ($L = 4$), is able to achieve an estimation accuracy similar to that of DUFE and APFE. Nevertheless, the high computational complexity involved in AAPFE in this case may prevent its use in practice as shall be clear from the complexity analysis presented later.

**Example 8.3 (Performance in the Presence of All Users With Equal Power)**: In this example, we first examine the accuracy of the proposed DUFE in the presence of $K = 4$ users with equal power ($\lambda_k = 1$). As in Test Example 8.1, we choose $N = 128$, $\Delta f_{\text{max}} = 0.32$, $N_I = 9$ and $N_c = 2$. Fig. 8.6 illustrates the MSE as a function of SNR. For comparison, the CRB and the performance curves of APFE and AAPFE are
Figure 8.4: Convergence behavior of various CFO estimators \((N = 256, K = 8, \Delta f_{\text{max}} = 0.32, \text{ and } \text{SNR} = 22 \text{ dB})\).

Figure 8.5: Convergence behavior of various CFO estimators \((N = 512, K = 8, \Delta f_{\text{max}} = 0.48, \text{ and } \text{SNR} = 22 \text{ dB})\).
Figure 8.6: Accuracy of various CFO estimators versus SNR in the presence of all users with equal power ($N = 128$, $K = 4$, and $\Delta f_{\text{max}} = 0.32$).

also included. Observe that the estimation accuracy of DUFE is close to the CRB and that of APFE at all SNR’s and is better than that of AAPFE in the high SNR regime where AAPFE exhibits a floor that worsens as $L$ decreases. This error-floor phenomenon was also observed in [39]. Fig. 8.6 also shows the performance curve of DUFE with reduced number of iterations. In this case, we set $N_I = 6$. It can be seen from Fig. 8.6 that, even with nearly half of the number of iterations, the MSE performance of DUFE is similar to that of AAPFE with $L = 4$.

Secondly, we assess the accuracy of the proposed DUFE in the presence of $K = 8$ users with equal power. As in Test Example 8.2, we consider the following two scenarios, $N = 256$ and $\Delta f_{\text{max}} = 0.32$, as well as $N = 512$ and $\Delta f_{\text{max}} = 0.48$. Based on the results obtained from the second example, we choose $N_I = 12$ and $N_I = 17$, respectively, which lead to convergence of DUFE. Fig. 8.7 illustrates the MSE as a function of SNR in both cases. For comparison, the corresponding CRB’s are also included. Observe that DUFE is able to achieve an estimation accuracy close to the respective CRB in both cases.
8.3 Performance and Complexity Comparison

Figure 8.7: Accuracy of DUFE versus SNR in the presence of all users with equal power $(N = 256, K = 8, \Delta f_{\text{max}} = 0.32; \text{ and } N = 512, K = 8, \Delta f_{\text{max}} = 0.48)$.

Example 8.4 (Resistance to the Near-Far Effect): We next investigate the resistance of various CFO estimators to the near-far effect. For this purpose, we use the simulation setup of the previous example (for $K = 4$) except that the powers of some users are now 6 dB higher than those of other users. This can be easily achieved by randomly selecting $\lambda_k = 1$ or 4 in (8.16). Fig. 8.8 shows the MSE versus SNR for DUFE, APFE, and AAPFE. Comparing Fig. 8.6 with Fig. 8.8, we observe that the effect of near-far problem on these three schemes is rather minimal. For DUFE, a similar observation has been made in the scenario where the powers of Users 1 and 2 are 6 dB higher than those of Users 3 and 4 as shown in Fig. 8.9.

8.3.2 Computational Complexity

We now evaluate the complexity of the proposed DUFE scheme, and compare that with the existing schemes. In fact, despite its high estimation accuracy, the heavy computations required by APFE for inverting a matrix of size $N_g \times N_g$ in (7.12) $N_c K N_\omega$ times and a
Figure 8.8: Accuracy of various CFO estimators versus SNR in the presence of near-far effect ($N = 128$, $K = 4$, and $\Delta f_{\text{max}} = 0.32$).

Figure 8.9: Accuracy comparison for the DUFE estimator in the presence of all users with equal power and in the presence of near-far effect with two near users (high-SNR, $\lambda_1 = \lambda_2 = 4$) and two far users (low-SNR, $\lambda_3 = \lambda_4 = 1$). $N = 128$, $K = 4$, and $\Delta f_{\text{max}} = 0.32$. 
matrix of size \((K - 1)N_g \times (K - 1)N_g\) in (7.14) \(N_cK\) times may prevent its application in practical systems. Thus, in the following, we will use only AAPFE for complexity benchmark comparison.

The comparison is divided into two parts. In the first part, we examine the matrix inversions required by both schemes. As shown in Section 8.1.3, in each iteration, DUFЕ requires to invert \(K_i\) times a \(2N_g \times 2N_g\) matrix. Note that \(K_i\) may vary from iteration to iteration. Denoting by \(K_u\) the average of \(\{K_i\}\), we approximately have \(K_u < K/2\). Based on this and the observations from simulations, we assume \(K_u = 3K/8\) for subsequent discussion. With \(N_I = JK\) iterations, DUFЕ requires a total of \(N_I K_u = 3JK^2/8\) matrix inversions of size \(2N_g \times 2N_g\). For AAPFE, following (7.14), one may find that a total of \(N_cK\) matrix inversions of size \((K - 1)N_g \times (K - 1)N_g\) are required. Clearly, for \(K > 2\), AAPFE needs to invert matrices whose dimensions linearly increase with \(K\). This is undesirable in practice, especially when the system requires to accommodate a large number of users, i.e., large \(K\). In this sense, the proposed DUFЕ is much advantageous as it requires inverting matrices with only a fixed small dimension, \(2N_g \times 2N_g\).

Since inverting a matrix of order \(Q\) is of \(O(Q^3)\) computational complexity, we have
\[
\frac{\text{matrix–inversion complexity of DUFЕ}}{\text{matrix–inversion complexity of AAPFE}} = \frac{\frac{3}{2}JK^2 \cdot 8N_g^3}{N_cK(K - 1)^3N_g^3} = \frac{3JK}{N_c(K - 1)^3}. \tag{8.17}
\]

For comparison, we choose \(N_c = 2\), and \(J = 2.25\) corresponding to the largest \(J\) value required in all our simulations. Using (8.17), Fig. 8.10 plots the ratio in percentage of matrix-inversion complexity of DUFЕ to that of AAPFE versus the number of users. Clearly, when \(K = 4\), DUFЕ requires only about half the amount of computation of AAPFE for matrix inversions. The savings become substantial as the number of users increases. Especially, when \(K \geq 8\), DUFЕ requires less than 10% of the computations required by AAPFE for matrix inversions.

We next analyze and compare the required computational loads of both schemes, excluding that required for matrix inversions. This is done by computing the total number of complex multiplications. Table 8.2 summarizes the results.

Before we proceed to the actual comparison between the total computational loads of the two schemes, we shall first discuss the choice of the related parameters, \(N_\omega\) and
Figure 8.10: The ratio in percentage of matrix-inversion complexity of DUF to that of AAPFE.

Table 8.2: Required computational load (excluding matrix inversions).

<table>
<thead>
<tr>
<th>Scheme</th>
<th>Items</th>
<th>Complex Multiplications</th>
</tr>
</thead>
<tbody>
<tr>
<td>DUF</td>
<td>Step 1</td>
<td>$N_I[K^2N_g^2 + (N + 1)KN_g + \frac{N}{2}log_2N]$</td>
</tr>
<tr>
<td></td>
<td>Step 2</td>
<td>$N_I[(N_sK^2 + (K - 1)^2)N_g^2 + (N_sNK(K + 1)N_g + N_sNK)$</td>
</tr>
<tr>
<td></td>
<td>Computation of $\Phi(\omega^{(i+1)})$</td>
<td>$N_IK_u[(4K^2 + 6K)N_g^2 + KNN_g^2]$</td>
</tr>
<tr>
<td>AAPFE</td>
<td>Eq.(7.14)</td>
<td>$N_cK[\frac{3}{2}(K - 1)^2N_g^2 + \frac{1}{2}(K - 1)N^2N_g]$</td>
</tr>
<tr>
<td></td>
<td>Eqs.(7.15),(7.16),(7.17)</td>
<td>$N_cKN_\omega[(L + 2)N_g^2 + (2L + 1)NN_g + (L + 1)N^2]$</td>
</tr>
</tbody>
</table>

$N_s$. Recall that $N_\omega$ is the number of CFO values over which the grid-search evaluation in AAPFE is performed. Thus, $N_\omega$ is a design parameter that defines the estimation resolution and is related to $\Delta f_{\text{max}}$. The parameter $N_s$ in DUF is also dependent on estimation resolution, but is unrelated to $\Delta f_{\text{max}}$. Since the local search is limited to a small range of the CFO values, we can select $N_s$ to be much smaller than $N_\omega$. As a rule
of thumb, we set $N_\omega = 200 \Delta f_{\text{max}}$ and $N_s = 10$ in all our simulations. Knowing this, we shall fix $\Delta f_{\text{max}}$ at 0.5 when calculating $N_\omega$, for fair comparison of complexity.

Furthermore, from Table 8.2, it is interesting to note that the computational complexity of Step 1 in DUFE, which also involves a grid-search (see (8.2)), is unrelated to the parameter $N_\omega$. The reason behind this is that, in Step 1, the system is treated as a single-user system, and therefore, the required grid-search operations can be efficiently performed by using fast Fourier transform (FFT) techniques, as described in [107].

Denote by $\beta(L)$ the ratio of the total computational complexity of DUFE to that of AAPFE (with $L$th order approximation). Table 8.3 lists and compares the ratios obtained for the design examples shown in Figs. 8.3, 8.4, and 8.5. From the comparison, it can be seen that, in all these cases, the required computational complexity of the proposed DUFE is less than that of even the simplest AAPFE with 1st-order approximation.

Table 8.3: Comparison of the required computational load between DUFE and AAPFE.

<table>
<thead>
<tr>
<th>CASE</th>
<th>$N$</th>
<th>$K$</th>
<th>$N_c$</th>
<th>$N_I = JK$</th>
<th>$N_\omega$</th>
<th>$N_s$</th>
<th>$K_u$</th>
<th>$\beta(1)$</th>
<th>$\beta(2)$</th>
<th>$\beta(3)$</th>
<th>$\beta(4)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>128</td>
<td>4</td>
<td>2</td>
<td>9</td>
<td>100</td>
<td>10</td>
<td>1.5</td>
<td>75%</td>
<td>55%</td>
<td>43%</td>
<td>36%</td>
</tr>
<tr>
<td>II</td>
<td>256</td>
<td>8</td>
<td>2</td>
<td>12</td>
<td>100</td>
<td>10</td>
<td>3</td>
<td>63%</td>
<td>52%</td>
<td>44%</td>
<td>38%</td>
</tr>
<tr>
<td>III</td>
<td>512</td>
<td>8</td>
<td>2</td>
<td>17</td>
<td>100</td>
<td>10</td>
<td>3</td>
<td>39%</td>
<td>31%</td>
<td>26%</td>
<td>22%</td>
</tr>
</tbody>
</table>

Before concluding the discussion on computational complexity, we present a brief comment on the processing latency involved in each iteration of DUFE. Recall from (8.5), (8.6) and (8.7) that the local search for each user can be performed independently, thanks to the approximation assumption in (8.6). This means that $\Delta \hat{\varphi}_{k}^{(e)}$ can be computed fully in parallel for all users in Step 2. Compared with AAPFE that requires a fully serial grid search (see Table 7.2), the proposed DUFE is more advantageous in handling the processing latency issue in practical implementation.
8.4 Concluding Remarks

In this chapter, we have proposed a novel iterative CFO estimation scheme for uplink OFDMA systems using GCAS. In each iteration, the proposed DUFE divides the users into two groups, in which the CFO estimates are updated differently by following a primitive CFO estimation and a local-search based CFO adjustment. The resulting CFO estimation offers accurate CFO estimates for all users in an efficient manner.

The high computational efficiency achieved by the proposed DUFE algorithm primarily arises from several aspects. First, the primitive estimation step can be efficiently implemented by FFT techniques. Second, the local-search based adjustment of CFO estimates is limited to a small range so that the annoying dense-grid-search is avoided. Third, we have proposed a scheme to decompose the inversion of a matrix, whose dimension is usually large and linearly increasing with the number of users, into a series of inversions of matrices with a much smaller and fixed dimension. Realtime inversion of large matrices is thus avoided. Overall, compared with other existing approaches, the proposed scheme strikes much better performance-complexity tradeoffs.
Chapter 9

CFO Estimation for MIMO-OFDMA Uplink Transmission

We have observed from Chapter 7 and Chapter 8 that maximum-likelihood (ML) CFO estimation in the uplink of an OFDMA system with generalized carrier-assignment scheme (GCAS) is a complex multiple-parameter estimation problem. The inclusion of multiple-input multiple-output (MIMO) processing makes the problem even more complicated. In this chapter, we propose to decompose the MIMO-OFDMA CFO estimation into a series of multiple-input single-output (MISO) ML estimations, each of which adopts the divide-and-update frequency estimator (DUFE) based iterative approach. Similar to that in single-input single-output OFDMA (SISO-OFDMA) systems, the proposed solution has affordable computational complexity while maintaining high estimation accuracy similar to the exact ML solution. Performance and complexity comparisons are provided with numerical results to illustrate the effectiveness of the proposed method.

9.1 MIMO-OFDMA Signal Model

Suppose that $K$ active users, each with $N_q$ transmit antennas, simultaneously communicate with the base station (BS), which has $N_r$ receive antennas, in the
9.1 MIMO-OFDMA Signal Model

uplink of an OFDMA system using \( N \) subcarriers. Let \( s_k^{(q)}(n) \) be the \( n \)th block of frequency-domain symbols sent from antenna \( q \) of the \( k \)th user, \( k = 1, 2, \ldots, K \) and \( q = 1, 2, \ldots, N_q \). The corresponding time-domain vector is given by \( c_k^{(q)}(n) = F_N s_k^{(q)}(n) \).

A cyclic prefix (CP) of length \( N_g \) is appended in front of \( c_k^{(q)}(n) \) for the purpose of eliminating inter-block interference (IBI). The resulting vector \( u_k^{(q)}(n) \) of length \( N_p = N + N_g \) is transmitted over the channel. Define the discrete-time baseband channel impulse response vector from transmit antenna \( q \) of the \( k \)th user to receive antenna \( r \) of BS as \( h_k^{(q,r)} = [h_k^{(q,r)}(0), h_k^{(q,r)}(1), \ldots, h_k^{(q,r)}(N_h - 1)]^T \), where \( N_h \) is a design parameter that depends on the duration of the transmit/receive filters and on the expected maximum channel delay spread.

The received discrete-time signal at receive antenna \( r \) of BS is the superposition of the signals from the transmit antennas of all users and is given by

\[
x^{(r)}(m) = \sum_{k=1}^{K} \left[ e^{j\omega_k m} \sum_{t=1}^{N_q} \sum_{l=0}^{N_h-1} h_k^{(q,r)}(l) u_k^{(q)}(m - l - \mu_k^{(q,r)}) \right] + v^{(r)}(m) \tag{9.1}
\]

where \( u_k^{(q)}(nN_t + i) = [u_k^{(q)}(n)]_i \), \( \omega_k = 2\pi\Delta f_k/N \) with \( \Delta f_k \) being the \( k \)th user’s CFO normalized by the subcarrier spacing, \( \mu_k^{(q,r)} \) is the timing error related to transmit antenna \( q \) of the \( k \)th user expressed in sampling intervals, and \( v^{(r)}(m) \) is the noise modeled as a circularly symmetric white Gaussian process with variance \( \sigma_v^2 \), with \( r = 1, 2, \ldots, N_r \). Here, we assume that all transmit–receive antenna pairs between a given user and BS have same CFO’s. The received samples \( x^{(r)}(m) \) are divided into adjacent OFDMA blocks of length \( N_p \) with the BS timing as a reference. The samples of each block are then converted into a \( N_p \times 1 \) vector, denoted by \( x^{(r)}(n) \). Finally, the CP is removed and the remaining samples are collected into a \( N \times 1 \) vector, \( y^{(r)}(n) \).

Following similar description as in Section 7.1, in the sequel, we will focus on a single training block \( (n = 0) \) and omit the temporal index \( n \) for notational simplicity. Under the assumption that \( N_g \geq N_h + \max_{k,q,r}\{\mu_k^{(q,r)}\} \), and \( |\Delta f_k| < 0.5 \), we can rearrange (9.1) as

\[
y^{(r)} = \sum_{k=1}^{K} D(\omega_k) U_k \xi_k^{(r)} + v^{(r)} \tag{9.2}
\]
where
\[
D(\omega_k) = \text{diag}(e^{j\omega_k N_g}, e^{j\omega_k (N_g+1)}, \ldots, e^{j\omega_k (N_g+N-1)})
\]
and
\[
\xi^{(r)}(k) = \left[ (\xi^{(1,r)}(k))^T, (\xi^{(2,r)}(k))^T, \ldots, (\xi^{(N_q,r)}(k))^T \right]^T
\]
and
\[
U_k = \left[ U^{(1)}_k, U^{(2)}_k, \ldots, U^{(N_q)}_k \right]
\]
with \(\xi^{(q,r)}(k) = \left[ 0_{\mu^{(q,r)}_{k} \times 1}^T,(h^{(q,r)}_k)^T \Phi^{(q,r)}_{k} - \mu^{(q,r)}_{k} - N_h \right]_1^T\) and \(U^{(q)}_k\) being a \(N \times N_g\) column-circulant matrix whose first column is \(c^{(q)}_k\).

Defining \(\omega := [\omega_1, \omega_2, \ldots, \omega_K]^T\), \(\xi^{(r)}(k) := \left[ (\xi^{(r)}_1(k))^T, (\xi^{(r)}_2(k))^T, \ldots, (\xi^{(r)}_K(k))^T \right]^T\), and \(Z(\omega) := [D(\omega_1)U_1, D(\omega_2)U_2, \ldots, D(\omega_K)U_K]\), we can rewrite (9.2) as
\[
y^{(r)}(k) = Z(\omega)\xi^{(r)}(k) + v^{(r)}(k), \quad \text{for } r = 1, 2, \ldots, N_r. \tag{9.3}
\]

It can be seen that the MIMO-OFDMA uplink transmission has been decomposed into \(N_r\) parallel MISO-OFDMA processing. For the \(r\)th MISO processing, the ML estimate of the CFO vector \(\omega, \hat{\omega}(r) := [\hat{\omega}_1(r), \hat{\omega}_2(r), \ldots, \hat{\omega}_K(r)]^T\), is given by
\[
\hat{\omega}(r) = \arg \max_{\hat{\omega}} \{ ||Z(\hat{\omega})y^{(r)}||^2 \} \tag{9.4}
\]
where \(\hat{\omega} := [\hat{\omega}_1, \hat{\omega}_2, \ldots, \hat{\omega}_K]^T\) is the trial variable representing the possible values of \(\omega\) during search, and \(Z(\hat{\omega}) := Z(\hat{\omega})\Phi(\hat{\omega})Z^H(\hat{\omega})\) with \(\Phi(\omega) = [Z^H(\omega)Z(\omega)]^{-1}\).

### 9.2 Iterative CFO Estimation

We start from obtaining the CFO estimation, \(\hat{\omega}(1)\), using the received signal \(y^{(1)}\) from the 1st MISO chain. The actual estimation follows the steps of the DUFU described in Section 8.1. As shown in Table 9.1, those steps can be repeated for the subsequent MISO chains. Note that, in Table 9.1, we have attached “\(r\)” or “\((r)\)” to some expressions to indicate that they are related to the \(r\)th MISO chain. Otherwise, those expressions have
the same meaning as they do in Section 8.1. Similar to the definition of \( \hat{\omega}^{(i)} \) in (8.1), for example, here, we define a \( K \times 1 \) vector \( \hat{\omega}^{(i)}(r) \) as

\[
\hat{\omega}^{(i)}(r) := \left[ \hat{\omega}_1^{(i)}(r), \cdots, \hat{\omega}_k^{(i)}(r), \cdots, \hat{\omega}_K^{(i)}(r) \right]^T \tag{9.5}
\]

where \( i \) denotes the iteration index and \( r \) denotes the MISO chain index. It should be noted that the estimation results obtained from the \( r \)th MISO chain, \( \hat{\omega}(r) \), shall be used as the initial estimates for processing the subsequent \( (r + 1) \)th MISO chain. In this way, the required number of iterations will decrease drastically as the receive antenna index increases. Finally, after obtaining the estimate from the \( N_r \)-th MISO chain, the CFO estimation for the MIMO-OFDMA system is given by

\[
\hat{\omega} = \frac{1}{N_r} \sum_{r=1}^{N_r} \hat{\omega}(r) \tag{9.6}
\]

which shall yield 10 \( \log_{10}(N_r) \) dB performance gain over the estimation based on a single MISO chain.

Following the discussion in Section 7.3 and referring to [39] and [143], the Cramér–Rao bound (CRB) for the estimation of \( \Delta f_k \) using the \( r \)th MISO chain is given by

\[
\text{CRB}_{\text{miso}}^{(r)}(\Delta f_k) = \frac{N^2 \sigma_v^2}{8 \pi^2} \left[ \Re \left\{ \left[ C^{(r)}(\omega) \right]^H \left( I_N - \hat{Z}(\omega) \right) C^{(r)}(\omega) \right\} \right]^{-1} k,k \tag{9.17}
\]

where, similar to that in (7.28), \( C^{(r)}(\omega) = [c_1^{(r)} \ c_2^{(r)} \ \cdots \ c_K^{(r)}] \) and \( c_k^{(r)} = D_N D(\omega_k) U_k \xi_k^{(r)} \), \( k \in \mathbb{Z}_1^K \). Correspondingly, the CRB for the MIMO estimation of \( \Delta f_k \) is given by

\[
\text{CRB}_{\text{mimo}}(\Delta f_k) = \frac{N^2 \sigma_v^2}{8 \pi^2 N_r^2} \sum_{r=1}^{N_r} \left[ \Re \left\{ \left[ C^{(r)}(\omega) \right]^H \left( I_N - \hat{Z}(\omega) \right) C^{(r)}(\omega) \right\} \right]^{-1} k,k \tag{9.18}
\]

### 9.3 Performance and Complexity Comparison

#### 9.3.1 Performance Evaluation

We simulate a MIMO-OFDMA system which has \( N = 256 \) subcarriers shared by \( K = 4 \) users. Each user has \( N_q = 2 \) transmit antennas and BS has \( N_r = 2 \) receive antennas.
Table 9.1: Description of the DUFE based CFO estimation for MIMO-OFDMA uplink.

\[
\text{INITIALIZE } \omega \text{ with } \hat{\omega}(0) = 0_{K \times 1}
\]

\[
\text{COMPUTE AND PRE-STORE } \Phi[\hat{\omega}(0)] = \{Z^H[\hat{\omega}(0)]Z[\hat{\omega}(0)]\}^{-1}
\]

\[
\text{BEGIN REPETITION for } r = 1 \text{ to } N_r \ldots \text{ } r\text{th chain}
\]

\[
\hat{\omega}^{(0)}(r) = \hat{\omega}^{(N_1[r-1])(r-1)} \quad \text{(Note: } \hat{\omega}^{(N_1[0])(0)} = \hat{\omega}(0))
\]

\[
\text{BEGIN ITERATION for } i = 0 \text{ to } N_I[r] - 1
\]

\[
\text{Step 1 : } \nonumber
\]

\[
\hat{Z}[\hat{\omega}^{(i)}(r)] = Z[\hat{\omega}^{(i)}(r)]\Phi[\hat{\omega}^{(i)}(r)]Z^H[\hat{\omega}^{(i)}(r)]
\]

\[
\delta_i(r) = \arg \max_{\delta_\omega} \{[y(r)]^T D(\hat{\delta})(r) Z[\hat{\omega}^{(i)}(r)]D^H(\hat{\delta})(r)y(r)\}
\]

\[
\varepsilon_i(r) = \|y(r) - D(\delta_i(r)) \hat{Z}[\hat{\omega}^{(i)}(r)]D^H(\delta_i(r))y(r)\|^2/\|y(r)\|^2
\]

\[
\hat{\phi}_i(r) = \hat{\omega}^{(i)}(r) + \delta_i(r)\hat{\delta}_i(r)
\]

\[
\text{Step 2 :}
\]

\[
\begin{align*}
\text{Begin Divide for } k = 1 \text{ to } K \\
\text{Begin Local Search for } q = 0 \text{ to } N_q \\
\end{align*}
\]

\[
\hat{\phi}_k^{(i)}(q) = \hat{\phi}_k^{(i)}(r) + (q - N_q)\Delta \phi;
\]

\[
\hat{Z}[(\hat{\phi}_k^{(i)}(q))] = Z[\hat{\phi}_k^{(i)}(q)]\Phi[\hat{\omega}^{(i)}(r)]Z^H[\hat{\phi}_k^{(i)}(q)]
\]

\[
\gamma_k(q) = \|y(r) - Z[\hat{\phi}_k^{(i)}(q)]y(r)\|^2/\|y(r)\|^2
\]

\[
\text{End Local Search}
\]

\[
q_k = \arg \min_q \{\gamma_k(q)\}, \quad q_k^{(i)}(r) = \gamma_k(q_k);
\]

\[
\Delta \hat{\phi}_k^{(i)}(r) = (q_k - \frac{N_q}{2})\Delta \phi
\]

\[
\text{End Divide}
\]

\[
\text{Begin Update for } k = 1 \text{ to } K
\]

\[
\text{if } \exists k \in \{1, 2, \ldots, K\} \text{ that } [\Delta \hat{\phi}_k^{(i)}(r)\delta_i^{(i)}(r)] \geq 0
\]

\[
\hat{\omega}_k^{(i+1)}(r) = \begin{cases} 
\hat{\phi}_k^{(i)}(r) + \mathcal{G}[\varepsilon_i^{(i)}(r), \eta_i^{(i)}(r)]\Delta \hat{\phi}_k^{(i)}, & \text{if } [\Delta \hat{\phi}_k^{(i)}(r)\delta_i^{(i)}(r)] \geq 0 \\
\hat{\omega}_k^{(i)}(r), & \text{otherwise}
\end{cases}
\]

\[
\text{else}
\]

\[
\hat{\omega}_k^{(i+1)}(r) = \begin{cases} 
\hat{\omega}_k^{(i)}(r) + \delta_i^{(i)}(r), & k = \arg \min_m \{\eta_m^{(i)}(r)\} \\
\hat{\omega}_k^{(i)}(r), & \text{otherwise}
\end{cases}
\]

\[
\text{End Update}
\]

\[
\text{Step 3: Compute } \Phi[\hat{\omega}^{(i+1)}(r)] \text{ using } \Phi[\hat{\omega}^{(i)}(r)] \text{ and } \hat{\omega}^{(i+1)}(r)
\]

\[
\text{END ITERATION}
\]

\[
\hat{\omega}(r) = \hat{\omega}^{(N_I[r])(r)}
\]

\[
\text{END REPETITION}
\]

\[
\hat{\omega} = \frac{1}{N_I} \sum_{r=1}^{N_r} \hat{\omega}(r)
\]
9.3 Performance and Complexity Comparison

We assume \( N_g = 28 \) and that the channel response of each user is generated according to the IEEE WLAN channel model with nine paths, corresponding to a delay spread of 40 ns with the signal bandwidth is 20 MHz. In particular, the channel coefficients are modelled as independent complex-valued Gaussian random variables with zero mean and an exponential power delay profile

\[
E \left\{ |h_k^{(q,r)}(l)|^2 \right\} = \lambda_0 \cdot \exp \{-l\}, \quad l = 0, 1, \cdots, 8, \quad t = 1, \cdots, N_q, \quad r = 1, \cdots, N_r. \tag{9.19}
\]

The constant \( \lambda_0 \) is chosen such that the total signal power of each user is normalized to unity.

We assume that the normalized CFO of each user, \( \Delta f_k \), is uniformly distributed over the interval \([-\Delta f_{\text{max}}, \Delta f_{\text{max}}]\) with \( \Delta f_{\text{max}} = 0.32 \), and the timing error, \( \mu_k^{(q,r)} \), is modelled as a small random positive integer. The subcarriers are randomly assigned to each user in all the simulations.

**Example 9.1 (Convergence Performance of MIMO CFO Estimators)**: Similar to that in a SISO OFDMA system, an important design parameter for using DUFE in a MIMO OFDMA system is the number of iterations \( N_I[r] \) needed to achieve convergence. Again, here and hereafter, we use “\([r]\)” to indicate that the related expression is for the \( r \)th MISO chain. Fig. 9.1 shows the minimum mean square (MSE) of the normalized CFO estimates provided by DUFE and APFE as a function of \( J \), where \( J = N_c \) is the number of cycles in the case of APFE, and \( J = N_I/K \) is the number of iterations per user in the case of DUFE. All users have equal power with SNR = 22 dB. We observe that DUFE requires about 8 iterations \((J_1 = 2, \quad N_I[1] = KJ_1 = 8)\) to converge based on the 1st MISO chain and 5 additional iterations \((J_2 = 1.25, \quad N_I[2] = KJ_2 = 5)\) to converge after switching to the 2nd MISO chain. Note that \( J_1 = 2 \) and \( J_2 = 1.25 \) correspond to the variation of \( J \) from 0 to 2 and 3 to 4.25, respectively, in Fig. 9.1. The APFE, on the other hand, achieves convergence in two cycles \((N_c[1] = J_1 = 2)\) based on the 1st MISO chain and requires one more iteration \((N_c[2] = J_2 = 1)\) to converge with the 2nd MISO chain. The proposed DUFE can achieve estimation accuracy close to that of APFE. No significant gains are observed with \( \{N_I[1] > 8, N_c[1] > 2\} \) in case of the 1st MISO chain.

![Convergence performance of DUFE and APFE for 2x1 MISO and 2x2 MIMO CFO estimations (N = 256, K = 4, $\Delta f_{\text{max}} = 0.32$, and SNR = 22 dB).](image)

**Example 9.2 (MSE Performance)**: In this experiment, we assess the accuracy of the DUFE based MIMO CFO estimator. Fig. 9.2 shows the MSE of the normalized CFO’s as a function of SNR. For benchmark comparison, the CRB curve given by (9.18) is also included. It can be seen that DUFE is able to achieve an estimation accuracy close to the CRB, and that the MIMO solution results in about 3 dB MSE performance gain over the MISO based estimation.

### 9.3.2 Computational Complexity

We now evaluate the complexity of the DUFE based MIMO CFO estimation scheme in comparison with the APFE method. The comparison can be divided into two parts. In the
9.3 Performance and Complexity Comparison

Figure 9.2: Accuracy of the proposed DUFE versus SNR in the presence of all users with equal power in a MIMO OFDMA system.

In the first part, we examine the matrix inversions required by both schemes. We first consider the estimation using the $r$th MISO chain. As shown in Section 8.1.3, in each iteration, DUFE requires to invert $K_i$ times a matrix of dimension $2N_qN_g \times 2N_qN_g$. Note that $K_i$ may vary from iteration to iteration. Denoting by $K_u$ the average of $\{K_i\}$, we have approximately $K_u < K/2$. Based on this and the observations from simulations, we assume $K_u = 3K/8$ for subsequent discussion. With $N_I[r] = J_rK$ iterations, DUFE requires a total of $N_I[r]K_u = \frac{3}{8}J_rK^2$ matrix inversions of size $2N_qN_g \times 2N_qN_g$. For the APFE, one may find that a total of $N_c[r]KN_\omega$ matrix inversions of size $N_qN_g \times N_qN_g$ and $N_c[r]K$ matrix inversions of size $(K - 1)N_qN_g \times (K - 1)N_qN_g$ are required. Following the discussion in Section 8.3.2, one may find that the implementation of matrix inversions involved in APFE is practically almost infeasible as $N_\omega$ is considerably large. Thus, we also use AAPFE, instead of APFE, for complexity comparison in this case. By using AAPFE, while the matrix inversions of size $N_qN_g \times N_qN_g$ can be avoided, the matrix inversions of size $(K - 1)N_qN_g \times (K - 1)N_qN_g$ are still necessary. The AAPFE method suffers considerable performance degradation, and, for $K > 2$, AAPFE needs to invert
matrices whose dimensions linearly increase with $K$. This is undesirable in practice, especially when the system requires to accommodate a large number of users, i.e., large $K$. In this sense, the DUFE based solution is much advantageous as it requires inverting matrices with only a fixed small size, $2N_qN_g \times 2N_qN_g$.

Since inverting a matrix of order $Q$ is of $\mathcal{O}(Q^3)$ computational complexity, for overall MIMO CFO estimation, we have

$$
\rho(K) = \frac{\text{matrix–inversion complexity of DUFE MIMO CFO estimation}}{\text{matrix–inversion complexity of AAPFE MIMO CFO estimation}} \times 100% 
$$

$$
= \frac{3}{8} \frac{\sum_{r=1}^{N_r} J_r K^2 \cdot 8N_q^2 N_g^3}{\sum_{r=1}^{N_r} N_c[r] K (K-1)^3 N_q^3 N_g^3} = \frac{3K \sum_{r=1}^{N_r} J_r}{(K-1)^3 \sum_{r=1}^{N_r} N_c[r]}. 
$$

(9.20)

For comparison, we can set $N_c[1] = 2$, $N_c[2] = 1$, $J_1 = 2$ and $J_2 = 1.25$ which correspond to the $N_c[r]$ and $J_r$ values obtained in the above simulations. Fig. 9.3 plots $\rho(K)$ from (9.20) for various $K$. Clearly, when $K = 4$, DUFE requires less than half of the computations of AAPFE for matrix inversions. The computational saving becomes

![Figure 9.3: The ratio (in percentage) of matrix-inversion complexity of the DUFE based $2 \times 2$ MIMO CFO estimator to that of the AAPFE based $2 \times 2$ MIMO CFO estimator, obtained using (9.20).]
9.3 Performance and Complexity Comparison

substantial as the number of users increases. Especially, when $K \geq 8$, the DUFE based method requires less than 10% of the computations required by the AAPFE based method for matrix inversions.

Next, we compare the computational complexities of the DUFE based and the AAPFE based MIMO CFO estimators, excluding that required for matrix inversions. The evaluation is performed by calculating the total number of complex multiplications. Table 9.2 summarizes the results.

Table 9.2: Computational complexities of the DUFE based and the AAPFE based MIMO CFO estimators (excluding matrix inversions) for $N_q \times N_r$ MIMO OFDMA system.

<table>
<thead>
<tr>
<th>Scheme</th>
<th>Items</th>
<th>Complex Multiplications</th>
</tr>
</thead>
<tbody>
<tr>
<td>DUFE</td>
<td>Step 1 $\sum_{r=1}^{N_r} N_{I}[r][K^2(N_qN_g)^2 + (N + 1)KN_qN_g + \frac{N^2}{2}\log_2 N]$</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Step 2 $\sum_{r=1}^{N_r} N_{I}[r][(N_qK^2 + (K - 1)^2)(N_qN_g)^2 + (N_qNK(K + 1) + 2N(K - 1))N_qN_g + N_qNKN]^{\frac{1}{2}}$</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Step 3 $K_u \sum_{r=1}^{N_r} N_{I}[r][4K^2 + 6KN_qN_g]^{\frac{1}{2}}$</td>
<td></td>
</tr>
<tr>
<td>AAPFE</td>
<td>Eq.(7.14) $\frac{K}{2}(K - 1)^2N(N_qN_g)^2 + \frac{1}{2}(K - 1)N^2N_qN_g$</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Eqs.(7.15), (7.16),(7.17) $KN_\omega \sum_{r=1}^{N_r} N_c[r][(L + 2)(N_qN_g)^2 + (2L + 1)NN_qN_g^{(L + 1)N^2}]$</td>
<td></td>
</tr>
</tbody>
</table>

Denote by $\beta(L)$ the ratio of the total computational complexity (excluding matrix inversions) of the DUFE based MIMO CFO estimator to that of the AAPFE based MIMO CFO estimator (with $L$th order approximation). Table 9.3 lists and compares the ratios obtained for the above design example. From the comparison, it can be seen that, for different values of $L$, the required computational complexity of the proposed DUFE based MIMO CFO estimator is comparable to or better than that of the AAPFE based MIMO CFO estimator.
9.4 Concluding Remarks

In this chapter, we have extended the DUFE scheme introduced in Chapter 8 to the GCAS based uplink MIMO-OFDMA system. We have shown that the MIMO-OFDMA CFO estimation can be decomposed into a series of MISO ML estimations. Similar to that in uplink SISO-OFDMA systems, in each iteration of a MISO chain processing, DUFE divides the users into two groups, in which the CFO estimates are updated differently by following a primitive CFO estimation and a local-search based CFO adjustment. The resulting CFO estimation offers accurate CFO estimates for all users in an efficient manner. In particular, by decomposing the inversion of a matrix, whose dimension is usually large and linearly increasing with both the number of users and the number of transmit antennas, into a series of inversions of matrices with a much smaller and fixed dimension, realtime inversion of large matrices is avoided. Overall, compared with existing approaches, the proposed scheme strikes much better performance-complexity tradeoffs.

Table 9.3: Ratio of the total computational complexities (excluding matrix inversions) of the DUFE based and the AAPFE based $2 \times 2$ MIMO CFO estimators ($N = 256$ and $K = 4$).

<table>
<thead>
<tr>
<th>Items</th>
<th>$N_c[r]$</th>
<th>$N_I[r]=J_rK$</th>
<th>$N_\omega$</th>
<th>$N_s$</th>
<th>$K_u$</th>
<th>$\beta(1)$</th>
<th>$\beta(2)$</th>
<th>$\beta(3)$</th>
<th>$\beta(4)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1st MISO Chain</td>
<td>2</td>
<td>8</td>
<td>100</td>
<td>10</td>
<td>1.5</td>
<td>91%</td>
<td>71%</td>
<td>58%</td>
<td>49%</td>
</tr>
<tr>
<td>2nd MISO Chain</td>
<td>1</td>
<td>5</td>
<td>100</td>
<td>10</td>
<td>1.5</td>
<td>114%</td>
<td>89%</td>
<td>73%</td>
<td>62%</td>
</tr>
<tr>
<td>$2 \times 2$ MIMO</td>
<td>/</td>
<td>/</td>
<td>/</td>
<td>/</td>
<td>/</td>
<td>99%</td>
<td>77%</td>
<td>63%</td>
<td>54%</td>
</tr>
</tbody>
</table>
Chapter 10

Conclusions

Development of robust and high-performance channel estimation and synchronization algorithms are known to be critically important for orthogonal frequency division multiplexing (OFDM) and OFDM multiple-access (OFDMA) systems. Applying existing algorithms to practical systems raises a lot of issues to OFDM and OFDMA system design. In this thesis, some critical issues associated with the development of OFDM and OFDMA systems have been investigated. The investigation has resulted in several innovative solutions, which are summarized as follows.

10.1 Thesis Summary

In the first part of this thesis, the maximum-likelihood (ML) solution for channel estimation in OFDM systems is addressed. The performance, in terms of mean-squared error (MSE), of the conventional ML estimator (MLE) has been analyzed and shown to be linearly related to the effective length of channel impulse response (ELCIR). From this, we concluded that tracking the variation in ELCIR is very important for conventional ML estimation to achieve optimum MSE performance. But, incorporating a real-time update of ELCIR into the MLE is computationally expensive. Therefore, we have proposed a modification to MLE, which systematically combines ML estimation with a frequency-domain smoothing technique. The proposed modification has been
presented in two forms, namely, optimum-smooth MLE (OMLE) and iterative-smooth MLE (IMLE).

We have shown that the proposed OMLE and IMLE are advantageous over the conventional MLE using theoretical analyses and simulations. The performance enhancement arises from two aspects. First, by introducing frequency-domain smoothing in MLE, the proposed methods perform similar to a conventional MLE with perfect knowledge of ELCIR. The fact that the proposed methods can be blind to variation in ELCIR makes their implementation simple and robust. Second, the simple and yet systematic combination of frequency-domain smoothing and MLE has proved to be very effective for reducing both channel noise and decision-directed (DD) detection errors. This makes the proposed methods promising candidates for performing robust, low-complexity and high-performance channel estimation under time-varying channel environments.

In the second part of this thesis, we have proposed a novel channel estimation and phase error mitigation solution for multi-band OFDM-based UWB systems. The solution has been shown to be practically effective and efficient in terms of performance and implementation complexity. The channel estimator is based on the least-square (LS) algorithm, but enhanced with a multi-stage procedure using a simplified, channel frequency response (CFR) weighted DD detection process as well as a frequency-domain smoothing process. Both analytical and numerical results show that the proposed scheme achieves performance comparable to that of the more sophisticated but practically infeasible ML solution, and it outperforms the conventional LS solution with about 4.3 ∼ 5.8 dB gain in terms of MSE performance and about 1 dB gain in terms of frame error rate (FER) performance under various highly noisy UWB channel environments.

The proposed phase error mitigation scheme for OFDM-UWB applications decomposes the phase error into the related sampling frequency offset (SFO) and common phase error (CPE), which are suppressed separately. The approaches proposed for estimation of SFO and CPE are pilot-tone based and CFR-weighted with the virtues of simplicity and ease of implementation. Each of them is improved by using a unique,
robust and yet simple error suppression technique. In particular, we have shown that, in the presence of both residual carrier frequency offset (CFO) and phase noise (PHN), the intra-OFDM-symbol CPE estimator can be significantly improved by employing an inter-OFDM-symbol smoothing technique for achieving further suppression of estimation error. For this, our analysis shows that the inter-OFDM-symbol smoothing with 1st-order filtering is preferable over that with 2nd-order filtering in practical implementation of OFDM-UWB devices. In addition, SFO compensation is performed jointly in time-domain and frequency-domain. This avoids the need for a time-domain interpolator which becomes expensive in silicon implementation of UWB devices. It has been shown that, with the proposed phase-error suppression technique, the overall system can achieve FER performance close to that of the system with no phase error even in the very low SNR regime.

In the third part of this thesis, we have devoted our effort to ML approaches for joint estimation of CFO, timing error, and channel impulse response of each active user in both single-input single-output (SISO) and multiple-input multiple-output (MIMO) OFDMA systems. We have investigated ML CFO estimation for OFDMA uplink transmission with generalized carrier-assignment scheme (GCAS), which is believed to be the most challenging task in OFDMA applications. We have developed a new approach, called as divide-and-update frequency estimator (DUF E), which outperforms the existing alternating-projection frequency estimator (APFE) and approximate APFE (AAPFE), in the sense that our DUF E approach has the lowest computational complexity while maintaining the high estimation accuracy feature of ML solutions. We have achieved this by decomposing the practically almost infeasible dense grid-search required in the APFE into several less computationally intensive iterations and transforming the inverse of a large matrix into a series of matrix inversions of small dimensions using the Woodbury matrix identity. Performance and complexity comparisons with comprehensive numerical results show that the proposed DUF E method is effective for CFO estimation in the GCAS based OFDMA uplink with good performance-complexity tradeoffs.
10.2 Directions for Future Work

Some topics of interest that may be considered for further research are listed below.

First, more considerations may be given to channel estimation for OFDM systems under time-varying channel environments. In Chapter 3 of this thesis, we have demonstrated that the performance improvement obtained by using our modified ML channel estimators under time-varying channel environments is quite promising. In particular, we have assumed that the modified MLE’s perform CFR estimation on a symbol-by-symbol basis, thereby making them suitable for use in fast time-varying channel environments. Combining the CFR estimates obtained from a number of consecutive OFDM symbols would benefit further reduction of estimation errors, under slowly time-varying channel environments. The resulting new algorithm may be required to continuously sense the variation of channels so that optimum inter-symbol estimation can be achieved.

Another topic of interest is related to extending our channel estimation and phase error mitigation schemes presented in this thesis to MIMO based wireless applications. In contrast with most related work in the literature, our proposed channel estimators, including the modified MLE solutions, for OFDM based wireless applications are based only on the assumption that subcarrier spacing is less than half of channel coherent bandwidth. Also, our phase error suppression algorithms are just based on the assumption that a limited number of pilot subcarriers are available in each OFDM symbol. In most practical MIMO-OFDM systems, such requirements can be easily satisfied. Thus, after certain fine-tuning and/or improvement, both the MLE and phase error suppression solutions proposed in this thesis can be envisaged for use in MIMO-OFDM based wireless communication systems, e.g., the IEEE 802.11n high throughput wireless local area network (WLAN) systems [28].

Finally, in our proposed DUFE scheme for CFO estimation in GCAS based OFDMA uplink transmission, inverting matrices of order $2N_g$ is still required ($N_g$ is the length of cyclic prefix of each OFDMA symbol). In particular, in a $N_q \times N_r$ MIMO-OFDMA system, inverting matrices of order $2N_q N_g \times 2N_q N_g$ is necessary. Future study on this
subject for exploring new solutions with no hardware unfriendly matrix inversions may be worthwhile. Replacing matrix inversions with certain approximation operations, for example, may be considered as one direction of future study on this issue. Furthermore, in our proposed CFO estimation solution for the $N_q \times N_r$ MIMO OFDMA uplink, the estimation results obtained from $N_r$ multiple-input single–output (MISO) processing chains are combined just by a simple averaging step. To obtain the optimum ML CFO estimation for MIMO-OFDMA uplink transmission, one may conduct further investigation on more sophisticated combination methods so that the receiver diversity of MIMO systems can be fully utilized in this case.
Bibliography


List of Publications

Journal Papers (accepted/submitted)


Conference Papers (published)


Appendix A

Derivation of Optimum $\alpha_h$

Assuming that the CFR, transmitted signal and additive white Gaussian noise are independent of each other, from (5.1) and (5.2), we have

$$E\{|\hat{h}_r^{(1)}(k)|^2\} = E_0 + \frac{1}{M_1}\sigma_r^2 = E_0 + E_0/(M_1 \cdot \text{SNR}_r) \tag{A.1}$$

$$E\{\hat{h}_r^{(1)}(k)\hat{h}_r^{*}(k)\} = E\{h_r(k)[\hat{h}_r^{(1)}(k)]^*\} = E_0 \tag{A.2}$$

$$E\{\hat{h}_r^{(1)}(k \pm \Delta_k)[\hat{h}_r^{(1)}(k)]^*\} = E\{h_r(k \pm \Delta_k)h_r^{*}(k)\} = \text{corr}(\pm \Delta_k)E_0, \; \Delta_k \in \mathbb{Z}_1^2 \tag{A.3}$$

and

$$E\{\hat{h}_r^{(1)}(k \pm 1)h_r^{*}(k)\} = E\{h_r(k)[\hat{h}_r^{(1)}(k \mp 1)]^*\} = \text{corr}(\pm 1)E_0 \tag{A.4}$$

$k \in \mathbb{Z}_0^{N-1}$. Using (A.1) to (A.4) and the property that $\text{corr}(\Delta_k) = \text{corr}^*(- \Delta_k)$ [136], we can derive $\text{MSE}_r^{(2)}$ from (5.2) and (5.3) as

$$\text{MSE}_r^{(2)} = \frac{E\{|\hat{h}_r^{(2)}(k) - h_r(k)|^2\}}{E_0} = \frac{(6 - 2\Re[4\text{corr}(1) - \text{corr}(2)])\alpha_h^2 + (6\alpha_h^2 - 4\alpha_h + 1)}{M_1 \cdot \text{SNR}_r}. \tag{A.5}$$

Next, we minimize $\text{MSE}_r^{(2)}$ with respect to $\alpha_h$. Setting to zero the gradient of $\text{MSE}_r^{(2)}$ with respect to $\alpha_h$, we get

$$\frac{\partial \text{MSE}_r^{(2)}}{\partial \alpha_h} = 2(6 - 2\Re[4\text{corr}(1) - \text{corr}(2)])\alpha_h + (12\alpha_h - 4)/(M_1 \cdot \text{SNR}_r) = 0. \tag{A.6}$$

From (A.6), it is straightforward to obtain (5.5).
Appendix B

Derivation of $P_{eb}^u$ and $C_{24}$

B.1 Derivation of $P_{eb}^u$

In this appendix, we derive the approximate upper bound of average bit error probability of the proposed CFR-weighted detector, $P_{eb}^u$. After the first stage of channel estimation, $\hat{h}_r^{(2)}(k)$, given by (5.3) can be expressed as

$$\hat{h}_r^{(2)}(k) = h_r(k) + \bar{e}_r(k)$$  \hspace{1cm} (B.1)

where the estimation error term, $\bar{e}_r(k)$, is a complex normal random variable with mean zero and variance $\bar{\sigma}_r^2 = E_0/(M_1 P_{mse}^t SNR_r)$. We note that $\bar{e}_r(k)$ is uncorrelated with $h_r(k)$ and $s_m^{(i)}(k)$, but is correlated with $\hat{h}_r^{(2)}(k)$. Referring to the discussion in [137, Section II-B], we can decompose $\bar{e}_r(k)$ into two uncorrelated terms as $\bar{e}_r(k) = \hat{e}_r(k) + \tilde{e}_r(k) = \hat{h}_r^{(2)}(k)\bar{\sigma}_r^2/E_0 + \bar{e}_r(k)$. Here, $\hat{e}_r(k)$ is a complex normal random variable with mean zero and variance $\hat{\sigma}_r^2 = E_0\bar{\sigma}_r^2/(E_0 + \bar{\sigma}_r^2)$, which is uncorrelated with $\hat{h}_r^{(2)}(k)$. Thus, (B.1) can be rewritten as

$$h_r(k) = (1 - \hat{\sigma}_r^2/E_0)\hat{h}_r^{(2)}(k) - \bar{e}_r(k).$$  \hspace{1cm} (B.2)

From (5.9), (5.1), (4.3) and (B.2), we have

$$z_m^{(i)}(k) + z_m^{(i)}(N-k) = (1 - \hat{\sigma}_r^2/E_0)|\hat{h}_r^{(2)}(k)|^2 s_m^{(i)}(k) - \hat{h}_r^{(2)}(k)s_m^{(i)}(k)\bar{e}_r(k)$$

$$+ \hat{h}_r^{(2)}(k)\phi_m^{(i)}(k) + (1 - \hat{\sigma}_r^2/E_0)|\hat{h}_r^{(2)}(N-k)|^2 [s_m^{(i)}(k)]^*$$

$$- \hat{h}_r^{(2)}(N-k)[s_m^{(i)}(k)]^*\bar{e}_r(N-k) + \hat{h}_r^{(2)}(N-k)\phi_m^{(i)}(N-k)$$

147
Derivation of $P^\text{ub}_e$

$k \in \mathbb{Z}_1^{R/2}$ and $k \notin \{p(l)\}_{l=0}^{P-1}$, $i \in \mathbb{Z}_0^5$, $r = |i|_3 + 1$ and $m \in \mathbb{Z}_1^2$. Let $r_0 = |2q|_3 + 1$ and $r_1 = |2q + 1|_3 + 1$ for $q \in \mathbb{Z}_0^2$ in the following. Obviously, $z_m^{(2q)}(k) + z_m^{(2q)}(N-k) + z_m^{(2q+1)}(k) + z_m^{(2q+1)}(N-k)$ is Gaussian distributed for given $s_m^{(2q)}(k)$, $s_m^{(2q+1)}(k)$, $\hat{h}_{r_0}(k)$, $\hat{h}_{r_1}(k)$, $\hat{h}_{r_0}(N-k)$ and $\hat{h}_{r_1}(N-k)$. Following the fact that $|s_m^{(i)}(k)|^2 = 1$, $E\{v_m^{(i)}(k)\} = 0$, $E\{v_m^{(i)}(k)^2\} = 0$, $E\{| \hat{e}_r(k) \|^2\} = 0$, $s_m^{(2q+1)}(k) = s_m^{(2q)}(k)$ and $\Re(x) = \frac{1}{2}(x + x^*)$, it is straightforward to show that the conditional mean and variance of $\Re[z_m^{(2q)}(k) + z_m^{(2q)}(N-k) + z_m^{(2q+1)}(k) + z_m^{(2q+1)}(N-k)]$ are given by

\[
E\{\Re[z_m^{(2q)}(k) + z_m^{(2q)}(N-k) + z_m^{(2q+1)}(k) + z_m^{(2q+1)}(N-k)]
\mid s_m^{(2q)}(k), \hat{h}_{r_0}(k), \hat{h}_{r_1}(k), \hat{h}_{r_0}(N-k), \hat{h}_{r_1}(N-k)\}
= (1 - \bar{\sigma}_r^2 / E_0) \left[ |\hat{h}_{r_0}(k)|^2 + |\hat{h}_{r_0}(N-k)|^2 + |\hat{h}_{r_1}(k)|^2 + |\hat{h}_{r_1}(N-k)|^2 \right] \Re[s_m^{(2q)}(k)]
\]

and

\[
\text{Var}\{\Re[z_m^{(2q)}(k) + z_m^{(2q)}(N-k) + z_m^{(2q+1)}(k) + z_m^{(2q+1)}(N-k)]
\mid s_m^{(2q)}(k), \hat{h}_{r_0}(k), \hat{h}_{r_1}(k), \hat{h}_{r_0}(N-k), \hat{h}_{r_1}(N-k)\}
= \frac{1}{2} (\bar{\sigma}_r^2 + \bar{\sigma}_r^2) \left[ |\hat{h}_{r_0}(k)|^2 + |\hat{h}_{r_0}(N-k)|^2 + |\hat{h}_{r_1}(k)|^2 + |\hat{h}_{r_1}(N-k)|^2 \right],
\]

respectively. Therefore, we get [139]

\[
\Pr\{\Re[z_m^{(2q)}(k) + z_m^{(2q)}(N-k) + z_m^{(2q+1)}(k) + z_m^{(2q+1)}(N-k)] < 0
\mid \Re[s_m^{(2q)}(k)] = c, \hat{h}_{r_0}(k), \hat{h}_{r_1}(k), \hat{h}_{r_0}(N-k), \hat{h}_{r_1}(N-k)\}
= Q\left(\sqrt{(1 - \bar{\sigma}_r^2 / E_0)^2 \left[ |\hat{h}_{r_0}(k)|^2 + |\hat{h}_{r_0}(N-k)|^2 + |\hat{h}_{r_1}(k)|^2 + |\hat{h}_{r_1}(N-k)|^2 \right]} \sigma_r^2 + \bar{\sigma}_r^2 \right)
\approx Q\left(\sqrt{|h_{r_0}(k)|^2 + |h_{r_0}(N-k)|^2 + |h_{r_1}(k)|^2 + |h_{r_1}(N-k)|^2} \sigma_r^2 + \bar{\sigma}_r^2 \right).
\]

(B.3)

The approximation in (B.3) follows from (B.2) since

\[
\hat{h}_r^{(2)}(k) \approx h_r(k) / (1 - \bar{\sigma}_r^2 / E_0).
\]

(B.4)
From Section 4.2, one can find that \(|h_r(k)|\) corresponds to the shadowing factor \(X\) and thus is log-normal distributed, i.e., \(20 \log_{10} |h_r(k)| \sim \mathcal{N}(0, \sigma^2)\). Since \(|h_r(k)|\) and \(|h_r(N-k)|\) are not independent, \(\Pr \{ \mathcal{R}[z_m^{(2q)}(k) + z_m^{(2q)}(N-k) + z_m^{(2q+1)}(k) + z_m^{(2q+1)}(N-k)] < 0 \} \) is upper-bounded under the assumption that \(|h_r_v(k)| = |h_r_v(N-k)| = |h_r_1(k)| = |h_r_1(N-k)|\) (without both frequency-domain and time-domain diversities), i.e.,

\[
\Pr \left\{ u_m^{(i)}(k) = -1 \mid \mathcal{R}[s_m^{(i)}(k)] = c \right\} = \Pr \left\{ \mathcal{R}[z_m^{(2q)}(k) + z_m^{(2q)}(N-k) + z_m^{(2q+1)}(k) + z_m^{(2q+1)}(N-k)] < 0 \mid \mathcal{R}[s_m^{(2q)}(k)] = c \right\} \leq \frac{1}{\sqrt{2\pi \sigma_x}} \int_{-\infty}^{\infty} Q(\alpha_p \cdot 10^{\frac{\mathcal{R}[s_m^{(i)}(k)]}{\sigma}}) e^{-\frac{x^2}{2\sigma^2}} dx \tag{B.5}
\]

where \(\alpha_p = \sqrt{\frac{4}{\sigma_x^2 + \sigma_y^2}} = 2 \sqrt{\frac{(M_1 R_{\text{RMS}}^2 + 1) SNR}{(M_1 R_{\text{RMS}}^2 + 1) SNR + 1 E_0}}\). Assuming that \(\mathcal{R}[s_m^{(i)}(k)]\) is equiprobably \(\pm c\), we have

\[
\Pr \left\{ u_m^{(i)}(k) \mathcal{R}[s_m^{(i)}(k)] < 0 \right\} = \Pr \left\{ u_m^{(i)}(k) = -1 \mid \mathcal{R}[s_m^{(i)}(k)] = c \right\} = \Pr \left\{ u_m^{(i)}(k) = 1 \mid \mathcal{R}[s_m^{(i)}(k)] = -c \right\}. \tag{B.6}
\]

A similar procedure can be applied for the derivation of \(\Pr \{ v_m^{(i)}(k) \mathcal{R}[s_m^{(i)}(k)] < 0 \}\), and we have

\[
P_e = \Pr \left\{ u_m^{(i)}(k) \mathcal{R}[s_m^{(i)}(k)] < 0 \right\} = \Pr \left\{ v_m^{(i)}(k) \mathcal{R}[s_m^{(i)}(k)] < 0 \right\} \tag{B.7}
\]

which is approximately upper-bounded by \(P_{\text{ub}}\) as shown in (5.15).

### B.2 Derivation of \(C_{24}\)

Following the definition of \(C_{24}\) in Section 5.4, we have

\[
C_{24} = \frac{\mathbb{E}\{(\hat{h}_r^{(4)} - h_r)(\hat{h}_r^{(2)} - h_r)^2\}}{\mathbb{E}\{|h_r|^2\}} = \frac{\mathbb{E}\{|\hat{h}_r^{(4)}(k) - h_r(k)|[\hat{h}_r^{(2)}(k) - h_r(k)]^2\}}{E_0}, \quad k \in \mathbb{Z}_1^{R/2} \cup \mathbb{Z}_{N-R/2}^{N-1}. \tag{B.8}
\]

From (5.2) and (5.3), we have

\[
\hat{h}_r^{(2)}(k) - h_r(k) = \alpha_h [h_r(k-1) + h_r(k+1)] - 2\alpha_h h_r(k) + \zeta_1, \quad k \in \mathbb{Z}_1^{R/2} \cup \mathbb{Z}_{N-R/2}^{N-1} \tag{B.9}
\]
where \( \zeta_1 \) is a zero-mean complex Gaussian random variable comprising the channel noise from the first stage of channel estimation. From (5.11) it follows that
\[
\hat{h}_r^{(4)}(k) - h_r(k) = \beta_h \left[ \hat{h}_r^{(3)}(k - 1) + \hat{h}_r^{(3)}(k + 1) \right] + (1 - 2\beta_h) \hat{h}_r^{(3)}(k) - h_r(k), \quad k \in \mathbb{Z}_1^R \cup \mathbb{Z}_N^{N-1} - \mathbb{Z}_R^{R/2}. \tag{B.10}
\]

Next, substituting (5.1) into (5.10) and neglecting the specific arrangement on pilot tones, we obtain
\[
\hat{h}_r^{(3)}(k) = \frac{C}{M^2} \sum_{m=1}^{2} \sum_{q=r-1}^{q=r+2} \left\{ h_r(k) s_m^{(q)}(k) \left[ u_m^{(q)}(k) - j v_m^{(q)}(k) \right] \right\} + \zeta_2, \quad k \in \mathbb{Z}_1^R \cup \mathbb{Z}_N^{N-1} - \mathbb{Z}_R^{R/2} \tag{B.11}
\]
where the noise term \( \zeta_2 = \frac{C}{M^2} \sum_{m=1}^{2} \sum_{q=r-1}^{q=r+2} \left\{ v_m^{(q)}(k) \left[ u_m^{(q)}(k) - j v_m^{(q)}(k) \right] \right\} \) with \( \text{E}\{\zeta_2 h_r(k)\} = 0 \) and \( \text{E}\{\zeta_1 \zeta_2\} = 0 \). Thus, from (B.11), it is straightforward to obtain
\[
\text{E}\{\hat{h}_r^{(3)}(k \pm \Delta_k) h_r^*(k)\} = \begin{cases} (1 - 2P_e) E_0, & \Delta_k = 0 \\ \text{corr}(\pm \Delta_k)(1 - 2P_e) E_0, & \Delta_k \in \{1, 2\}. \end{cases} \tag{B.12}
\]

Applying (B.9), (B.10) and (B.12) to (B.8), we have
\[
C_{24} = 2\alpha_h \left\{ (1 - 2P_e) (3\beta_h - 1 + \Re [(1 - 4\beta_h) \text{corr}(1) + \beta_h \cdot \text{corr}(2)]) \right\} - \Re [\text{corr}(1)] + 1 \right\}. \tag{B.13}
\]
Further, following (5.14) and using \( \text{MSE}_d \approx \text{MSE}_b \), we have
\[
P_e = (\text{MSE}_d - 1/\text{SNR}_r)/4 \approx (\text{MSE}_b - 1/\text{SNR}_r)/4 = (1/\text{SNR}_b - 1/\text{SNR}_r)/4. \tag{B.14}
\]
Substituting (B.14) into (B.13), we obtain (5.20).
Appendix C

Derivation of \( \mathbb{E}\{ f(e^{j\hat{\theta}_m(i)}) \} \) and MSE_{cpe}

C.1 Derivation of \( \mathbb{E}\{ f(e^{j\hat{\theta}_m(i)}) \} \)

For notational brevity, we simply replace the subscripts/superscripts \((i)_{m}, (i1)_{m1}\) and \((i2)_{m2}\) with the subscripts \(n, n-1\) and \(n-2\), respectively, in all CPE related variables. Applying (6.25) and \(\theta_n = \phi_n + \vartheta_n\) to (6.32), we obtain

\[
f(e^{j\hat{\theta}_n}) = (1 - \alpha_c)(e^{j\phi_n}e^{j\hat{\theta}_n} + w_n) + \alpha_1 f(e^{j\hat{\theta}_{n-1}}) + \alpha_2 f(e^{j\hat{\theta}_{n-2}}). \tag{C.1}
\]

Applying \(e^{j\theta_{n-1}} = \nu^*e^{j\theta_n}\) to (C.1), we further obtain

\[
\mathbb{E}\{f(e^{j\hat{\theta}_n})\} = (1 - \alpha_c)e^{j\vartheta_n} \sum_{l=0}^{n} a_l (\nu^*)^{n-l-n} \mathbb{E}\{e^{j\phi_l}\}, \quad n \geq 2 \tag{C.2}
\]

where \(a_n = 1, a_{n-1} = \alpha_1, a_{l-1} = \alpha_1 a_{l-1} + \alpha_2 a_l, l \in \mathbb{Z}_3^n\), and \(a_0 = (\alpha_c a_1 + \alpha_2 a_2)/(1 - \alpha_c)\).

In particular, for \(n \in \mathbb{Z}_1^n\), equation (6.32) reduces to \(f(e^{j\hat{\theta}_n}) = e^{j\theta_n}\) and \(f(e^{j\hat{\theta}_1}) = (1 - \alpha_c)e^{j\vartheta_1} + \alpha_c f(e^{j\hat{\theta}_n})\). Thus, we have

\[
\mathbb{E}\{f(e^{j\theta_0})\} = e^{j\theta_0} \mathbb{E}\{e^{j\phi_0}\}; \quad \mathbb{E}\{f(e^{j\theta_1})\} = (1 - \alpha_c)e^{j\vartheta_1} \mathbb{E}\{e^{j\phi_1}\} + \alpha_c e^{j\theta_0} \mathbb{E}\{e^{j\phi_0}\}. \tag{C.3}
\]

Referring to the discussion in [141, Section III-A], from (4.5), we have

\[
\mathbb{E}\{e^{j\phi_n(k)}\} = e^{-[n(N+N_g)+k]\beta T_s}, \quad k \in \mathbb{Z}_0^{N-1}. \tag{C.4}
\]
Using (4.13), we obtain
\[ E\{e^{j\phi_n}\} = \frac{1}{N} \sum_{k=0}^{N-1} E\{e^{j\phi_n(k)}\} = \frac{1}{N} \sum_{k=0}^{N-1} e^{-[n(N+N_g)+k]\pi\beta T_s} = \lambda_1 \lambda_0^n \]  \hspace{1cm} (C.5)
where \( \lambda_0 = e^{-(N+N_g)\pi\beta T_s} \) and \( \lambda_1 = \frac{1-e^{-\pi\beta T_s N}}{N(1-e^{-\pi\beta T_s})} \). Combining (C.2) and (C.3) by using (C.5) and \( E\{e^{j\theta_n}\} = e^{j\theta_n}E\{e^{j\phi_n}\} \) yields
\[ E\{f(e^{j\theta_n})\} = \begin{cases} E\{e^{j\theta_n}\}, & n = 0 \\ [(1 - \alpha_e) + \alpha_e \frac{\nu^*}{\nu_0}] E\{e^{j\theta_n}\}, & n = 1 \\ (1 - \alpha_e) \Omega E\{e^{j\theta_n}\}, & n \geq 2 \end{cases} \]  \hspace{1cm} (C.6)
where \( \Omega = \sum_{t=0}^{n} a_t (\frac{\nu^*}{\nu_0})^{n-t} \). By computing \( \Omega - \alpha_1 \Omega \frac{\nu^*}{\nu_0} - \alpha_2 \Omega (\frac{\nu^*}{\nu_0})^2 \), it is straightforward to obtain
\[ \Omega = \frac{1 + (\frac{\nu^*}{\nu_0})^n [a_0 - \alpha_1 a_1 - \alpha_2 a_2 - (\alpha_1 a_0 + \alpha_2 a_1) \frac{\nu^*}{\nu_0} - \alpha_2 a_0 (\frac{\nu^*}{\nu_0})^2]}{1 - \alpha_1 \frac{\nu^*}{\nu_0} - \alpha_2 (\frac{\nu^*}{\nu_0})^2}, \quad n \geq 2. \]  \hspace{1cm} (C.7)
Furthermore, when \( n \gg 1 \), we have
\[ \Omega \approx \frac{1}{1 - \alpha_1 \frac{\nu^*}{\nu_0} - \alpha_2 (\frac{\nu^*}{\nu_0})^2}. \]  \hspace{1cm} (C.8)
Applying (6.28) and (C.8) to (C.6), we obtain (6.33).

\section*{C.2 Derivation of MSE\textsubscript{cpe}}

The MSE\textsubscript{cpe} is given by
\[ \text{MSE}_{\text{cpe}} = E\{|f(e^{j\theta_n}) - e^{j\theta_n}|^2\} = 1 + E\{|f(e^{j\theta_n})|^2\} - E\{f(e^{j\theta_n})e^{-j\theta_n}\} - E\{e^{j\theta_n}f^*(e^{j\theta_n})\}. \]  \hspace{1cm} (C.9)
Let \( A_1 = E\{|f(e^{j\theta_n})|^2\} \) and note that \( E\{|f(e^{j\theta_n})|^2\} \approx E\{|f(e^{j\theta_{n-1}})|^2\} \) for \( n \gg 1 \).

From (C.1), we can obtain
\[ A_1 = (1-\alpha_e)^2(1+\sigma_w^2) + (\alpha_e^2 + \alpha_2^2) A_1 \]
\[ + \alpha_1 (1-\alpha_e) [E\{e^{j\theta_n}f^*(e^{j\theta_{n-1}})\} + E\{f(e^{j\theta_{n-1}})e^{-j\theta_n}\}] \]
\[ + \alpha_2 (1-\alpha_e) [E\{e^{j\theta_n}f^*(e^{j\theta_{n-2}})\} + E\{f(e^{j\theta_{n-2}})e^{-j\theta_n}\}] + \alpha_1 \alpha_2 A_2 \]  \hspace{1cm} (C.10)
where $A_2 = E\{ f(e^{j\hat{\theta}_{n-1}}) f^*(e^{j\hat{\theta}_{n-2}}) \} + E\{ f(e^{j\hat{\theta}_{n-2}}) f^*(e^{j\hat{\theta}_{n-1}}) \}$. Using the approximation $E\{ f(e^{j\hat{\theta}_{n-1}}) f^*(e^{j\hat{\theta}_{n-2}}) \} \approx E\{ f(e^{j\hat{\theta}_{n-2}}) f^*(e^{j\hat{\theta}_{n-3}}) \}$, $n \gg 1$, we obtain

$$A_2 = (1 - \alpha_c)\left[ E\{ e^{j\hat{\theta}_n} f^*(e^{j\hat{\theta}_{n-1}}) \} + E\{ f(e^{j\hat{\theta}_{n-1}}) e^{-j\hat{\theta}_n} \} \right] + 2\alpha_1 A_1 + \alpha_2 A_2. \tag{C.11}$$

Combining (C.10) and (C.11) yields

$$A_1 = \frac{1 - \alpha_c}{\alpha_0} \left[ \alpha_1 \left( E\{ e^{j\theta_n} f^*(e^{j\hat{\theta}_{n-1}}) \} + E\{ f(e^{j\hat{\theta}_{n-1}}) e^{-j\theta_n} \} \right) + \alpha_2 (1 - \alpha_2) \left( E\{ e^{j\hat{\theta}_n} f^*(e^{j\hat{\theta}_{n-2}}) \} + E\{ f(e^{j\hat{\theta}_{n-2}}) e^{-j\hat{\theta}_n} \} \right) + (1 - \alpha_2) (1 - \alpha_c) (1 + \sigma_w^2) \right]. \tag{C.12}$$

where $\alpha_0 = 1 - \alpha_2 - \alpha_1^2 - \alpha_2^2 - \alpha_2 \alpha_1 + \alpha_2^3$.

Moreover, from (C.1), we have

$$E\{ f(e^{j\hat{\theta}_n}) e^{-j\hat{\theta}_n} \} = (1 - \alpha_c) + \alpha_1 E\{ f(e^{j\hat{\theta}_{n-1}}) e^{-j\hat{\theta}_n} \} + \alpha_2 E\{ f(e^{j\hat{\theta}_{n-2}}) e^{-j\hat{\theta}_n} \}. \tag{C.13}$$

and

$$E\{ e^{j\theta_n} f^*(e^{j\hat{\theta}_n}) \} = (1 - \alpha_c) + \alpha_1 E\{ e^{j\hat{\theta}_n} f^*(e^{j\hat{\theta}_{n-1}}) \} + \alpha_2 E\{ e^{j\hat{\theta}_n} f^*(e^{j\hat{\theta}_{n-2}}) \}. \tag{C.14}$$

By applying (C.12), (C.13) and (C.14) to (C.9), we can obtain

$$\text{MSE}_{cpe} = \frac{1}{\alpha_0} \left[ A_0 + g_1 E\{ f(e^{j\hat{\theta}_{n-1}}) e^{-j\hat{\theta}_n} \} + g_2 E\{ f(e^{j\hat{\theta}_{n-2}}) e^{-j\hat{\theta}_n} \} + g_1 E\{ e^{j\hat{\theta}_n} f^*(e^{j\hat{\theta}_{n-1}}) \} + g_2 E\{ e^{j\theta_n} f^*(e^{j\hat{\theta}_{n-2}}) \} \right]. \tag{C.15}$$

where $A_0 = \alpha_0 (2\alpha_c - 1) + (1 - \alpha_2)(1 - \alpha_c) \sigma_w^2$, $g_1 = \alpha_1 (1 - \alpha_c - \alpha_0)$ and $g_2 = \alpha_2 [(1 - \alpha_2)(1 - \alpha_c) - \alpha_0]$. We note that

$$E\{ e^{j(\theta_n - l \hat{\theta}_n)} \} = E\{ e^{j(\phi_n - l \phi_n)} e^{j(\theta_n - l \hat{\theta}_n)} \} = (\nu^*)^l E\{ e^{j(\phi_n - l \phi_n)} \}, \quad n \geq 2, \quad l \in \mathbb{Z}_1^n. \tag{C.16}$$

Using (C.16) and (C.1) in (C.15), we can obtain

$$\text{MSE}_{cpe} = \frac{1}{\alpha_0} \left[ A_0 + (1 - \alpha_c) \sum_{l=1}^{n} b_{n-l} (\nu^*)^l E\{ e^{j(\phi_n - l \phi_n)} \} + \nu^l E\{ e^{j(\phi_n - l \phi_n)} \} \right], \tag{C.17}$$
for $n \geq 2$, where $b_n = g_2/\alpha_2$, $b_{n-1} = g_1$, $b_{n-2} = \alpha_1 b_{n-1} + \alpha_2 b_1$, $l \in \mathbb{Z}_3^n$, and $b_0 = (\alpha_c b_1 + \alpha_2 b_2)/(1 - \alpha_c)$.

Referring to [141, Section III-A], we have

$$E \{ e^{j(\phi_n - \phi_n)} \} = \frac{1}{N^2} E \left\{ \sum_{k=0}^{N-1} e^{j\phi_0(k)} \sum_{k=0}^{N-1} e^{-j\phi_n(k)} \right\}$$

$$= \frac{1}{N^2} \sum_{k_1=0}^{N-1} \sum_{k_2=0}^{N-1} E \left\{ e^{j\phi_0(k_1)} e^{-j\phi_n(k_2)} \right\}$$

$$= \frac{1}{N^2} \sum_{k_1=0}^{N-1} \sum_{k_2=0}^{N-1} e^{-\pi \beta T_s \nu |l(N+N_g)+k_2-k_1|}, \quad n \geq 2, \ l \in \mathbb{Z}_1^n. \quad (C.18)$$

Further algebra on (C.18) reduces it to

$$E \{ e^{j(\phi_n - \phi_n)} \} = \lambda_1 \lambda_2 \lambda_0^l, \quad n \geq 2, \ l \in \mathbb{Z}_1^n \quad (C.19)$$

where $\lambda_2 = \frac{1-e^{\pi \beta T_s \nu N}}{N(1-e^{\pi \beta T_s \nu})}$. Similarly, we can obtain

$$E \{ e^{j(\phi_n - \phi_n)} \} = \lambda_1 \lambda_2 \lambda_0^l, \quad n \geq 2, \ l \in \mathbb{Z}_1^n. \quad (C.20)$$

Combining (C.17), (C.19) and (C.20) yields

$$\text{MSE}_{cpe} = \frac{1}{\alpha_0} \left[ A_0 + (1 - \alpha_c) \lambda_1 \lambda_2 (\Omega_1 + \Omega_2) - \frac{2(1 - \alpha_c) g_2 \lambda_1 \lambda_2}{\alpha_2} \right], \quad n \geq 2. \quad (C.21)$$

where $\Omega_1 = \sum_{l=0}^n b_{n-l} (\lambda_0 \nu^*)^l$ and $\Omega_2 = \sum_{l=0}^n b_{n-l} (\lambda_0 \nu)^l$. Following a similar procedure for obtaining $\Omega$ in (C.7), for $n \gg 1$, we can derive $\Omega_1$ and $\Omega_2$ which are given by

$$\Omega_1 \approx \frac{b_n + (b_{n-1} - \alpha_1 b_n) \lambda_0 \nu^*}{1 - \alpha_1 \lambda_0 \nu^* - \alpha_2 (\lambda_0 \nu^*)^2}, \quad \Omega_2 \approx \frac{b_n + (b_{n-1} - \alpha_1 b_n) \lambda_0 \nu}{1 - \alpha_1 \lambda_0 \nu - \alpha_2 (\lambda_0 \nu)^2}. \quad (C.22)$$

We note that $\Omega_1 = \Omega_2^*$. Substitution of (C.22) into (C.21) yields (6.35).