A NOVEL APPROACH TO SPARSE SIGNAL RECONSTRUCTION

AND

NOISE REDUCTION

by

Mehdi Korki

Thesis submitted in partial fulfilment of the requirements for the degree of Doctor of Philosophy

FACULTY OF SCIENCE, ENGINEERING, AND TECHNOLOGY
SWINBURNE UNIVERSITY OF TECHNOLOGY
2016
ABSTRACT

Sparse signal recovery has a variety of applications in many fields. Many sparse signal recovery algorithms have been developed to reconstruct the sparse signal. Generally, these algorithms aim to find a solution to an underdetermined system of equations, where the solution is expected to be sparse. In some applications, the unknown signal to be estimated has an additional structure. If the structure of the signal is exploited, the better recovery performance can be achieved. A block-sparse signal, in which the nonzero samples manifest themselves as clusters, is an important structured sparsity. Block-sparsity has a wide range of applications in signal processing. However, existing algorithms for block-sparse signal recovery use the independent and identically distributed (i.i.d) model to describe the cluster structure of the nonzero elements of the unknown signal, which restricts their applicability and performance. Because many practically important signals, e.g. the impulsive noise in Power Line Communication (PLC), do not satisfy the i.i.d. condition, it is necessary to develop reconstruction algorithms for block-sparse signals using a more adequate signal model.

This thesis presents a novel Block Iterative Bayesian Algorithm (Block-IBA) for reconstructing block-sparse signals with unknown block structures. Block-IBA uses a more adequate Bernoulli-Gaussian hidden Markov model (BGHMM) to characterize the non-i.i.d. block-sparse signals commonly encountered in practice. Utilizing Bayesian hypothesis testing (BHT), this research presents another novel block Bayesian hypothesis testing algorithm (BBHTA) for structure-agnostic block-sparse signal recovery. Using Block-IBA, this work also presents a new impulsive noise reduction method for Orthogonal Frequency Division Multiplexing (OFDM) systems.

First, the Block-IBA for the recovery of structure-agnostic block-sparse signals is proposed. The Block-IBA iteratively estimates the amplitudes and positions of the
block-sparse signal using the steepest-ascent based Expectation-Maximization (EM), and effectively selects the nonzero elements of the block-sparse signal by a diminishing thresholding. Numerical experiments and simulations on synthetic and real-life data show the superior performance of Block-IBA to many state-of-the-art block-sparse signal reconstruction algorithms.

Second, a novel BBHTA is presented for reconstructing the structure-agnostic block-sparse signals. BBHTA uses a joint detection-and-estimation structure comprising a block-sparse support detector and a nonzero amplitude estimator. The support samples are detected using Bayesian hypothesis testing (BHT), and the nonzero amplitude samples are estimated by linear minimum mean-square error estimation (LMMSEE). Numerical experiments demonstrate the effectiveness of BBHTA.

Finally, using the Block-IBA, a new receiver for bursty impulsive noise estimation and rejection is proposed for OFDM systems. This novel receiver utilizes the guard band null subcarriers and data subcarriers for the impulsive noise estimation and cancellation. Unlike some other general OFDM transceivers which use time-domain interleaving (TDI) to cancel impulsive noise, we present a new receiver that suppresses the bursty channel noise based on Block-IBA estimation of the noise, which removes the delay due to TDI and saves memory space. Numerical experiments show that the proposed receiver outperforms existing receivers under the block-sparse impulsive noise environment.
To my lovely wife, Maliheh
Acknowledgments

I am deeply indebted to my Principal Coordinating Supervisor, Prof. Jingxin Zhang, for his constant supports. I thank him for supervising my research work incessantly and providing me with invaluable advice and technical support. He always kindly guided my research with patience and encouragement.

I owe my deepest gratitude to my Associate Supervisor, Prof. Cishen Zhang, whose support and guidance enabled me to accomplish the work. His commitment and belief in my abilities have enabled me to pursue my interests in my research field.

I would also like to thank Dr. Hadi Zayyani, for his valuable technical advice and numerous creative suggestions.

Finally, and always, I thank my lovely wife, Maliheh Tahmasbi. Without her love, dedication and devotion, none of these is ever possible.
Declaration

I hereby certify that this thesis entitled “A Novel Approach to Sparse Signal Reconstruction and Noise Reduction” is my own work, except where due reference is made in the text and that, to my best knowledge and belief, it has not been submitted to this university or to any other university or institution for a degree.

Signed

Mehdi Korki

13 May 2016
# Table of Contents

Abstract ........................................... ii

Dedication ......................................... iv

Acknowledgments .................................... v

Declaration ......................................... vi

List of Tables ..................................... xii

List of Figures ..................................... xiii

List of Abbreviations ............................... xvi

Notation ............................................ xix

1 Introduction ...................................... 1

1.1 Motivation .................................. 1

1.2 The Problem Statement and Objectives .... 2

1.3 Significance of the Work .................... 3

1.4 Contributions ................................ 5

1.5 Organization of the Thesis ................. 7

1.6 Publications .................................. 8

2 Background ...................................... 10

2.1 Introduction ................................ 10

2.2 Sparse Signal Recovery: Models and Algorithms ........ 11
2.3 Block Sparse Signal Recovery: Models and Algorithms .......................... 16
2.4 Sparse Signal Recovery and Its Applications ................................. 20
  2.4.1 Block Sparse Impulsive Noise Reconstruction in Communication Systems ........................................................................ 21
  2.4.2 Block Sparse Image Reconstruction in MRI .............................. 22
2.5 Impulsive Noise Models in Wireless and Power line Communication (PLC) Systems .......................................................... 23
2.6 OFDM Modulation in Digital Communication Systems ................... 27
2.7 Impulsive Noise Receivers in OFDM Systems ................................. 33
  2.7.1 Time-Domain Preprocessors Receivers ................................ 34
  2.7.2 Iterative Receivers .......................................................... 35
  2.7.3 Factor Graph Receivers .................................................. 36
  2.7.4 Sparse Impulsive Noise Cancellation Receivers .................... 37
2.8 Open Problems Identified in the Literature ................................. 39
2.9 Conclusion .............................................................................. 41

3 Iterative Bayesian Reconstruction of Non-IID Block-Sparse Signals 42
  3.1 Introduction .......................................................................... 42
  3.2 Signal Model ......................................................................... 44
  3.3 Optimum Estimation of $w$ ..................................................... 46
    3.3.1 MAP Estimation of $s$ ..................................................... 47
    3.3.2 MAP Estimation of $\theta$ using Gamma Prior ........................ 48
  3.4 Block Iterative Bayesian Algorithm ........................................ 52
    3.4.1 Main Idea ...................................................................... 52
    3.4.2 Discussion ...................................................................... 56
  3.5 Learning The Signal Model Parameters ..................................... 57
3.6 Analysis of Global Maximum and Local Maxima

3.6.1 Analysis of Global Maxima

3.6.2 Analysis of Local Maxima

3.6.3 Analysis of Global Maximum of Overall Block-IBA

3.7 Numerical Evaluation

3.7.1 Performance of Block-IBA versus Block size

3.7.2 Performance of Block-IBA versus algorithm parameters

3.7.3 Effect of Sparsity Level on the Performance

3.7.4 Effect of Signal to Noise Ratio (SNR) on the Performance

3.7.5 Real-World Data Experiment

3.8 Conclusion

4 Bayesian Hypothesis Testing for Block Sparse Signal Recovery

4.1 Introduction

4.2 Signal Model

4.3 The Proposed Algorithm

4.3.1 Support Detection Using Bayesian Hypothesis Testing

4.3.2 Amplitude Estimation Using LMMSE

4.4 Simulation Results

4.5 Conclusion

5 Block-Sparse Impulsive Noise Reduction in OFDM Systems - A Novel Iterative Bayesian Approach

5.1 Introduction

5.2 System Model

5.2.1 OFDM Transmission Model

5.2.2 Impulsive Noise Model
5.3 Formulation of Impulsive Noise Receiver ........................................... 100
5.3.1 Problem Statement ................................................................. 100
5.3.2 Estimation of $e$ ................................................................. 103
5.4 Block Iterative Bayesian Algorithm for Impulsive Noise Estimation .... 107
5.4.1 Block Iterative Bayesian Algorithm (Block-IBA) Using Null Tones 107
5.4.2 Extension of Block-IBA using All Tones ................................. 110
5.4.3 Learning The Impulsive Noise Model Parameters .................. 113
5.4.4 Null Tone Placement ............................................................ 116
5.5 Simulation Results ................................................................. 120
5.5.1 Performance of Uncoded OFDM System ................................. 123
5.5.2 Performance of Uncoded OFDM System with Random Placement of Null Tones ......................................................... 125
5.5.3 Performance of Uncoded OFDM System Under Different Parameters of Noise ................................................................. 127
5.5.4 Performance of Coded OFDM System .................................... 128
5.5.5 Complexity ................................................................. 130
5.6 Conclusion ................................................................. 130

6 Conclusion ........................................................................... 133
6.1 Summary of Results ............................................................. 133
6.2 Future Work ................................................................. 135

Bibliography ........................................................................... 137

Appendices ........................................................................... 164
A Derivation of Steepest Ascent Formulation ................................. 164
B MAP Update Equation for the Signal Model Parameter $p_{01}$ 164
<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>C</td>
<td>Proof of Lemma 1 165</td>
</tr>
<tr>
<td>D</td>
<td>Proof of Lemma 2 167</td>
</tr>
<tr>
<td>E</td>
<td>MAP Update Equation for the Signal Model Parameter ( \sigma_\theta ) in (4.26) 171</td>
</tr>
<tr>
<td>F</td>
<td>MAP Update Equation for the Signal Model Parameter ( p_{10} ) 171</td>
</tr>
<tr>
<td>G</td>
<td>Derivation of Steepest Ascent Formulation in (5.36) 173</td>
</tr>
</tbody>
</table>
## List of Tables

2.1 Statistical models for i.i.d. impulsive noise and their applications in communication systems. ......................................................... 26

3.1 Performance of Block-IBA on MRI DataSet Compared to other algorithms .................................................................................. 73

5.1 Iterative estimation of actual parameters in the case of $M = 80$, $N = 256$, $p = 0.9$, $p_{10} = 0.01$, $p_{01} = 0.09$, $\sigma_\theta = 0.3147$ $\sigma_n = 0.03$. ........................................ 123

5.2 Comparison of the mean runtime of the algorithms for 1000 OFDM symbols under BGHMM noise .................................................. 130
List of Figures

2.1 The simplified OFDM system model in which the FFT spreads the impulsive noise (in red) over all subcarriers. 29

2.2 The performance of standard OFDM system versus single carrier (SC) system in the impulsive noise environment. 31

3.1 Performance of all algorithms vs $p_{01}$ for $N = 192$, $M = 512$. (a) NMSE versus $p_{01}$ in noisy scenario, SNR = 15dB. (b) Success rate versus $p_{01}$ in noiseless scenario. Results are averaged over 400 simulations. 67

3.2 Support vector s samples for $p_{01} = 0.09$, $p_{01} = 0.45$, and $p_{01} = 0.9$. 68

3.3 Performance of the Block-IBA versus parameter $\alpha$ for $N = 192$, $M = 512$ and $\sigma_\theta = 1$. In (a), the number of active sources, $k$, is fixed to 50 and the effect of SNR is investigated. In (b), SNR is fixed to 15 dB and the effect of sparsity factor is assessed. Values of $k$ are 30, 60, 80. Results are averaged over 400 simulations. 69

3.4 Effect of the initial values of the threshold on the Block-IBA. The simulation parameters are $M = 512$, $N = 192$, $p = 0.9$, $p_{01} = 0.09$, $\sigma_\theta = 1$ and $\mu = 10^{-6}$. Results are averaged over 400 simulations. 70

3.5 NMSE versus normalized sparsity ratio for different algorithms. Simulation parameters are $M = 256$, $N = 96$, $\sigma_\theta = 1$, SNR = 15dB and $p_{01} = 0.45$. Results are averaged over 400 runs. 71

3.6 NMSE vs. SNR (dB) for different algorithms. Simulation parameters are $M = 512$, $N = 192$, $\sigma_\theta = 1$, and $p_{01} = 0.45$. Results are averaged over 400 runs. 72
3.7 MRI image reconstruction performance for different algorithms accompanied by the corresponding error images. ................................. 75

4.1 NMSE (dB) versus SNR for BBHTA and BPA with $p = 0.9$, $p = 0.8$, and $p = 0.7$. The results are averaged over 400 trials. ................................. 88

4.2 NMSE versus $p_{01}$ for BBHTA and other algorithms. The results are averaged over 400 trials. ................................. 89

5.1 The conventional coded OFDM system. ................................. 95

5.2 The first-order (two-state) Markov chain model. ................................. 97

5.3 Trellis diagram of block-sparse impulsive noise. ................................. 98

5.4 Three realization of the noise model generated by BGHMM for different values of noise memory $\eta$. The impulsive to background noise power ratio (INR) is 20 dB. 99

5.5 The proposed coded OFDM system with block-sparse impulsive noise (IN) mitigation in the receiver. ................................. 103

5.6 SER vs. the number of null tones for 4-QAM uncoded OFDM system with total 256 subcarriers in block-sparse impulsive noise environment. ................................. 114

5.7 Block-sparse impulsive noise mitigation receiver. ................................. 116

5.8 SER vs. SNR for 4-QAM uncoded OFDM system with total 256 subcarriers, and 80 null subcarriers under the block-sparse impulsive noise. (a) Impulsive noise with two-state Markov chain. (b) Impulsive noise with three-state Markov chain. ................................. 126

5.9 SER vs. SNR for 4-QAM uncoded OFDM system with total 256 subcarriers, and 80 null subcarriers under the i.i.d. impulsive noise. ................................. 127

5.10 Caption for LOF ................................. 128
5.11 SER vs. SNR for 4-QAM uncoded OFDM system with total 256 subcarriers, and 80 null subcarriers in impulsive noise environment. (a) BGHMM impulsive noise with $p = 0.8$ and $p_{10} = 0.01$. (b) BGHMM impulsive noise with $p = 0.9$ and $p_{10} = 0.03$.

5.12 BER vs. SNR for 4-QAM coded OFDM system with total 256 subcarriers, and 80 null subcarriers in impulsive noise environment. (a) Impulsive noise with two-state Markov chain. (b) Impulsive noise with three-state Markov chain.
List of Abbreviations

ADC    Analogue to Digital Converter
AWGN   Additive White Gaussian Noise
A/D    Analogue to Digital
BGHMM  Bernoulli-Gaussian Hidden Markov Model
BHT    Bayesian Hypothesis Testing
BBHTA  Block Bayesian Hypothesis Testing Algorithm
BPA    Bayesian Pursuit Algorithm
BP     Belief Propagation
BM     Boltzmann Machine
BSBL   Block Sparse Bayesian Learning
BICM   Bit Interleaved Coded Modulation
BER    Bit Error Rate
CS     Compressed Sensing
CoSaMP Compressed Sampling Matching Pursuit
CluSS  Clustered Sparse Solver
DWT    Discrete Wavelet Transform
DoA    Direction of Arrival
DFT    Discrete Fourier Transform
D/A    Digital to Analogue
EM     Expectation Maximization
EEG    ElectroEncephaloGram
FOCUSS FOCal Underdetermined System Solver
FFT    Fast Fourier Transform
FSK    Frequency Shift Keying
FEC    Forward Error Correction
GM     Gaussian Mixture
GHMM   Gaussian Hidden Markov Model
GAMP   Generalized Approximate Message Passing
IBA    Iterative Bayesian Algorithm
IRLS   Iteratively Reweighted Least Squares
ISI    Inter-Symbol Interference
ICI    Inter-Carrier Interference
LMMSEE Linear Minimum Mean Square Error Estimation
LDPC   Low Density Parity Check
MAP    Maximum a Posteriori Probability
MRI    Magnetic Resonance Imaging
MMV    Multiple Measurement Vector
MCMC   Markov Chain Monte Carlo
MEG    MagnetoEncephaloGram
MMSE   Minimum Mean Squared Error
NP     Non-deterministic Polynomial-time
OFDM   Orthogonal Frequency Division Multiplexing
OMP    Orthogonal Matching Pursuit
PLC    Power Line Communication
PSD    Power Spectral Density
PC     Pattern Coupled
PPP    Poisson Point Processes
PRIME  PoweRline Intelligent Metering Evolution
P/S  Parallel to Serial
QoS  Quality of Service
QPSK Quadrature Phase Shift Keying
SMV  Single Measurement Vector
SCA  Sparse Component Analysis
SBL  Sparse Bayesian Learning
SNR  Signal to Noise Ratio
S/P  Serial to Parallel
SC  Single Carrier
SINR Signal to Impulsive Noise Ratio
TDI  Time Domain Interleaving
TFDI Time-Frequency Domain Interleaving
URP  Unique Representation Property
Notation

Lower-case letters, e.g. $x$, bold-faced lower-case letters, e.g. $\mathbf{x}$, and bold-faced upper-case letters, e.g. $\mathbf{X}$, denote scalars, vectors, and matrices, respectively.

$(\cdot)^T$ Transpose
$(\cdot)^H$ Hermitian transpose
$(\cdot)^{-1}$ Matrix inversion
$(\hat{\cdot})$ Estimate
$\text{det} (\cdot)$ Matrix determinant
$\odot$ Hadamard (element-by-element) product
$\|\cdot\|_0$ $\ell_0$-norm of a vector
$\|\cdot\|_1$ $\ell_1$-norm of a vector
$\|\cdot\|_2$ $\ell_2$-norm of a vector
$I_M$ $M \times M$ identity matrix
$p_X (x)$ Probability density function (PDF) of a random variable $X$
$\Pr \{\cdot\}$ Probability of an event
$\mathcal{N} (\mathbf{b}, \mathbf{C})$ Gaussian distribution with mean $\mathbf{b}$ and covariance matrix $\mathbf{C}$
$\text{diag} (\mathbf{x})$ A matrix with the elements of vector $\mathbf{x}$ on the main diagonal
$\text{Tr} (\cdot)$ Trace of a matrix
$\mathbb{E} (\cdot)$ Expectation
$\mathcal{CN} (\mathbf{b}, \mathbf{C})$ Complex Gaussian distribution with mean $\mathbf{b}$ and covariance matrix $\mathbf{C}$
$\mathbb{R}^N$ Space of real vectors of dimension $N$
$\mathbb{R}^{N \times N}$ Space of real matrices of dimension $N \times N$
$\mathbb{C}^N$ Space of complex vectors of dimension $N$
$\mathbb{C}^{N \times N}$ Space of complex matrices of dimension $N \times N$
$S$ Set of elements
Sub-matrix or sub-vector corresponding to the rows or elements indexed by the set $T$
Chapter 1
Introduction

1.1 Motivation

Recognizing the ubiquitous applications of the compressed sensing and sparse signal recovery, we aim through this research effort to develop robust and computationally efficient block-sparse signal recovery algorithms and evaluate their performances in some applications such as impulsive noise reduction in communication systems. The application of the proposed algorithms in the receiver of orthogonal frequency division multiplexing (OFDM) systems may provide a much improved alternative impulsive noise receiver to the existing impulsive noise receivers in terms of performance gain and delay removal due to time-domain interleaving (TDI) in the existing receivers.

Although many algorithms have been developed for block-sparse signal recovery, most of them use independent and identically distributed (i.i.d.) signal model to represent the block structure of the nonzero elements of the unknown signal, which may restrict their applicability and performance. It is therefore necessary to develop a reconstruction algorithm using a more practical signal model. Also, future communications systems (e.g., wireless networks) must be able to combat interference and thus offer quality of service (QoS) guarantees. Since the impulsive noise is one of the major impairments to wireless and wireline communication systems, it is important to redesign the conventional receiver to cope with the impulsive noise.

The above arguments provide the main motivation for this research which addresses the development of more practical and robust block-sparse signal recovery algorithms and their applications to impulsive noise mitigation in communication
systems.

1.2 The Problem Statement and Objectives

To reconstruct the structure-agnostic block sparse signal, some algorithms have been reported in the literature. But all these algorithms assume the i.i.d. block structure of the signal which is impractical in many applications, e.g. the bursty impulsive noise in Power Line Communication (PLC). Also, most existing algorithms for block-sparse signal recovery assume the prior knowledge of the signal model parameters, which is often unavailable in practical settings particularly when working with real-life datasets. However, automatically learning these unknown parameters is important and might significantly improve the performance of the block-sparse signal recovery algorithm. Moreover, these algorithms lack an effective method for selecting the most probable support set based on the underlying structure of the signal.

The impulsive noise is the principal cause of errors in digital communication networks, e.g. PLC networks. For instance, practical experiments in PLC show that the Power Spectral Density (PSD) of the impulsive noise exceeds the PSD of the background noise by a minimum of 10-15 dB and may sometimes reach more than 50 dB. Moreover, the impulsive noise in PLC is block-sparse or bursty which significantly degrade the performance of the system since the samples of this bursty impulsive noise are no longer statistically independent. Hence, the design of a modem in wireless and wireline communication systems is a challenging problem, particularly in coping with the various sources of impulsive noise. To deal with harsh channel conditions of wireless and PLC networks and to achieve higher data rates, it is inevitable to employ the impulsive noise mitigation techniques.

The main objectives of this thesis are:
• to develop efficient block sparse signal recovery algorithms which use a more adequate model for capturing the block sparsity of the impulsive noise in practical applications such as PLC and automatically learn the parameters of the statistical signal model.

• to propose a specific OFDM receiver that estimates and rejects the block-sparse impulsive noise to improve the system performance and removes the drawbacks of general OFDM receivers.

In the following chapters, we will address these objectives through developing efficient block-sparse signal recovery algorithms and designing a specific receiver for bursty impulsive noise mitigation in OFDM systems. Before this, in the next chapter, we proceed with providing some background information concerning the model and algorithms of sparse- and block-sparse signal recovery and their applications, different impulsive noise models in wireless and PLC networks, and some prior work on OFDM receiver design in impulsive noise environment.

1.3 Significance of the Work

Conventional methods for sampling signals or images utilize Shannon’s theorem. Shannon’s theorem implies that the sampling rate must be at least twice the maximum frequency present in the signal, i.e. Nyquist rate. The Nyquist rate sampling is the underlying principle in almost all signal acquisition protocols utilized in consumer audio and visual electronics, medical imaging devices, and radio receivers to name a few. For instance, the standard analogue-to-digital converter (ADC) uniformly samples the signal at or above the Nyquist rate. However, compressed sensing (CS), which is a modern sampling/sensing model, aims to accurately recover sparse signal with only a few nonzero elements from far fewer samples than the conventional method
uses. CS is a simple and efficient sampling method which samples at low rate and later uses computational power to recover the original sparse signal from an incomplete set of measurements. Moreover, many signals are sparse or compressible in an appropriate basis. For instance, wavelet transform of images has many small coefficients with relatively a few large coefficients that capture most of information. Also, the time-domain impulsive noise in many wireless and PLC systems is sparse. The common CS algorithms to recover sparse solutions are $\ell_1$-minimization and greedy algorithms. CS and sparse signal recovery have various applications such as data compression, designing fast channel coding, and computationally-efficient data acquisition to name a few. In particular, using CS in data acquisition to design low rate sampling devices could have an enormous impact on designing the simple hardware in data acquisition. In practice, the sparse signal has additional structure, e.g. block-sparsity. Block sparsity, in which the nonzero elements manifest themselves as clusters, has important applications in block-sparse impulsive noise estimation in PLC and clustered-sparse channel estimation. To recover the sparse and block-sparse signals, many algorithms have been proposed. However, these algorithms have limitations that restrict their applicability in practical settings. For instance, sparse or block-sparse signal models in these algorithms do not reflect the practical signal models in many applications, e.g. bursty impulsive noise in PLC. Hence, it is essential to develop algorithms that utilize more realistic and practical signal model and efficiently recover the sparse signal. Since the impulsive noise is the major impairment to digital communication networks, it is necessary to develop algorithms and methods to mitigate the effect of impulsive noise and thus improve the performance and increase the data rate.
1.4 Contributions

The main contributions of this thesis are summarized as follows:

1. **Developing An Iterative Bayesian Algorithm for Reconstructing Non-IID Block-Sparse Signals:** In this contribution, we develop a novel iterative Bayesian algorithm (Block-IBA) which

   - uses a Bernoulli-Gaussian hidden Markov model (BGHMM) [1] for the block-sparse signals. This model better captures the burstiness (block structure) of the impulsive noise and hence is more adequate in practical applications such as PLC.

   - incorporates, different to the other algorithms [2]–[6], a diminishing threshold for determining the active columns of the measurement matrix $\Phi$, which effectively selects the nonzero elements of the signal $w$. Using this technique, Block-IBA improves the reconstruction performance for the block-sparse signals.

   - uses a maximum \textit{a posteriori} (MAP) estimation procedure to automatically learn the parameters of the statistical signal model (e.g. the elements of state-transition matrix of BGHMM), averting complicated tuning updates.

2. **Developing a Bayesian Hypothesis Testing Algorithm for Block Sparse Signal Recovery:** In this contribution, a low-computational Bayesian algorithm is developed based on the Bayesian hypothesis testing (BHT). We refer to the proposed algorithm as block Bayesian hypothesis testing algorithm (BB-HTA), which possesses the following features.
• BBHTA jointly detects the supports by BHT and estimates the amplitudes by linear minimum mean-square error estimation (LMMSEE). BHT renders BBHTA a robust support detection against additive measurement noise.

• Using BHT and different to the existing greedy algorithms such as Byesian pursuit algorithm (BPA) [7], BBHTA searches for the start and termination of the active support blocks. This search yields two simple thresholds to detect and recover the supports of the block-sparse signals, at a low computation price.

• The proposed BBHTA is a double-looped and turbo-like approach, the inner loop being the serial procedure for detecting the supports and the outer loop being a “turbo-like” approach for estimating the amplitudes using LMMSEE. This novel implementation offers a more accurate recovery of block-sparse signals.

3. Designing A Robust and Efficient OFDM Receiver in Impulsive Noise Channels: This contribution builds on the developed algorithm of the first contribution, which utilizes the Block-IBA to design a novel OFDM receiver, in the impulsive noise channels, with the following features.

• Different to the other general OFDM receivers [8], [9], it exploits the block-sparse structure of the impulsive noise for estimation and removal, and hence removes the delay due to the Time Domain Interleaving (TDI) in general OFDM receivers.

• It uses a MAP procedure to estimate the variance and the state transition matrix of Markov chain model for the noise, which averts complicated
tuning updates.

- Although the developed OFDM receiver is specifically designed for block-sparse impulsive noise estimation and removal, it still shows a good performance in sparse impulsive noise channels. This in turn demonstrates its robustness against the impulsive noise distribution.

1.5 **Organization of the Thesis**

The thesis is organized as follows:

Chapter 2 presents an overview of the basic concepts that are used in this work and also prior work in the literature. It begins with the models and algorithms of sparse- and block-sparse recoveries. Then, it introduces some applications of block-sparse signal reconstruction in communication systems and MRI image recovery. The different impulsive noise models in wireless and PLC networks are introduced and the basic OFDM modulation in digital communication systems is described. The chapter ends with some discussions of prior work on OFDM receiver design in impulsive noise environment and the specific research problems identified in the literature.

Chapter 3 presents the proposed Block-IBA for reconstructing block-sparse signals with unknown block structures. It starts with the statistical modeling of the block-sparse signal using the Bernoulli-Gaussian Hidden Markov model (BGHMM). Then, it proposes the optimum estimation of the unknown block sparse signal using MAP solution. Based on the MAP solution, a novel Block-IBA is developed. The analyses of the global and local maxima properties of the Block-IBA are also presented. Finally, the experimental results on synthetic and real-life data are presented.

Chapter 4 presents the novel BBHTA for the recovery of structure-agnostic
block-sparse signals. BBHTA uses the same BGHMM for the block-sparse signal model. This chapter starts with presenting the proposed BBHTA including support detection using BHT and the amplitude estimation using LMMSEE. The chapter concludes with the presentation and discussion of the numerical results on the performance of BBHTA.

Chapter 5 presents the novel block-sparse impulsive noise receiver for OFDM systems in wireless and PLC networks. It begins with the description of the OFDM and the busy impulsive noise models. Then, it investigates the formulation and estimation of impulsive noise. The novel receiver for bursty impulsive noise estimation and cancellation, based on the Block-IBA, is then presented. The chapter ends with the presentation and discussion of the numerical results of the Block-IBA receiver in the bursty impulsive noise environment.

Finally, Chapter 6 summarizes the contributions of this thesis, highlights the important results obtained, and outlines the avenues for future research.

1.6 Publications

Publications emanating from this work are listed as follows.


Transactions on Communications, vol. 64, no. 1, pp. 271-284, Jan. 2016. (Chapter 5).

Chapter 2
Background

2.1 Introduction

Sparse signal recovery frequently occurs in the fields of signal processing, compressed sensing, information theory, pattern recognition, statistics, neuroscience, bioinformatics and machine learning. Moreover, the block-sparse signal recovery, in which the nonzero samples manifest themselves as clusters, has a wide range of applications in signal processing, system biology, image reconstruction, and block-sparse impulsive noise estimation in wireless and Power Line Communication (PLC) networks.

This chapter presents the relevant background for the models and applications of sparse and block-sparse signal recoveries, impulsive noise models in wireless and PLC systems, and impulsive noise receivers in orthogonal frequency division multiplexing (OFDM) systems. Sections 2.2 and 2.3 briefly discuss the models and algorithms of sparse and block-sparse signal recoveries, respectively, and highlight some of their shortcomings that are addressed by this thesis. Section 2.4 highlights some important applications of sparse signal recovery and gives some practical scenarios where blocks-sparse signal recovery plays an important role. Section 2.5 discusses the statistical impulsive noise models that is used in the literature to represent the aperiodic and periodic impulsive noise in wireless and PLC systems. Section 2.6 presents the basic model for OFDM modulation and briefly discusses the effect of impulsive noise on its performance. Section 2.7 addresses prior work on designing OFDM receivers in impulsive noise, highlight some of their drawbacks, and how they relate to the receiver proposed in this thesis. Section 2.8 addresses some of the research prob-
lems and shortcomings that have been identified in the literature. Finally, Section 2.9 presents some concluding remarks.

2.2 Sparse Signal Recovery: Models and Algorithms

Consider the general linear model

\[ y = \Phi w + n, \quad (2.1) \]

where \( \Phi \in \mathbb{R}^{N \times M} \) is a known matrix, \( y \in \mathbb{R}^N \) is the available measurement vector, and \( n \in \mathbb{R}^N \) is an unknown Gaussian corrupting noise. We aim to estimate the original unknown signal \( w \in \mathbb{R}^M \) when \( N \ll M \). The linear model presented in (2.1) is the most fundamental sparse signal recovery [10]-[14] model which is also called single measurement vector (SMV) model. In general, the known matrix \( \Phi \) is assumed to have the unique representation property (URP) [10], i.e. any \( N \) columns of \( \Phi \) are linearly independent. In different applications, the matrix \( \Phi \) and the vectors \( y \) and \( w \) have different names. For instance, in compressed sensing (CS) [14], \( \Phi \) is called the sensing matrix or the measurement matrix, \( y \) is called the measurement vector, where \( w \) is the signal to be recovered. In the context of sparse component analysis (SCA) [15], \( \Phi \) is the mixing matrix, \( y \) is the mixing vector and \( w \) is the source vector. In signal representation [16], \( \Phi \) is called the basis matrix or the dictionary matrix, \( y \) is called signal vector and \( w \) is called sparse coefficient vector.

Under the condition \( N \ll M \) the underdetermined systems of linear equations in (2.1) has an infinite number of solutions, which makes the problem challenging and requires appropriate prior knowledge about the unknown signal \( w \). An appropriate prior knowledge that can lead to recovery of \( w \) is the sparsity, namely, the majority of the elements of the unknown vector \( w \) are zero (or near zero), while only a few
components are nonzero. Hence, it is feasible to find the true solution of $w$ with small errors [11], or even exactly in some cases [12], provided that it is sufficiently sparse.

Knowing the sparsity of vector $w$ a priori, a theoretical approach to recover the signal $w$ is to search for the solution with the minimum $\ell_0$-norm. In fact, the sparsest solution of (2.1) can be found by minimizing $\|w\|_0$. However, this approach is computationally intractable (i.e., NP-hard) [17]. This is because it requires a combinatorial search over the entire solution space, which is computationally a daunting task when $M$ is large. Therefore, different methods and algorithms have been proposed to find the sparsest solution in a tractable manner. These methods can be divided into different categories which are listed as follows.

- **Convex Optimization Algorithms**: These popular methods suggest a convexification procedure to solve the problem. Under some appropriate conditions for $\Phi$ and $w$, it is possible to find the exact solution of (2.1) by solving the following $\ell_1$ minimization problem $[11], [18], [19]:$

$$
P_1 : \text{minimize } \|w\|_1 \quad \text{subject to } \|y - \Phi w\|_2^2 \leq \lambda, \quad (2.2)$$

where $\lambda$ is a regularizer. Another equivalent form of (2.2) which is a theoretically proven and practically effective approach that leads to the optimal solution of the signal $w$ is

$$
\hat{w} = \arg\min_w \beta \|y - \Phi w\|_2^2 + \tau \|w\|_1, \quad (2.3)
$$

where $\tau$ is the regularization parameter controlling the degree of the sparsity of the solution. Moreover, the effect of Gaussian noise $n$ in (2.1) with zero mean and variance $\beta^{-1/2}$ is implicitly embedded in (2.3). The most successful
approaches to solve (2.2) or (2.3) are Basis Pursuit Denoising [13] and Lasso [20]. However, these algorithms require tuning the regularizer $\lambda$ or $\tau$. Some methods including model selection [21], [22], cross-validation and the $L$-curve method [23], [24] have been developed to optimally tune the parameters $\lambda$ and $\tau$. However, there are some applications where tuning is very difficult or even infeasible. Hence, optimal selection of $\lambda$ or $\tau$ remains challenging and an ongoing research problem [25], [26]. Moreover, these algorithms generally perform a biased estimation of the solution. That is, the global minimum of (2.2) or (2.3) and the sparsest solution does not necessarily coincide unless $\Phi$ and $w$ satisfy strict conditions. Unlike the Bayesian approaches, these algorithms only utilize the sparsity of the unknown signal $w$ without considering any a priori statistical information or specific structure on the signal to be estimated. However, considering the a priori statistical information or further exploiting the structure of the signal can lead to better recovery performance. More importantly, the Lasso-type algorithms compared to Bayesian approaches such as sparse Bayesian learning (SBL) [27], are not robust against scale variation of the solution. In other words, if $w_{\text{SBL}}$ is the optimal solution provided by SBL with sensing matrix $\Phi$, when we rescale the sensing matrix $\Phi$ with a diagonal matrix $D$, i.e. $\Phi \rightarrow D\Phi$ the optimal solution of SBL algorithm is also rescaled, i.e. $w_{\text{SBL}} \rightarrow Dw_{\text{SBL}}$. In contrast, the Lasso-type algorithms provide the solutions without such a linear relationship between them. This makes the Lasso-type algorithms inefficient in source localization and other regression applications when the rescaled version of $\Phi$ is used.

- Non-convex Optimization Algorithms: The sparse reconstruction algorithms in this category use minimum $\ell_p$-norm to replace minimum $\ell_1$-norm where $0 < p < 1$.
\( p < 1 \). Hence, the effective approach to recover the signal \( w \) in (2.1) is to solve the following optimization problem

\[
\hat{w} = \arg\min_w \beta \|y - \Phi w\|_2^2 + \tau \|w\|_p \quad (0 < p < 1) ,
\]

where \( \|w\|_p = \left( \sum_i |w_i|^p \right)^{1/p} \). Solving the minimization problem in (2.4) generally results in an iterative reweighted algorithm [10], [28], [29], [30]. The most popular algorithms among these reweighted algorithms are iteratively reweighted least squares (IRLS) [28] and the FOCal Underdetermined System Solver (FOCUS) [10], [30] algorithms. In particular, the FOCUSS algorithm has been widely applied to neuromagnetic source localization [31]. In the non-convex optimization algorithms, tuning the optimal value of \( \tau \) is still a difficult task.

- Smoothed \( \ell_0 \)-norm Algorithms: In these algorithms, the \( \ell_0 \)-norm of \( w \) is approximated with a smooth and continuous function [32], [33]. In other words, the smooth measure of sparsity is provided by the smoothed \( \ell_0 \) algorithm. The \( \ell_0 \)-norm of \( w \) is defined as \( \|w\|_0 = \sum_i I (w_i \neq 0) \), where \( I (\cdot) \) is the indicator function. The algorithms in this category are fast and their recovery performance in noiseless scenarios is excellent. However, they lose robustness to noise in noisy scenarios. Different methods [34] have used different continuous functions to approximate the \( \ell_0 \)-norm in noisy scenarios. However, their performance improvement is not as significant as it is in the noiseless scenarios.

- Greedy Algorithms: The algorithms in this category find the support of the unknown signal \( w \) sequentially and iteratively [35]–[41]. They use the correlation values between the columns of matrix \( \Phi \) and the measurement vector \( y \) to select
the true supports of the sparse vector \( \mathbf{w} \). Although these algorithms have low computational complexity and they are fast, the correlations between all pairs of the columns of matrix \( \Phi \) has a significant effect on their recovery performance. In addition, they are unable to operate in noisy regime for good recovery performance. Hence, their applications are limited by the abovementioned drawbacks to some specific problems such as source localization and tracking.

- **Message Passing Algorithms:** Another popular algorithms are message passing algorithms [42]–[50]. These algorithms require relatively few iterations to converge, thus they are computationally efficient, particularly in the applications with fast implementations (e.g., FFT and DWT). They also show good recovery performance in some applications such as joint communication channel/symbol estimation [51], compressive imaging [52] and dynamic compressive sensing problem [53]. Although this family of algorithms is a powerful method for CS recovery, it may diverge in some cases, particularly for generic matrix \( \Phi \) where the entries of \( \Phi \) are not independent and identically distributed (i.i.d.) elements [54]. In addition, in [50] an adaptive-damping method has been proposed to prevent the divergence, however, theoretical guarantees are lacking.

- **Bayesian Algorithms:** Bayesian algorithms are one of the powerful families of algorithms. Mainly, these types of algorithms can be classified into the following two subcategories:

  - **Bayesian Greedy Algorithms:** The algorithms in this subcategory are the counterpart of the greedy algorithms category. Among these algorithms the most popular are the Bayesian pursuit algorithms [7], [55], [56]. Particularly, the Bayesian pursuit algorithm (BPA) in [7] used Bayesian hypothesis testing (BHT) for the detection and recovery of the supports in
the context of sparse representations. While this algorithm enjoys the simplicity of the greedy pursuit algorithms, it also utilizes the Bayesian approach to optimally select the active atoms of the dictionary matrix $\Phi$. BPA [7] uses the correlations between measurement vector $y$ and the columns of matrix $\Phi$ and applies a binary BHT to obtain an activity rule in which the correlations are compared with a threshold. This activity rule is then used for the detection and recovery of the supports. Recently, an algorithm using BHT with belief propagation, called BHT-BP, has been introduced in noisy sparse recovery [57]. BHT-BP utilizes the nonparametric belief propagation (n-BP) for the detection of the supports. This algorithm uses the low-density parity-check codes (LDPC)-like measurement matrix whose compressing capability performance is worse than that of the dense matrices. However, LDPC-like matrices are capable of fast generation of CS measurements.

– Sparse Bayesian Learning (SBL) Algorithms: Sparse Bayesian learning algorithms (SBL) [58]–[64] are another subcategory of Bayesian algorithms. As the SBL algorithms perform well in the sparse signal reconstruction even when the columns of $\Phi$ are highly coherent, they have a wide applications in direction of arrival (DOA) estimation, neuroelectromagnetic source localization, and feature selection in bioinformatics.

2.3 Block Sparse Signal Recovery: Models and Algorithms

In some applications, the unknown signal to be estimated (i.e., $w$ in (2.1)) has an additional structure. If such structure is exploited, the recovery performance can be further improved. A block-sparse signal, with clustered nonzero samples, is an impor-
tant structured sparsity [65]–[70]. Block-sparsity has a wide range of applications in multiband signals [71], audio signals [72], structured compressed sensing [73], and the multiple measurement vector (MMV) model [74]. The group/block structure implies that the original unknown signal $w$ consists of a sequence of blocks, i.e.,

$$
w = \left[ w_1, \ldots, w_{d_1}, \ldots, w_{d_{g-1}+1}, \ldots, w_{d_g} \right]^T,
$$

where $w[i]$ denotes the $i$th block with length $d_i$ which are not necessarily identical. In the block partition (2.5), only $k \ll g$ vectors $w[i]$ have nonzero Euclidean norm. The linear model (2.1) with the block partition (2.5) represents the general mathematical model of block sparsity which is called the canonical block-sparse model [67], [68].

Given the a priori knowledge of block partition, a few algorithms such as Block-OMP [65], mixed $\ell_2/\ell_1$ norm-minimization [66], group LASSO [67], model-based CoSaMP [68], and Group Basis Pursuit [75] work effectively in the block-sparse signal recovery. For instance, the mixed $\ell_2/\ell_1$ norm-minimization which is the extension of (2.3) for block-sparse regularization can be written as

$$
\hat{w} = \arg\min_w \beta \|y - \Phi w\|_2^2 + \tau \|w\|_{1,2},
$$

where $\|\cdot\|_{1,2}$ represents the mixed $\ell_2/\ell_1$-norm with

$$
\|w\|_{1,2} = \sum_{i=1}^g \|w[i]\|_2, \quad \|w[i]\|_2 = \sqrt{w[i]^T w[i]}.
$$

It can be seen that when $d_i = 1, \forall i$, the optimization formulation in (2.6) is reduced to the traditional $\ell_1$-norm based formulation in (2.3). These algorithms require the knowledge of the block structure (e.g. the location and the lengths of the blocks) in
However, in many applications, such prior knowledge is often unavailable. For instance, the accurate tree structure of the coefficients for the clustered sparse representation of the images is unknown \textit{a priori}. The impulsive noise estimation in Power Line Communication (PLC) is often cast into a block-sparse signal reconstruction problem, where the impulsive noise (i.e. signal $w$) occurs in bursts with unknown locations and lengths [1], [76].

To recover the structure-agnostic block-sparse signal, some algorithms, e.g. CluSS-MCMC [2], BM-MAP-OMP [3], Block Sparse Bayesian Learning (BSBL) [4], pattern-coupled SBL (PC-SBL) [5], [77], have been proposed recently, which require less \textit{a priori} information. For instance, in BSBL model [4], the overlapping structure of the covariance matrix is converted into a block diagonal structure using an expanded model, hence, the conventional BSBL algorithm can be directly applied. In addition, in [5] the same problem of reconstructing structure-agnostic block-sparse signals has been addressed. Unlike the conventional sparse Bayesian learning (SBL) framework, a coupled hierarchical Gaussian framework has been considered in [5], [77] to recover the structure-agnostic block-sparse signals. Therefore, the dependent hyperparameters encourage the clustered patterns and eliminate isolated coefficients whose pattern is different from that of its neighboring coefficients. In a recent work presented in [6], a nested SBL algorithm has been proposed for reconstructing the block-sparse signal with intra-block correlations between the samples of the blocks. This algorithm has been derived using a monotonically convergent nested Expectation Maximization (EM) and a Kalman filtering based learning framework. However, nested SBL requires the length and the number of blocks in the block sparse signal. Also, it performs well with some specific measurement matrix $\Phi$, whose entries are randomly chosen from Bernoulli ($\{+1,-1\}$) distribution.

All the above mentioned algorithms use the i.i.d. model to describe the cluster
structure of the nonzero elements of the unknown signal, which restricts their applicability and performance. Because many practically important signals, e.g. the impulsive noise in PLC, do not satisfy the i.i.d. condition, it is necessary to develop reconstruction algorithms for block-sparse signals using a more adequate signal model. Also, in the above mentioned algorithms, it is likely to choose unreliable column set of the matrix $\Phi$, which may result in inappropriate selection of nonzero elements of the signal $w$. Hence, it is necessary to design an adaptive method to select the most probable support set based on the underlying structure of the signal. The ability of the algorithm to automatically tune up the signal (i.e. $w$) model parameters is important, particularly when working with real-world datasets, but it is not provided by most of the existing block-sparse signal recovery algorithms (e.g., [2]–[6], [77]).

To tackle the above mentioned problems, in Chapter 3, we propose a novel iterative Bayesian algorithm (Block-IBA) which is based on MAP estimation and iterative Expectation Maximization. More importantly, most existing sparse- and block-sparse signal recovery algorithms utilize the signal estimation rather than support detection. However, recent research studies show that the existing estimation-based algorithms, e.g. Lasso [20], leaves a potentially large gap between their best performances and the theoretical limit for the noisy support recovery [78]–[80]. Hence, it is important to develop algorithms that use joint detection-and-estimation procedure [81], [82] to provide a robust support recovery against noise for block-sparse signals. Motivated by this, in Chapter 4, we develop a novel block Bayesian hypothesis testing algorithm (BBHTA) that takes a joint detection of support and estimation of amplitudes for block-sparse signal recovery. However, in the following section we will review the two important applications of block-sparse signal recovery. The first application, which is widely used in OFDM communication systems, is the bursty impulsive noise reduction using the block-sparse signal recovery. The second application is the block-sparse
image reconstruction in magnetic resonance imaging (MRI).

2.4 Sparse Signal Recovery and Its Applications

Sparse signal recovery, which is a solution to the problem based on the general linear model in (2.1), frequently occurs in the fields of signal processing, statistics, neuroscience and machine learning. Examples of common applications, among many others, include

- compressed sensing [83], [84]
- sparse component analysis (SCA) [85],
- sparse representation [13], [86]–[88],
- source localization [89], [90],
- direction of arrival (DOA) estimation [91],
- EEG/MEG source localization [92]–[94]
- adaptive signal processing [95]
- array signal processing [96]–[98]
- high-dimensional statistics [99], [100]
- wireless telemonitoring of physiological signals [101]–[104]
- wireless sensor networks [105]–[107]
- pattern recognition [108]–[110]
- rapid MR imaging [111], [112]
• radar imaging [113], [114]

• speech and audio processing [115]–[117]

• medical data analysis [118]–[120]

• astronomical data analysis [121]–[124]

• exploration seismology [125], [126]

• sparse models in economics and finance [127]

• neuronal data analysis [128]–[130]

In the following subsections, two simple applications of sparse signal recovery are presented. For introducing these applications, the model in (2.1) is used.

2.4.1 Block Sparse Impulsive Noise Reconstruction in Communication Systems

One of the important applications of sparse signal recovery is the block-sparse reconstruction of impulsive noise in wireless and wired communication systems. In this application, the original signal \( \mathbf{w} \) (i.e., impulsive noise), is sparse in the time domain. Thus, we have \( \Phi = \mathbf{F}_{\text{partial}} \) in (2.1), where \( \mathbf{F}_{\text{partial}} \) is the partial Discrete Fourier Transform (DFT) matrix, which is built by appropriately selecting the rows of DFT matrix \( \mathbf{F} \). Therefore, the model in (2.1) can be rewritten as

\[
\mathbf{y} = \mathbf{F}_{\text{partial}} \mathbf{w} + \mathbf{n}, \tag{2.8}
\]

where \( \mathbf{F}_{\text{partial}} \in \mathbb{C}^{N \times M} \) is the sensing matrix. To reconstruct \( \mathbf{w} \), a sparse signal recovery algorithm can be used to solve the underdetermined inverse problem in (2.8).
Once, the sparse signal $w$ has been recovered, it is subtracted from the received signal to mitigate the effect of impulsive noise $w$ on the communication system (see Chapter 5). Compared to the other impulsive noise reduction methods in communication systems, the advantage of sparse signal recovery is that if the sparse signal recovery algorithm accurately estimates the impulsive noise, it can completely cancel the effect of impulsive noise in the receiver and bring a significant gain in the performance of the communication system.

In some applications such as Power Line Communication (PLC), the impulsive noise $w$ has a block-sparse or bursty structure as described in (2.5), which is capable of erasing several symbols in the received signal. However, the existing receivers do not exploit the block-sparse structure of the impulsive noise and they mostly use the time-domain interleaving (TDI) and time-frequency-domain interleaving (TFDI) [131], [132] to mitigate the effect of impulsive noise. These techniques are ineffective at low to moderate signal to noise ratio (SNR). Moreover, these interleavers require a large memory to store the continuous time-domain signal. Collecting a large number of time-domain signal samples also introduces a long delay in the receiver detection procedure [133]. In Chapter 3, we present a novel iterative Bayesian algorithm (Block-IBA), which exploits the structure of the bursty impulsive noise to accurately estimate the impulsive noise and to avoid the delay due to TDI.

### 2.4.2 Block Sparse Image Reconstruction in MRI

In this application, $Z \in \mathbb{R}^{n \times K}$ is the original image to be recovered. Assume that we have $m$ linear measurement $Y \in \mathbb{R}^{m \times K}$ of the unknown original image $Z$ as

$$Y = \Phi Z, \quad (2.9)$$
where $\Phi$ comprises $m$ ($m < n$) rows drawn from an $n \times n$ orthogonal transform matrix (e.g., a Fourier transform matrix). Images usually demonstrate the block-sparsity structures, particularly on over-complete basis such as wavelet or discrete cosine transform (DCT) basis. Assume $Z$ is block-sparse in an appropriate transform domain $W$ (e.g., wavelet transform domain), described by the orthogonal transform matrix $W \in \mathbb{R}^{n \times n}$. Thus, we have $S_z = WZ$, which is block-sparse. Using (2.9), we have

$$Y = BS_z,$$

(2.10)

where $B = \Phi W^{-1}$. To recover $S_z$, given $\Phi$ and $W$, a block-sparse signal reconstruction algorithm can be used to solve the underdetermined problem (2.10). Having recovered the block-sparse vector $S_z$, the original image $Z$ can be immediately obtained by $Z = W^{-1}S_z$. Note that, if $W$ is an orthogonal wavelet matrix we have $W^{-1} = W^T$.

In Chapter 3, we will present a novel block-sparse iterative Bayesian algorithm (Block-IBA) which is capable of accurately recovering the MRI images (from their block-sparse representations in wavelet domain) at a very low computational cost.

### 2.5 Impulsive Noise Models in Wireless and Power line Communication (PLC) Systems

In digital communication systems, the information bits are transmitted from the transmitter to the receiver via a channel or physical medium. The major impairments to a wireless or wired communication system are multipath propagation effects of the communication channel and noise. Noise is defined as random fluctuations in the received signal caused by disturbances from natural or man-made sources. In the design of the communication systems, the effects of the random fluctuations are
considered as an additive noise process. The samples of the additive noise process are identically and independently chosen from a complex Gaussian distribution. This model is called additive white Gaussian noise (AWGN), which is the most widely used model in the contemporary communication systems [134].

Although AWGN is an appropriate model for thermal noise in the receiver circuitry, it is unable to describe the characteristics of the noise in modern communication systems. The real-life measurements from the terrestrial wireless installations, in various frequency bands up to 4 GHz, show that the additive noise is impulsive with power spectral density (PSD) reaching up to 40 dB above the thermal background noise PSD [135]–[140]. The noise in both narrowband and broadband PLC is not only highly impulsive but also block-structured (bursty) [141]–[143]. The noise in power substations has also been shown to be impulsive with correlated samples of impulses [144]. Impulsive noise is usually generated by various sources such as partial discharges, corona noise, motor ignitions, radio broadcasting services and other wireless equipments, and switching transients of AC appliances. In PLC network, the impulsive noise is mainly generated by switching transient of various AC appliances in home and business connected to the PLC network [143], [145].

Impulsive noise is generally classified into aperiodic (socalled asynchronous in PLC systems) and periodic impulsive noise [143]. Aperiodic impulsive noise consists of random impulses with short duration between some microseconds and milliseconds. In fact, the PLC network acts as an antenna and picks up this interference from the in-band and aliased signals of wireless equipments and broadcasting services [145]. Moreover, the uncoordinated interference from neighboring PLC devices is generally categorized as asynchronous impulsive noise in narrowband PLC networks [142]. The PSD of these impulses can be up to 50 dB stronger than the background noise PSD [143]. Thus, the aperiodic impulsive noise is one of the major impairments to reliable
communications in broadband PLC [143]. In contrast, periodic impulsive noise (also known as cyclostationary noise in PLC systems) occurs periodically in time with longer bursts. This type of noise, also known as block-sparse impulsive noise or bursty impulsive noise [146], consists of bursts spanning over one or more received symbols and is capable of erasing several received symbols in the communication system.

In general, impulsive noise model is controlled by a process that switches from zero outputs to burst outputs of certain lengths. Therefore, the appropriate model of the impulsive noise can be completely determined by the switching law and the sample distributions. The sampling frequency controls the number of samples in impulses which could be one sample or a burst of samples. This in turn results in an i.i.d. model or non-i.i.d. bursty model of the impulsive noise. Many memoryless models such as general Gaussian Mixture (GM) [147], Bernoulli-Gaussian [148], symmetric alpha stable [149] and Middleton class A, B and C models [150] consider the sparse samples of the impulsive noise to be i.i.d. These i.i.d. models of impulsive noise are appropriate for modeling the random impulses from uncoordinated interferers, e.g. aggressive spectrum reuse, or non-communicating electronic devices in PLC networks.

Middleton used Poisson point processes (PPP) to model the random impulses, resulting in “Middleton class-A” and “Middleton class-B” noise models. These models have been recently reviewed in [150]. Using the spatial and temporal PPPs, the Middleton models have been extended to modeling of impulsive noise in many wireless and PLC networks. In [151], [152], it has been shown that the interference in the multiple access communication systems, where the interferers are assumed to follow Poisson distribution in the entire plane, has a symmetric alpha stable model. Particularly, this symmetric alpha stable model is applicable to interferers with homogeneous Poisson fields. In fact, the impulsive nature of the multiple access interference con-
firms the appropriateness of the symmetric alpha stable model. This model of noise is appropriate for some wireless networks, e.g., wireless sensor and ad hoc networks. When the Poisson field of interferers are confined to a finite-area interference region with or without a guard zone around the receiver, the interference statistics follows a Middleton class-A distribution [147]. This model is applicable to the co-channel or out-of-cell interference in cellular networks and also dense WiFi networks. In the context of Poisson clusters scenarios, where the interferers are clustered in space and each interferer is distributed based on an independent spatial Poisson process, the interference follows a Gaussian mixture distribution [147]. This model is appropriate to represent the out-of-cell interference in cellular networks with user clustering and two-tier femtocell networks. The characteristics and applications of these models are presented in Table 2.1.

Table 2.1: Statistical models for i.i.d. impulsive noise and their applications in communication systems.

<table>
<thead>
<tr>
<th>Model</th>
<th>Characteristics</th>
<th>Application</th>
</tr>
</thead>
<tbody>
<tr>
<td>Symmetric Alpha Stable</td>
<td>Characteristic function: ( \Phi (\omega) = \exp (-\sigma</td>
<td>\omega</td>
</tr>
<tr>
<td>Gaussian Mixture</td>
<td>( p(x) = \sum_{k=0}^{M-1} \pi_k N(x; 0, \gamma_k) ) ( M ): number of components ( \pi_k ): component probability ( \gamma_k ): component variance</td>
<td>Cellular networks with user clustering Two-tier femtocell network</td>
</tr>
<tr>
<td>Middleton class-A</td>
<td>( p(x) ) in Gaussian Mixture with ( M = \infty ) ( A ): impulsive index ( \Gamma ): Gaussian to impulsive power ratio ( \pi_k = e^{-A \Delta_k^k} / k! ) ( \gamma_k = \gamma (k/A + \Gamma) / (1 + \Gamma) ) ( \gamma ): variance</td>
<td>Cellular networks Dense WiFi networks</td>
</tr>
</tbody>
</table>
In many applications such as PLC, the noise has bursty structure with correlated samples and hence, the noise model requires some memory to represent the correlation between the samples that in turn captures the bursty structure of the noise. Gaussian Hidden Markov model (GHMM) is an appropriate and tractable model which is widely used to model bursty structure impulsive noise in PLC networks [143]. The characteristics of GHMM can be given as

\[
p(z_k \mid z_{k-1}, \cdots, z_0) = p(z_k \mid z_{k-1}),
\]

\[
p(z_k = j \mid z_{k-1} = i) = T_{ij}, \tag{2.11}
\]

\[
p(x_j \mid z_j) = \mathcal{N}(x_j; 0, \gamma_j),
\]

where \(T\) and \(\gamma_j\) are the state transition matrix and variance in state \(j\). In (2.11), a Markov chain model has been used for the Gaussian Mixture (GM) state \(z_k\). In Chapter 5, we will use Bernoulli-Gaussian hidden Markov model (BGHMM) to represent the block-sparse impulsive noise in wireless communication and PLC systems. BGHMM is a special form of GHMM, which is capable of approximating rather precisely the PDF of additive non-Gaussian noise in many communication systems such as PLC.

### 2.6 OFDM Modulation in Digital Communication Systems

Orthogonal frequency division multiplexing (OFDM) is widely adopted in many wireless communication standards (e.g., IEEE802.11a, DVB family, IEEE802.16e (WiMAX), and 3GPP Long Term Evolution (LTE) [153]) and Power Line Communication (PLC) standards (e.g., PRIME [154], G3-PLC [155], and IEEE1901 [145]).

In OFDM, data is sent in parallel on a number of different frequencies. In fact, the transmission bandwidth, in OFDM, is divided into many subbands. The
Center frequency of each subband is called subcarrier. Hence, data is transmitted over several subbands simultaneously. In the context of coherent detection, channel estimation is necessary at the receiver. To serve this purpose, some pilot subcarriers, carrying known symbols, are inserted in the transmitted OFDM signal. Further, null subcarriers (with zero power) are loaded at the edges of the spectrum of the transmitted OFDM symbol to reduce the out-of-band emissions, facilitate cut-off filter requirements, and enable the interference reduction. In many OFDM standards, adaptive loading is used to maximize the total rate of the system using the adaptive modulation [156]. In this technique, the data rate and power of each subband is varied based on the subband’s gain, or signal to noise ratio (SNR), to increase the throughput of the system.

Conventional OFDM systems are mostly designed based on the DFT, which is simply implemented by the fast Fourier transform (FFT). Figure 2.1 shows the simplified block diagram of the transmitter and receiver of a typical OFDM system. \( N \) data symbols are serially transmitted and then this sequence of symbols is converted, from serial to parallel (S/P), to form an appropriate parallel vector for the IFFT input. Hence, the sequence of \( N \) data symbols are pre-coded by the IFFT of length \( N \) to form the time-domain signal \( x_t = F^H x \). The parallel time-domain signal \( x_t \) is converted to serial by a P/S converter and then is converted to analogue signal using a digital to analogue (D/A) converter. The resulting analogue signal is then transmitted through a multipath (or frequency selective) channel \( H \) and contaminated by interference (e.g., AWGN and impulsive noise) at the receiver. At the receiver, after analogue to digital (A/D) conversion, the time-domain received signal is given by

\[
    r = Hx_t + n = HF^H x + n, \quad (2.12)
\]
where \( r, x, \) and \( n \in \mathbb{C}^{N \times 1} \) are the received OFDM signal, the transmitted symbols, and the interference, respectively. \( H \in \mathbb{C}^{N \times N} \) is a cyclic matrix, formed by the discrete-time channel impulse response \( h \).

The received signal \( r \) is demodulated by applying the FFT to obtain the frequency-domain received signal as

\[
y = Fr = H \odot x + Fn,
\]

where \( \Lambda = FH^H = \text{diag}(H), \) \( H = \sqrt{N}Fh, \) and \( \odot \) denotes the Hadamard (element-wise) product. Finally, the received symbols are detected by applying minimum distance decoding on each symbol. The advantage of OFDM system is that each
subcarrier (e.g., subcarrier $k$) experiences a frequency-flat channel (e.g., $H_k$). This in turn, significantly facilitates the equalization of the frequency-selective channel. However, the abovementioned receiver is only appropriate under AWGN channel. This is because the statistical properties of AWGN remains unchanged under the unitary matrix transformation $F$. This receiver, however, is inappropriate in the context of the additive non-Gaussian noise (e.g., impulsive noise) and its performance is significantly degraded [157]–[159].

Single Carrier (SC) modulation is one of the basic modulation schemes employed in most communication systems. SC modulation encodes the transmitted data by varying the amplitude, phase, or frequency of the carrier. SC modulations such as frequency shift keying (FSK) and quadrature phase shift keying (QPSK) have been widely used in narrowband PLC [160]. However, due to frequency-selective fading channel, these modulation schemes are inappropriate for broadband PLC. This is because the multipath effect of the PLC channel results in a significant intersymbol interference (ISI) when the SC modulation is used. Therefore, it is necessary to employ the expensive channel equalizers which increase the complexity of the communication system (e.g., PLC system). In contrast, the equalization procedure of the multipath channel in OFDM systems is significantly facilitated as a wideband multipath channel is transformed into a set of parallel frequency-flat channels. Another drawback of SC modulation is that it is not robust against the impulsive noise for high SNR range. To investigate the effect of the impulsive noise on the communication system, we compare the standard OFDM system with the SC system presented in Figure 2.2. As it is seen from Figure 2.1, the modulation in OFDM system is performed by IFFT. Mathematically, this modulation is equalent to $F^H x$, i.e., multiplication of the frequency-domain transmitted signal $x$ by unitary matrix $F^H$. This distributes the information of each symbol $x_i$ across a much longer OFDM symbol with $N$ symbol
<table>
<thead>
<tr>
<th>Single Carrier (SC) System:</th>
<th>OFDM System:</th>
</tr>
</thead>
<tbody>
<tr>
<td>N symbols</td>
<td>Interference (AWGN+Impulsive)</td>
</tr>
</tbody>
</table>

Figure 2.2: The performance of standard OFDM system versus single carrier (SC) system in the impulsive noise environment.
times duration. This process provides a kind of diversity encoding, for impulsive noise channels, which is presented in [158], [159]. Moreover, at the receiver the effect of impulsive noise energy is spread over all \( N \) subcarriers, due to the operation of DFT, which in turn reduces the impact of the impulsive noise at any time sample. However, in the SC system without the DFT, the energy of the impulsive noise concentrates in time which is capable of erasing that symbol transmitted at that time sample. Hence, the symbol error rate curve in Figure 2.2 demonstrates two regions of different performances. In the first region, SC outperforms the standard DFT-based OFDM. This is because, in this region, the SNR is relatively low and as a result the impulse energy is high enough to erase the entire OFDM symbol. In contrast, the effect of impulse energy is localized in SC and thus SC outperforms the OFDM receiver in the first region. In the second region, the SNR is high and consequently the impulse energy is sufficiently low. The effect of this low-energy impulse is further reduced by spreading over the entire OFDM symbol and thus the OFDM receiver outperforms the SC.

Although DFT-based OFDM system outperforms the SC system in the high SNR regime under impulsive noise environment, it is not optimal receiver under impulsive noise environment. In fact, in standard OFDM system each subcarrier is demodulated independently. However, when the time-domain noise is impulsive, its samples in the frequency-domain are highly dependent and thus tone-by-tone demodulation is no longer optimal. Therefore, it is necessary to redesign the receiver in order to mitigate the effect of impulsive noise. Hence, in the following section we will briefly study some different approaches of impulsive noise reduction techniques which are commonly used in the receiver of wireless and wired communication systems to alleviate the effect of impulsive noise.
2.7 Impulsive Noise Receivers in OFDM Systems

Prior research work to mitigate the effect of impulsive noise in OFDM systems has been done at both transmitter and receiver sides. Particularly, the transmitters have been redesigned to cope with the block-sparse impulsive noise. For instance, the transmitters in the existing communication system standards (e.g., narrowband PLC) utilize the forward error correction (FEC) coding and frequency-domain block interleaving to reduce the effect of block-sparse impulsive noise. In some narrowband PLC standards such as G3-PLC [155] and IEEE standard 1901.2 [161], multilayer FEC codes (i.e., concatenation of convolutional, Reed-Solomon, and repetition codes) have been used to improve the error correction capability of FEC codes in impulsive noise environment. Although multilayer FEC coding improves the reliability of the system, it reduces the throughput since it requires bandwidth expansion. Moreover, majority of PLC standards used bit-level interleaving, or frequency-domain interleaving prior to IFFT to enhance the performance of the communication system. The sample-level time-domain interleaving (TDI) and time-frequency-domain interleaving (TFDI) have been proposed in [131], [132] to reduce the impact of bursty impulsive noise by spreading the long noise bursts into shorter ones. Although TDI and TFDI have superior performance over frequency-domain interleaving techniques, they are only effective at high SNR, but not at low to moderate SNR. Moreover, these interleavers require a large memory to store the continuous time-domain signal. Collecting a large number of time-domain signal samples also introduces a long delay in the receiver detection procedure [133]. If the structure of the block-sparse impulsive noise is exploited by the receiver, the delay due to TDI and TFDI techniques in the receiver detection can be avoided and a better performance can be expected.

At the receiver side of OFDM systems, different approaches have been pro-
posed in the literature to mitigate the effect of impulsive noise. For the simplicity of the presentation, these approaches are categorized into different categories which are presented in the following subsections.

2.7.1 Time-Domain Preprocessors Receivers

These types of receivers detect the amplitude of impulsive noise using simple threshold tests or nonlinear estimators. In this approach, the received time-domain signal is preprocessed at the receiver front-end of the conventional OFDM demodulator (i.e., DFT demodulator) by clipping or blanking (or combined clipping/blanking) [162]–[165] or nonlinear MMSE estimation [158]. In the thresholding techniques, based on the central limit theorem with sufficiently large number of subcarriers, the time-domain OFDM samples are considered as i.i.d. Gaussian variables which facilitates the detection of impulses using a simple threshold test. In the nonlinear estimator techniques, for instance, in [158] the time-domain OFDM sample at time instant \( t \) (i.e., \( u_t \)) is treated as a Gaussian random variable and then the MMSE estimate of \( u_t \) given the received sample \( r_t \) and the impulsive noise model, i.e. \( \mathbb{E}\{u_t \mid r_t\} \), is derived. After pre-processing, the result is fed to a DFT-based OFDM demodulator for decoding. In the memoryless nonlinearity techniques (or thresholding techniques), the time-domain OFDM sample is compared with a threshold \( T \): if the magnitude of the sample exceeds \( T \), its magnitude is either clipped or blanked. In case of combined clipping/blanking there are two thresholds \( T_1 \) and \( T_2 \) (\( T_1 < T_2 \)), where \( T_1 \) is the clipping threshold and \( T_2 \) is blanking threshold. Here, if the magnitude of the sample is between \( T_1 \) and \( T_2 \), it is clipped to \( T_1 \) and if it exceeds \( T_2 \), it is blanked. Although these techniques offer low computational complexity, they perform poorly at a certain low signal to impulsive noise ratio (SINR) [159] and also for higher order modulations [158]. This is because these techniques ignore the signal space diversity
encoding offered by OFDM modulation [159] and thus, their performance degrades at low SINR, when the impulses power is comparable to the OFDM signal power. Hence, at low SINR, the receiver is more prone to erroneously threshold the actual OFDM sample that is not contaminated by an impulse. Moreover, clipping/blanking acts as a nonlinear sample-by-sample preprocessor in the time domain, resulting in the distortion of signal constellation and intercarrier interference (ICI) in the frequency domain [166]. Also, these techniques are unable to eliminate the variation in the received signal, caused by frequency-selective channels, and thus their applications to the real-world systems are significantly limited. To overcome these problems, some iterative receivers have been proposed which are discussed in the following subsection.

### 2.7.2 Iterative Receivers

The approach in these receivers is to iterate between preprocessors in time-domain and OFDM decoding in the frequency-domain. For instance, in the current iteration a preprocessor such as clipping or blanking is applied to the time-domain received signal, which is followed by symbol detection and time-domain correction (or syndrome decoding) for the next iteration [167], [168]. These techniques have low computation complexity, however, they have the same limitations as the time-domain preprocessors receivers. In addition, these techniques need to be implemented in an ad hoc manner to increase the convergence rate. In [157], [169], the authors used the central limit theorem and approximated the impulsive noise and OFDM signal with Gaussian distributions in the frequency domain and time domain, respectively. Then, using these models, they have designed linear MAP and MMSE estimators and detectors in the time and frequency domain and sequentially and iteratively applied them to mitigate the effect of the impulsive noise in each iteration. Despite the significant performance gains of these techniques, the main drawback of them is that they are parametric
approaches and they require tuning the simulation parameters, which limits their applications and implementations in practical systems. As an alternative technique, the combination of TDI technique after IDFT and blanking nonlinearity at the receiver has been proposed in [170]. Although this approach utilizes zero forcing (ZF) and MMSE equalizers to compensate the multipath effect, channel estimation and synchronization are required before deinterleaving.

2.7.3 Factor Graph Receivers

The factor-graph-based receivers [171] are appropriate to jointly estimate channel taps, symbols, and bits at a low computation complexity. The nodes of the factor graph exchange the messages, generally in the form of pdfs, using the belief propagation methods such as the sum-product algorithm [172]. Since the factor graph representation of the OFDM system model has the loopy nature, the sum-product algorithm is unable to exactly compute the various marginal functions from global functions and thus a variety of approximation methods have been proposed [173]–[176]. For instance, in [176], the generalized approximate message passing (GAMP) algorithm [177] has been combined with a turbo soft-input soft-output decoder to achieve near-optimal joint clustered-sparse-channel estimation and decoding of bit-interleaved coded-modulation (BICM)-OFDM with complexity of $O(N \log N)$. Recently, [178] extended the factor-graph-based receiver in [176] to an impulsive noise receiver, yielding a near-optimal performance. This receiver, which is based on the GAMP algorithm [177], jointly estimates the channel taps, the impulse noise samples, the finite-alphabet symbols, and the unknown bits. Although these receivers have a low implementation complexity due to fast implementation of FFT, the underlying algorithm (i.e., message passing algorithm) may diverge due to the non-i.i.d. entries of the Fourier matrix [54]. Moreover, the proposed receiver in [178] assumes the
prior knowledge of the impulsive noise parameters, however, the information about these parameters may not be available in practical settings and they might have to be estimated by the receiver. In Chapter 5, we propose a novel Block-IBA receiver that automatically learns the unknown parameters of the impulsive noise using a maximum a posteriori (MAP) estimation.

2.7.4 Sparse Impulsive Noise Cancellation Receivers

Based on the assumption of the sparse structure of the impulsive noise in the time domain, these approaches attempt to recover the impulsive noise utilizing the received signal on the known tones (e.g., either null or pilot tones). The reconstructed impulsive noise vector is then subtracted from the time-domain received signal and the resulting signal is fed to DFT receiver for demodulation and decoding. In [179], the author utilized the relationship between the DFT transform in OFDM with known tones and error control coding (i.e., Reed-Solomon coding) and proposed to estimate the impulsive noise vector using the frequency algebraic interpolation techniques. The extension of this approach to more general settings was later proposed in [180], [181]. This impulsive noise reduction method uses algebraic sparse reconstruction techniques with precoding and algebraic frequency domain interpolation reinforced by Reed-Solomon coding. But this method is very sensitive to the AWGN (background noise) level and quantization errors. Considering the limitations of these methods and to cope with the very same problem, a compressed sensing (CS)-based sparse reconstruction technique has been proposed in [182], [183], which models the time-domain impulsive noise as a sparse vector and uses $\ell_1$ norm-minimization based CS reconstruction to estimate the vector from the observed OFDM null and pilot tones. However, this technique is computationally complex and only exploits the sparsity information without taking into account the a priori statistical information.
of the impulsive noise. The extension of this method referred to as block-based CS has been proposed in [76], which uses mixed $\ell_2/\ell_1$ norm-minimization to estimate the block-sparse impulsive noise. These methods are feasible only for very sparse impulsive noise signals, e.g. one impulse in a 256-subcarrier OFDM system with 30 known subcarriers [8], [9], [182], and are infeasible for practical sparsity rates [182], [8].

More advanced impulsive noise estimation and cancellation approaches, which also exploit the sparsity information of the time-domain impulsive noise, have been proposed in [8], [9]. The method in [8] combines the joint symbol detection and impulsive noise estimation using sparse Bayesian learning (SBL). The SBL receiver uses the TDI in the receiver to spread the long bursty impulses into shorter ones. However, using TDI introduces a long delay in the receiver detection procedure and requires a large memory to store the continuous time-domain signal [133]. The approach in [9] exploits the structure of the sensing matrix and the null tones in the OFDM systems to estimate and cancel impulsive noise. This receiver assumes the parameters of the impulsive noise to be known, however, the prior knowledge of the noise parameters is often unavailable in practical systems and might have to be estimated by the receiver. In Chapter 5, we propose an OFDM receiver that uses the information in null tones to estimate and remove the bursty impulsive noise, and uses the information from data tones to perform joint OFDM symbol detection and noise estimation. Although these techniques resemble those in [8], [9], the proposed receiver uses a novel Block-IBA to exploit the block sparsity of the bursty impulsive noise for optimal estimation, which reduces the computational cost and removes the delay caused by TDI in [8]. In contrast to [9] that assumes the parameters of the impulsive noise to be known, the proposed receiver automatically estimates the block-sparse impulsive noise parameters using MAP estimation.
2.8 Open Problems Identified in the Literature

The following open research problems have been identified in the literature:

1. Most existing algorithms [2]–[6], [77] for reconstructing the structure-agnostic block sparse signal models use i.i.d. samples to represent the block structure of the nonzero elements of the signal, which limits their applications and degrade their performances. The i.i.d. models are not appropriate for practical applications, e.g. bursty impulsive noise in PLC, in which the samples are no longer statistically independent. Hence, it is necessary to develop block-sparse signal recovery algorithms using a more realistic signal model, e.g. Bernoulli-Gaussian hidden Markov model (BGHMM) [143]. More importantly, most algorithms assume the prior knowledge of the signal model parameters, while in practical settings (particularly when working with real-life datasets) this information is often unavailable and it might have to be automatically estimated by the algorithm.

2. Most existing sparse- and block-sparse signal recovery and CS algorithms have been developed based on the signal estimation rather than support detection. However, recent research studies show that the existing estimation-based algorithms, e.g. Lasso [20], leaves a potentially large gap between their best performances and the theoretical limit for the noisy support recovery [78]–[80]. Based on this theoretical investigation, it is useful to develop algorithms that jointly detect and estimate the supports and amplitudes of the signal [81], [82], respectively to provide a robust support recovery against noise for block-sparse signals.

3. Block-sparse or bursty impulsive noise is a major impairment to wireless and
PLC networks. In the literature, different approaches have been proposed to mitigate the effect of bursty impulsive noise, from both transmitter and receiver sides. From the perspective of redesigning the transmitter, the time-domain interleaving (TDI) and time-frequency-domain interleaving (TFDI) have been proposed in [131], [132] to mitigate the effect of bursty impulsive noise. However, these techniques are only effective at high SNR and they also introduce a long delay in the receiver detection procedure [132], [133]. From the perspective of redesigning the receiver, different approaches have been proposed in the literature to mitigate the impact of impulsive noise including time-domain preprocessors receivers, iterative receivers, factor graph receivers, and sparse impulsive noise recovery receivers. The time-domain preprocessors only perform well at high signal to impulsive noise ratio (SINR) [159] and they perform poorly for higher order modulation [158]. In addition, the preprocessing techniques (e.g., clipping/blanking) introduce the distortion of signal constellations and intercarrier interference (ICI) in the frequency domain [166]. The same limitations of time-domain preprocessors receivers carry over into the iterative receivers. Besides, these receivers need to be implemented in an ad hoc manner to increase the convergence rate. The main drawback of the factor graph receivers is that the underlying algorithm, i.e., message passing algorithm, may diverge due to non.i.i.d. entries of Fourier matrix [54]. The sparse impulsive noise recovery receivers are only effective on very sparse impulsive noise signals, e.g. one impulse in a 256-subcarrier OFDM system with 30 known subcarriers [8], [9], [182], and are infeasible for practical sparsity rates [182], [8]. In particular, the more advanced SBL receiver [8] uses the TDI in the transceiver, which introduces a long delay in the receiver detection procedure and requires a large memory to store the continuous time-domain signal. Hence, it is important to
design a specific receiver, for mitigating the effect of bursty impulsive noise, which resolves the aforementioned limitations.

In the following chapters, we aim to fill the aforementioned gaps through proposing a novel block iterative Bayesian algorithm (Block-IBA) for reconstructing non-i.i.d. block-sparse signals, proposing block Bayesian hypothesis testing algorithm (BBHTA) for structure-agnostic block-sparse signal recovery, and designing a novel OFDM receiver for mitigating bursty impulsive noise in digital communication systems.

2.9 Conclusion

In this chapter, the relevant background in the literature including the models and applications of sparse and block-sparse signal recoveries, different statistical models for impulsive noise in wireless and PLC systems, and prior work on designing OFDM receivers in impulsive noise environment have been presented. Also, some research gaps and open problems have been identified in the literature, which are addressed in the following chapters. In particular, it has been found that the existing block-sparse signal recovery algorithms use i.i.d. models to describe the block structure of the nonzero elements of the signal, which limits their applicability and their performance. Hence, in the next chapter, using a more realistic Bernoulli-Gaussian hidden Markov model (BGHMM), a novel iterative Byesian algorithm (Block-IBA) is proposed to reconstruct the block-sparse signal with unknown structure.
Chapter 3
Iterative Bayesian Reconstruction of Non-IID Block-Sparse Signals

3.1 Introduction

Compressed sensing (CS) and sparse signal recovery aim to measure and recover the sparse signal $w$ from underdetermined systems of linear equations presented by (2.1). In practice, the sparse signal $w$ has additional structures. If such structures are exploited, the better recovery performance can be achieved. The clustered nonzero samples of a block-sparse signal (see (2.5)) is an important structured sparsity [65]-[70], with applications in block-sparse impulsive noise estimation in Power Line Communication (PLC) [76] and clustered-sparse channel estimation [176]. Assuming the knowledge of the block structure (e.g., the location and the length of the blocks) in (2.5), a few effective algorithms [65]-[68] have been developed. However, in many applications, e.g. the bursty impulsive noise in PLC [1], such prior knowledge is often unavailable. As a result, some algorithms requiring less a priori knowledge have recently been proposed [2]-[5]. But all these algorithms assume the i.i.d. block structure which is impractical in many applications, e.g. the bursty impulsive noise in PLC. Also, these algorithms do not employ an effective method to select the most probable support set based on the underlying structure of the signal. In addition, these algorithms lack an automatic tuning up of the signal (i.e., $w$) model parameters, which is important in dealing with real-world datasets.

To tackle the aforementioned shortcomings, this chapter presents a novel Block Iterative Bayesian Algorithm (Block-IBA) for reconstructing block-sparse signals with unknown block structures. Unlike the existing algorithms for block sparse signal re-
covery which assume the cluster structure of the nonzero elements of the unknown signal to be independent and identically distributed (i.i.d.), we use a more adequate Bernoulli-Gaussian hidden Markov model (BGHMM) to characterize the non-i.i.d. block-sparse signals commonly encountered in practice. Block-IBA effectively selects the nonzero elements of the block-sparse signal by a diminishing thresholding. Using simple tuning updates, derived by a maximum a posteriori (MAP) estimation method, Block-IBA automatically estimates the unknown parameters of the statistical signal model (e.g., the elements of state-transition matrix of BGHMM). The proposed Block-IBA reconstructs the supports and the amplitudes of block-sparse signal $w$ using an expectation maximization (EM) algorithm when its block structure is completely unknown. In the expectation step (E-step) the amplitudes of the signal $w$ are estimated iteratively whereas in the maximization step (M-step), the supports of the signal $w$ are estimated iteratively. To this end, we utilize a steepest-ascent algorithm after converting the estimation problem of discrete supports to a continuous maximization problem. Although the steepest-ascent algorithm has been used in the literature for recovering the sparse signals (e.g. [184]), investigation of this method is unavailable in the literature of block-sparse signal recovery. The global convergence of Block-IBA is analyzed and proved based on the non-i.i.d. property of BGHMM and error vector method. The proposed Block-IBA offers more reconstruction accuracy than the existing state-of-the-art algorithms for the non-i.i.d. block-sparse signals. This is verified on both synthetic and real-world signals, where the block-sparse signal comprises a large number of narrow blocks.

The rest of the chapter is organized as follows. In Section 3.2, we present the signal model. In Section 3.3, the optimum estimation of unknown signal $w$ using MAP solution is proposed. Based on the MAP solution, a novel Block-IBA is developed in Section 5.4. The estimation of signal model parameters is presented in Section
5.4.3. Section 3.6 analyzes the global and local maxima properties of the Block-IBA. Experimental results are presented in Section 3.7. Finally, conclusions are drawn in Section 3.8.

3.2 Signal Model

In this chapter, the linear model of (2.1) is considered as the measurement process of an underlying signal which is block sparse. The measurement matrix $\Phi$ is assumed known and its columns are normalized to have unit $\ell_2$-norms. Furthermore, we model the noise in model (2.1) as a stationary, additive white Gaussian noise (AWGN) process, with $n \sim \mathcal{N}(0, \sigma_n^2 I_N)$. To model the block-sparse sources ($w$), we introduce two hidden random processes, $s$ and $\theta$ [53], [184]. The binary vector $s \in \{0, 1\}^M$ describes the support of $w$, denoted $S$, while the vector $\theta \in \mathbb{R}^M$ represents the amplitudes of the active elements of $w$. Hence, each element of the source vector $w$ can be characterized as

$$w_i = s_i \cdot \theta_i,$$

(3.1)

where $s_i = 0$ gives $w_i = 0$ for $i \notin S$ and $s_i = 1$ gives $w_i = \theta_i$ for $i \in S$. In vector form, (3.1) can be written as

$$w = S\theta, \quad S = \text{diag}(s) \in \mathbb{R}^{M \times M}.$$  

(3.2)

To model the block-sparsity of the source vector $w$, we assume that its supports are correlated such that $s$ is a stationary first-order Markov process defined by two transition probabilities: $p_{10} \triangleq \Pr\{s_{i+1} = 1|s_i = 0\}$ and $p_{01} \triangleq \Pr\{s_{i+1} = 0|s_i = 1\}$. Moreover, it can be shown that in the steady state we have the following relation between the transition probabilities and the probabilities in a given state:
The two parameters \( p \) and \( p_{10} \) completely describe the state process of Markov chain. As a result, the remaining transition probability can be determined as \( p_{01} = \frac{p_{10}}{1 - p_{10}} \).

The length of the blocks of the block-sparse signal is determined by parameter \( p_{01} \), namely, the average number of consecutive samples of ones is specified by \( 1/p_{01} \) in Markov chain.

Intending to represent the PDF of the source signal \( w_i \) by a Gaussian Mixture distribution, we further assume that the amplitude vector \( \theta \) has a Gaussian distribution with \( \theta \sim \mathcal{N}(0, \sigma_\theta^2 I_M) \). From (3.1) it is obvious that \( p(w_i|s_i, \theta_i) = \delta(w_i - s_i\theta_i) \), where \( \delta(\cdot) \) is the Dirac delta function. Using the marginalization rule, we can remove \( s_i \) and \( \theta_i \) from \( p(w_i|s_i, \theta_i) \) to find the PDF of the \( w_i \)'s as

\[
p(w_i) = p\delta(w_i) + (1 - p)\mathcal{N}(w_i; 0, \sigma_\theta^2),
\]

where \( \sigma_\theta^2 \) is the variance of \( \theta \).

Equation (3.5) is the well known BGHMM which is a special form of Gaussian Mixture Hidden Markov model (GHMM). It has been used in the literature to represent the PDF of a large class of signals including that of [53]. It is shown in [178] that with properly tuned parameters, GHMM can approximate rather precisely the PDF of the additive noise in many communication systems such as PLC. This kind of noise is bursty (clustered) [142], [143] and highly impulsive, with the peak amplitudes up to 50 dB above the AWGN (or background noise) level [143]. Hence, its samples
are block-sparse and non-i.i.d. These results effectively demonstrate that the point-mass distribution at $w_i = 0$ and the hidden variables $s_i$ in BGHMM allow implicit expression of the block-sparsity of a general class of non-i.i.d. signals, including the MR image signal shown in Subsection 3.7.5. Based on this observation, we propose to use the BGHMM (3.5) as a model to develop a Bayesian reconstruction algorithm for a general class of non-i.i.d. block sparse signals.

Unlike the memoryless models such as Bernoulli-Gaussian model [148], [185], [186] which consider the sparse samples to be i.i.d., the BGHMM [143], [146], [187] with the first-order Markov chain model allows better description of the typical bursty nature of a class of non-i.i.d. sparse samples.

It is observed from (3.5) that the development of the amplitude vector $\theta$ is independent of the sparsity of the random process, $s$. Hence, some of the amplitudes are pruned out by inactive coefficients (those associated with $s_i = 0$). In fact, the nonzero amplitudes $\theta_i$ are the results of the amplitudes of $w_i$ conditioned on $s_i = 1$. The assumed Gaussian distribution of $\theta$ and the first-order hidden Markov model of $s$ lead to a simple BGHMM. Using higher-order Markov processes and/or more complex mixture of Gaussian model, the framework of the Block-IBA can easily be extended to more general cases. To reduce the complexity of the reconstruction algorithm to be developed, we focus on the first-order Markov processes and Bernoulli-Gaussian model in this work.

### 3.3 Optimum Estimation of $w$

To obtain the optimum estimate of $w$, we pursue a MAP approach. We first determine the MAP estimate of $s$ which maximizes the posterior probability $p(s|y)$. After estimating $s$, the estimation of unknown original signal $w$ can be obtained by the
estimation of $\theta$.

### 3.3.1 MAP Estimation of $s$

Using the Bayes’ rule, we can rewrite $p(s|y)$ as

$$p(s|y) = \frac{p(s)p(y|s)}{\sum_{s'} p(s')p(y|s')}, \quad (3.6)$$

where the summation is over all the possible $s$ vectors describing the support of $w$. Note that the denominator in (3.6) is common to all posterior likelihoods, $p(s|y)$, and thus can be ignored as it is a normalizing constant. To evaluate $p(s)$, we know that the $s$ vector is a stationary first-order Markov process with two transition probabilities given in Section 3.2. Therefore, $p(s)$ is given by

$$p(s) = p(s_1) \prod_{i=1}^{M-1} p(s_{i+1}|s_i), \quad (3.7)$$

where $p(s_1) = p(1-s_1)(1-p)^{s_1}$ and

$$p(s_{i+1}|s_i) = \begin{cases} 
(1 - p_{10})(1-s_{i+1})p_{10}^{s_{i+1}} & \text{if } s_i = 0, \\
 p_{01}^{(1-s_{i+1})}(1 - p_{01})^{s_{i+1}} & \text{if } s_i = 1.
\end{cases} \quad (3.8)$$

It remains to calculate $p(y|s)$. As $w|s$ is Gaussian, $y$ is also Gaussian with zero mean and the covariance

$$\Sigma_s = E[yy^T|s] = \sigma_n^2 I_N + \sigma_\theta^2 \Phi S \Phi^T, \quad (3.9)$$

where $S = \text{diag}(s)$ as defined in (3.2). Therefore, up to an inessential multiplicative
constant factor \((\frac{1}{\pi^N})\), we can write the likelihood function as

\[
p(y|s) = \exp \left( -\frac{1}{2} y^T \Sigma_s^{-1} y \right) \frac{1}{\det(\Sigma_s)}, \tag{3.10}
\]

Hence, the MAP estimate of \(s\) is given by

\[
s_{\text{MAP}} = \arg\max_s p(s)p(y|s), \tag{3.11}
\]

where \(p(s)\) is calculated using (3.7) and (3.8), whereas the prior likelihood \(p(y|s)\) is given by (3.10). The maximization is performed over all \(2^M\) possible sets of \(s\) vectors.

### 3.3.2 MAP Estimation of \(\theta\) using Gamma Prior

After the binary vector \(s\) is estimated, we complete the estimation of the original unknown signal \(w\) by estimating the amplitude samples of the \(\theta\) vector. To this end, we estimate the amplitudes with considering hyperprior over the inverse of the variance. Full details are given below.

Following the Sparse Bayesian Learning (SBL) framework [58], we consider a Gaussian prior distribution for amplitude vector \(\theta\):

\[
p(\theta; \gamma_i) \sim \mathcal{N}(0, \Sigma_0^{-1}) \tag{3.12}
\]

where \(\Sigma_0 = \text{diag}(\gamma_1, \gamma_2, \cdots, \gamma_M)\). Furthermore, \(\gamma_i\) are the non-negative elements of the hyperparameter vector \(\gamma\), that is \(\gamma \triangleq \{\gamma_i\}\). Based on the SBL framework, we use Gamma distributions as hyperpriors over the hyperparameters \(\{\gamma_i\}\):

\[
p(\gamma) = \prod_{i=1}^{M} \text{Gamma}(\gamma_i | a, b) = \prod_{i=1}^{M} \Gamma(a)^{-1} b^a \gamma_i^{a-1} e^{-b\gamma_i}
\]
where \( \Gamma(a) = \int_0^\infty t^{a-1}e^{-t}dt \) is the Gamma function. To obtain non-informative Gamma priors, we assign very small values, e.g. \( 10^{-4} \) to two parameters \( a \) and \( b \).

Note that, as we have already estimated the support vector \( s \) presented in Section 3.3.1, using SBL on top of Bernoulli-Gaussian model of samples \( w_i \) would be a reasonable prior. This is because with the knowledge of the support samples \( s_i \) the SBL accurately estimates the amplitude samples \( \theta_i \) without considering the spatial correlation of the signal samples \( w_i \). Although a Gaussian prior over the amplitude vector \( \theta \) is computationally more efficient, estimation by SBL leads to more accurate results (see simulation results in Section 3.7.1).

From (3.2), we can rewrite the linear model of (2.1) as

\[
y = \Phi S \theta + n = \Psi \theta + n
\]  

where \( \Psi = \Phi S \). Therefore, from the linear model of (3.13) and given the support vector \( s \), the likelihood function also has Gaussian distribution:

\[
p\left( y \mid \theta; \sigma^2_n \right) \sim \mathcal{N}_{y\theta} \left( \Psi \theta, \sigma^2_n I_N \right).
\]  

Using the Bayes’ rule the posterior \( p\left( \theta \mid y; \gamma, \sigma^2_n \right) \) is found as a multivariate Gaussian with its mean and covariance given respectively as

\[
\mu_\theta = \sigma^{-2}_n \Sigma_\theta \Psi^T y,
\]

\[
\Sigma_\theta = \left( \sigma^{-2}_n \Psi^T \Psi + \Sigma_0 \right)^{-1}
\]

\[
= \Sigma_0^{-1} - \Sigma_0^{-1} \Psi^T \left( \sigma^2_n I_N + \Psi \Sigma_0^{-1} \Psi^T \right)^{-1} \Psi \Sigma_0^{-1}.
\]

Therefore, given the hyperparameters \( \gamma_i \) and noise variance \( \sigma^2_n \), the MAP esti-
mate of $\theta$ is

$$
\hat{\theta}_{MAP} = \mu_{\theta} = (\Psi^T\Psi + \sigma_n^2\Sigma_0)^{-1}\Psi^Ty
$$

(3.18)

$$
= \Sigma_0^{-1}\Psi^T(\Psi\Sigma_0^{-1}\Psi^T + \sigma_n^2I_N)^{-1}y,
$$

(3.19)

where (3.19) follows the identity equation $(A + BB^T)^{-1}B = A^{-1}B(I + B^TA^{-1}B)^{-1}$, and $\Sigma_0 = \text{diag}(\gamma_1, \gamma_2, \cdots, \gamma_M)$. Moreover, the hyperparameters $\gamma_i$ control the sparsity of the amplitudes $\theta_i$. Sparsity in the samples of the amplitudes occur when particular variables $\gamma_i \to \infty$, whose effect forces the $i$th sample to be pruned out from the amplitude estimate.\footnote{In practice, we observe that when the estimates $\gamma_i$ become very large, e.g. $10^5$ so that the coefficient of the $i$th sample is numerically indistinguishable from zero, then the associated sample in $\theta$ is set to zero.} To calculate $\hat{\theta}_{MAP}$, we can also use (3.15) directly in which we have two options for obtaining the covariance matrix $\Sigma_{\theta}$ using (3.16) and (3.17). Note that, the computational complexity for estimation of $\Sigma_{\theta}$ is different in (3.16) and (3.17). An $M \times M$ matrix inversion is required using (3.16), whereas an $N \times N$ matrix inversion is needed in (3.17).

When the noise variance ($\sigma_n^2$) is also unknown, we can place conjugate gamma prior on the inverse of the variance (i.e. $\beta \triangleq \sigma_n^{-2}$) as $p(\beta) = \text{Gamma}(\beta | c, d)$, where $c = d = 10^{-4}$. In fact, the complexity of posterior distribution will be alleviated by using conjugate priors. To estimate the hyperparameters, we utilize the Relevance Vector Learning (RVL) which is maximization of the product of the marginal likelihood (Type-II maximum likelihood) and the priors over the hyperparameters $\gamma$ and $\beta$ ($\sigma_n^2$) [58]. Given the priors, the likelihood of the observations can be given as

$$
p(y \mid \gamma, \beta, a, b, c, d) = \mathcal{N}(y \mid 0, \Psi\Sigma_0^{-1}\Psi + \beta^{-1}I_N)
$$

$$
\times p(\gamma) \times p(\beta).
$$

(3.20)
A maximum likelihood (ML) estimator which maximizes (3.20) can be used to find the unknown hyperparameter $\gamma$ and $\beta$. To this end, we use an iterative expectation maximization (EM) to compute the unknown variables. Hence, to compute the ML estimate of the unknown hyperparameters $\gamma$ and $\beta$, we treat $\theta$ as the latent variables and apply the EM algorithm. Moreover, we define $\Theta \triangleq \{\gamma, \beta\}$ for brevity. The EM algorithm proceeds by maximizing the following expression

$$Q(\Theta) = E_{\theta|y, \Theta^{(k)}} \left[ \log(p(y|\theta, \beta) p(\theta|\gamma) p(\gamma)p(\beta)) \right]$$

$$= E_{\theta|y, \Theta^{(k)}} \left[ \log(p(\gamma) p(\theta|\gamma)) \right] + E_{\theta|y, \Theta^{(k)}} \left[ \log(p(\beta) p(y|\theta, \beta)) \right],$$

(3.21)

where $\Theta^{(k)}$ refers to the current estimate of hyperparameters. To estimate $\Theta$, we observe that the first and second summands in (3.21) are independent of each other. Hence, the estimate of $\gamma$ and $\beta$ is separated into two different optimization problems. For obtaining $\gamma$, the following iterative expression can be solved

$$\gamma^{(k+1)} = \arg\max_{\gamma} E_{\theta|y, \Theta^{(k)}} \left[ \log(p(\gamma) p(\theta|\gamma)) \right].$$

(3.22)

Therefore, an update for hyperparameter $\gamma$ by computing the first derivative of the first summand of (3.21) with respect to $\gamma$ can be expressed as

$$\gamma_i^{(k+1)} = \frac{1 + 2a}{\left(\mu_{\theta,i}^{(k)}\right)^2 + \Sigma_{\theta,ii}^{(k)} + 2b},$$

(3.23)

where $\mu_{\theta,i}$ denotes the $i$th entry of $\mu_{\theta}$ in (3.15) and $\Sigma_{\theta,ii}$ denotes the $i$th diagonal element of the covariance matrix $\Sigma_{\theta}$ in (3.16) or (3.17).

Following the same method, we need to solve the following optimization prob-
lem to estimate $\beta$:

$$
\beta^{(k+1)} = \arg\max_{\beta} \mathbb{E}_{\theta | y, \Theta^{(k)}} [\log(p(\beta) p(y | \theta, \beta))].
$$

(3.24)

Hence, the $\beta$ learning rule is calculated by setting the first derivative of the second summand in (3.21) with respect to $\beta$ to zero, resulting in

$$
\frac{1}{\beta^{(k+1)}} = \frac{1}{N + 2c} \left\{ \left\| y - \Psi \mu_{\theta}^{(k)} \right\|^2_2 
+ \left( \beta^{(k)} \right)^{-1} \sum_{i=1}^{M} \left[ 1 - \gamma_i^{(k)} \Sigma_{\theta,ii}^{(k)} \right] + 2d \right\}.
$$

(3.25)

Having estimated the posterior probability of $s$ and MAP estimate of amplitude vector $\theta$, the estimation of unknown original signal $w$ is complete. However, the evaluation of (3.11) over all $2^M$ possible sets of $s$ vectors is a computationally daunting task when $M$ is large. The difficulty of this exhaustive search is obvious from (3.6)-(3.11). Hence, in the following section, we propose an Iterative Bayesian Algorithm, referred to as Block-IBA, which reduces the complexity of the exhaustive search.

### 3.4 Block Iterative Bayesian Algorithm

#### 3.4.1 Main Idea

Finding the solution for (3.11) through combinatorial search is computationally intensive. This is because the computation should be done over the discrete space. One way around this exhaustive search is to convert the maximization problem into a continuous form. Therefore, in this section we propose a method to convert the problem into a continuous maximization problem and apply a steepest-ascent algorithm to find the maximum value. To this end, we model the elements of $s$ vector as
a Gaussian Mixture (GM) with two Gaussian variables centered around 0 and 1 with sufficiently small variances. Hence, each discrete element of \( s \) vector, i.e. \( s_i \) can be given as

\[
p(s_i) \approx p \mathcal{N}(0, \sigma_0^2) + (1 - p) \mathcal{N}(1, \sigma_0^2). \tag{3.26}
\]

Moreover, the other elements of \( s \) vector, i.e. \( s_{i+1} \) \((i = 1, \cdots, M-1)\) can be expressed as

\[
p(s_{i+1}) \approx \begin{cases} 
(1 - p_{10}) \mathcal{N}(0, \sigma_0^2) + p_{10} \mathcal{N}(1, \sigma_0^2) & \text{if } s_i = 0, \\
p_{01} \mathcal{N}(0, \sigma_0^2) + (1 - p_{01}) \mathcal{N}(1, \sigma_0^2) & \text{if } s_i = 1.
\end{cases} \tag{3.27}
\]

In order to find the global maximum of (3.11) we decrease the variance \( \sigma_0^2 \) in each iteration of the algorithm gradually, which averts the local maximum of (3.11) (This is similar to simulated annealing algorithms and graduated non-convexity [188]). Although we have converted the discrete variables \( s_i \) to the continuous form, finding the optimal value of \( s \) using (3.11) is still complicated. Thus, we propose an algorithm that estimates the unknown original signal \( \mathbf{w} \) by estimating its components (\( s \) and \( \theta \) in (3.2)) iteratively. We follow a two-step approach to estimate the \( \mathbf{w} \) vector. In the first step, we estimate the amplitude vector \( \theta \) (i.e., \( \hat{\theta} \)) based on the known estimation of \( s \) (i.e., \( \hat{s} \)) vector and the mixing observation vector \( \mathbf{y} \). We call this expectation step (E-step).

Having assumed the Gamma distribution as hyperpriors over the hyperparameters \( \{\gamma_i\} \) as explained in Section 3.3.2, the following equation similar to (3.18) and (3.19) can be derived as

\[
\hat{\theta} = \left( \hat{\Psi}^T \hat{\Psi} + \sigma_n^2 \Sigma_0 \right)^{-1} \hat{\Psi}^T \mathbf{y} \tag{3.28}
\]

\[
= \Sigma_0^{-1} \hat{\Psi}^T \left( \hat{\Psi} \Sigma_0^{-1} \hat{\Psi}^T + \sigma_n^2 I_N \right)^{-1} \mathbf{y}, \tag{3.29}
\]
where $\hat{\Psi} = \Phi \hat{S}$.

We call the second step of our approach maximization step (M-step). In this step, we find the estimate of $s$ with the assumption of known vector $\hat{\theta}$ and the observation vector $y$. Therefore, we can write the MAP estimate of $s$ as

$$\hat{s}_{\text{MAP}} = \arg\max_s p(s \mid y, \hat{\theta}) \equiv \arg\max_s p(s \mid \hat{\theta}) p(y \mid s, \hat{\theta}) \equiv \arg\max_s p(s) p(y \mid s, \hat{\theta}) \equiv \arg\max_s \log(p(s)) + \log(p(y \mid s, \hat{\theta})).$$

Using (3.26) and (3.27), we can express $p(s)$ as a continuous variable

$$p(s) = p(s_1) \prod_{i=1}^{M-1} p(s_{i+1} \mid s_i)$$

$$= p \exp \left( -\frac{s_1^2}{2\sigma_0^2} \right) + (1 - p) \exp \left( -\frac{(s_1 - 1)^2}{2\sigma_0^2} \right) \times \prod_{i=1}^{M-1} \left\{ [p_{01} + (1 - p_{10})] \exp \left( -\frac{s_{i+1}^2}{2\sigma_0^2} \right) + [p_{10} + (1 - p_{01})] \exp \left( -\frac{(s_{i+1} - 1)^2}{2\sigma_0^2} \right) \right\}.$$ \hfill (3.31)

It remains to calculate $p(y \mid s, \hat{\theta})$ in (3.30). From (3.13), we can write

$$p(y \mid s, \hat{\theta}) = \frac{1}{\left( \sqrt{2\pi} \sigma_n^2 \right)^M} \exp \left( -\frac{\|y - \Psi \hat{\theta}\|_2^2}{2\sigma_n^2} \right),$$

where $\Psi = \Phi S$. After calculating the two summands in (3.30), we can express the M-step as

$$\hat{s} = \arg\max_s L(s),$$

\hfill (3.33)
where
\[
\mathcal{L}(s) = \log (p(s_1)) + \sum_{i=1}^{M-1} \log(p(s_{i+1} | s_i)) - \frac{1}{2\sigma^2_n} \left\| y - \Psi \hat{\theta} \right\|^2_2.
\] (3.34)

We can find the optimal solution of (3.33) by performing the steepest-ascent method. The expression for obtaining the sequence of optimal solutions in this method can be given as
\[
s^{(k+1)} = s^{(k)} + \mu \frac{\partial \mathcal{L}(s)}{\partial s} \bigg|_{s=s^{(k)}},
\] (3.35)
where \(\mu\) is the step-size of the steepest-ascent method. The gradient term in (3.35) can be expressed in a closed form (see Appendix A). Therefore (3.35) can be rewritten as
\[
s^{(k+1)} = s^{(k)} + \frac{\mu}{\sigma_0^2} g(s) + \frac{\mu}{\sigma_n^2} \text{diag}(\Phi^T(\Psi \hat{\theta} - y)) \cdot \hat{\theta},
\] (3.36)
where \(g(s) = g_1(s) + g_2(s)\) as derived in Appendix A. Moreover, the two scalar functions \(g_1(s_1)\) and \(g_2(s_{i+1})\), \(i = 1, 2, \ldots, M - 1\), can be given as
\[
g_1(s_1) = \frac{ps_1 \exp(\frac{-s_1^2}{2\sigma_0^2}) + (1-p)(s_1 - 1) \exp(\frac{-(s_1-1)^2}{2\sigma_0^2})}{p \exp(\frac{-s_1^2}{2\sigma_0^2}) + (1-p) \exp(\frac{-(s_1-1)^2}{2\sigma_0^2})},
\] (3.37)
\[
g_2(s_{i+1}) = \frac{q_1 s_{i+1} \exp(\frac{-s_{i+1}^2}{2\sigma_0^2}) + q_2 (s_{i+1} - 1) \exp(\frac{-(s_{i+1}-1)^2}{2\sigma_0^2})}{q_1 \exp(\frac{-s_{i+1}^2}{2\sigma_0^2}) + q_2 \exp(\frac{-(s_{i+1}-1)^2}{2\sigma_0^2})},
\] (3.38)
where \(q_1 = p_{01} + (1 - p_{10})\) and \(q_2 = p_{10} + (1 - p_{01})\). Note that in the computation we decrease \(\sigma_0\) in the consecutive iterations (i.e., \(\sigma_0^{(k+1)} = \alpha \sigma_0^{(k)}\), \(0.6 < \alpha < 1\)) to guarantee the global maxima of (3.34). As the step size \(\mu\) has a great effect on the convergence of the Block-IBA, its proper range is analytically determined in Section 3.6 to guarantee the convergence of Block-IBA (with a probability close to one). If the columns of \(\Phi\) are normalized to have unit \(\ell_2\)-norms, the range for step size \(\mu\) can
be expressed as

$$0 < \mu < \frac{2}{\sigma_0^2 + \frac{MM^2}{\sigma_n^2}}, \quad (3.39)$$

where $M^* = \sigma_0 Q^{-1}(1 - \frac{N(0.99)}{2})$ and $Q^{-1}(\cdot)$ is the inverse Gaussian Q-function.

We initialize the proposed Block-IBA with the minimum $\ell_2$-norm solution and use a diminishing threshold $Th^{(k+1)} = \alpha Th^{(k)}$, $0.6 < \alpha < 1$ for determining the active columns of the measurement matrix $\Phi$, which effectively selects the nonzero elements of the signal $w$. In fact, the value of $Th$ determines the number of nonzero elements in $s$ vector.

Finally, it is observed from (3.36) that the second summand controls the block sparsity of the $w$ vector, whereas the third summand controls the noise power, i.e. $||y - \Phi w||_2^2$. For instance, when the value of $\sigma_n$ is much smaller than the value of $\sigma_0$, the third summand dominates the second summand and the block sparsity of $w$ is ignored while the optimal solution satisfies $y = \Phi w$. However, to obtain a meaningful solution in terms of block sparsity and noise power we should select appropriate values for $\sigma_0$ and $\sigma_n$ which are comparable to each other.

### 3.4.2 Discussion

The main difference between Block-IBA and its counterpart in [184] is the E-step. In [184], the authors used a minimum mean-square error (MMSE) estimation to estimate the amplitude vector $\theta$, while here, we utilized the Relevance Vector Learning (RVL) and Gamma prior over the hyperparameters $\gamma$, which are more accurate than MMSE estimation. Unlike the iterative Bayesian Algorithm (IBA) in [184], in M-step we gradually reduce the step-size parameter $\mu$ in each iteration (see Algorithm 3.1) to improve the convergence rate and accuracy. Also, different to IBA in [184], we use a diminishing threshold so that the measurement matrix $\Phi$ effectively selects the
nonzero elements of signal $w$.

The analysis of global maxima in Section 3.6.1 also differs from that in [184]. This is because the cost function $\mathcal{L}(w)$ in this paper is for BGHMM which is completely different from the classic spike-and-slab model used in [184]. Moreover, in Section 3.6.2, we have used the error vector method to show that the steepest-ascent algorithm in (3.35) actually converges to the optimal value (local maxima). This is a stronger stability analysis than that of [184] which only shows the steepest-ascent algorithm will converge. However, as the condition (D.11) of the step-size parameter $\mu$ for the guaranteed convergence of the steepest-ascent algorithm depends on the eigenvalues of the matrices comprising the amplitudes of the signal model, the overall expressions of $\mu$ in [184] and (3.39) are the same.

In this section, we have presented the first and the second steps of the Block-IBA, i.e. E-step and M-step given in (3.29) and (3.36), respectively. In the next section, we complete the Block-IBA by estimating the unknown parameters of the signal model in Section 3.2.

### 3.5 Learning The Signal Model Parameters

The signal model presented in Section 3.2 is characterized by Markov chain parameters $p$ and $p_{01}$, and the AWGN variance $\sigma_n^2$. It is likely that some or all of these parameters will require tuning to obtain better estimate of the unknown original signal. For this purpose, we develop some estimation algorithms which work together with Block-IBA in Section 5.4 to learn all of the model parameters iteratively from the available data.

In Section 3.3.2, we have derived an update in (3.25) for $\beta \overset{\Delta}{=} \sigma_n^{-2}$ when we assumed the Gamma distribution as hyperprior over the hyperparameter $\gamma$ and the Gamma prior for the inverse of the noise variance $\sigma_n^2$. Moreover, using the MAP
estimation method we can express the following update equations for the rest of parameters

\[ p^{(k+1)} = \frac{\|s\|_0}{M}, \quad (3.40) \]

\[ p_{01}^{(k+1)} = \frac{\sum_{i=1}^{M-1} s_i (1 - s_{i+1})}{\sum_{i=1}^{M-1} s_i}. \quad (3.41) \]

A complete derivation of the update in (3.40) can be found in [184], while the derivation of (3.41) is presented in Appendix B. Before starting the estimate of unknown parameters in each iteration, it is essential to first initialize the parameters at reasonable values. For instance, the initial value for \( p^{(0)} \) is an arbitrary value between 0.5 and 1. Furthermore, the initial value for \( \sigma_n \) can be given as \( \sigma_n^{(0)} = \sqrt{\frac{\sum_{i=1}^{N} y_i^2 - (\frac{\sum_{i=1}^{N} y_i}{N})^2}{N}} \).

Algorithm 3.1 provides a pseudo-code implementation of our proposed Block-IBA that gives all steps in the algorithm including E-step, M-step, and Learning Parameter-step. The \( Th^{(k)} \) in the algorithm is a diminishing threshold, calculated at each iteration \( k \) by \( Th^{(k+1)} = \alpha Th^{(k)}, 0.6 < \alpha < 1 \). Its value determines the number of nonzero elements in \( s \) vector. By gradually decreasing the threshold in the algorithm, the nonzero elements of \( s \) vector are picked out. In particular, line 12 of Algorithm 3.1 describes the procedure to accurately choose (based on the threshold \( Th^{(k)} \)) the nonzero elements of the support vector \( s \).

By numerical study, we empirically find that the initial value of the threshold should be \( Th^{(0)} = 0.5 \) to achieve reasonable performance. This is because the value of \( Th^{(0)} \) specifies the number of nonzero elements in \( s^{(0)} \) vector. We will elaborate on this parameter in Section 3.7.

As shown below, the matrices \( \Sigma_0^{(k)} \) and \( (\sigma_n^{(k)})^2 I_N + \Psi^{(k)}\Sigma_0^{(k)-1}\Psi^{(k)T} \) in Step 4 of Algorithm 3.1 are always invertible, and hence the Algorithm 3.1 works well
Algorithm 3.1 The overall Block-IBA estimation.

Input: $y$, $\Phi$, $k_{max}$, $r_{max}$, $\alpha (0.6 < \alpha < 1)$, $\mu$, and $\epsilon$ ($\epsilon < 1$)

Initialize: Choose $p(0) \in [0, 1]$, $\sigma_{n}^{(0)} = \sqrt{\frac{\sum_{i=1}^{N}y_{i}^2 - (\sum_{i=1}^{N}y_{i})/N}{\frac{p(0)}{N} + \frac{(1-p(0))}{N}}}$, $T_h(0) = 0.5$, $\theta^{(0)} = \Phi^T(\Phi\Phi^T)^{-1}y$, $s^{(0)} = (\theta(0) > T_h(0))$, $w^{(0)} = s^{(0)} \odot \theta^{(0)}$, set difference = 1, $k = 0$.

1: while (difference $> \epsilon$ and $k \leq k_{max}$) do
2: E-step: $S^{(k)} = \text{diag}(s^{(k)})$, $\Psi^{(k)} = \Phi S^{(k)}$.
3: $\theta^{(k)} = \sigma_n^{(k)}(\sigma_n^{(k)} + 1)\Psi^{(k)}^{T}y$.
4: $\Sigma_{\theta}^{(k)} = \Sigma_{\theta}^{(k)} - \Psi^{(k)}^{T}(\sigma_n^{(k)}(\sigma_n^{(k)} + 1)\Psi^{(k)})(\sigma_n^{(k)}(\sigma_n^{(k)} + 1)\Psi^{(k)})^{-1}$
   \hspace*{1cm} $\times \Psi^{(k)}\Sigma_{\theta}^{(k)}$. \hspace*{1cm}
5: M-step:
6: for $r = 0, \cdots, r_{max} - 1$ do
7: $s^{(r+1)} = s^{(r)} + \frac{\mu(r)}{\sigma_{0}^{(r)} \sigma_{0}^{(r)} g(s^{(r)}) + \frac{\mu(r)}{\sigma_{n}^{(r)}} \text{diag}(\Phi^T(\Psi^{(k)}\theta^{(k)} - y)) \cdot \theta^{(k)}$.
8: $\sigma_{0}^{(r+1)} = \alpha \sigma_{0}^{(r)}$, $\mu^{(r+1)} = \alpha \mu^{(r)}$.
9: $S^{(r)} = \text{diag}(s^{(r)})$, $\Psi^{(r)} = \Phi S^{(r)}$.
10: end for
11: Decreasing Threshold: $T_h^{(k+1)} = \alpha T_h^{(k)}$, $0.6 < \alpha < 1$.
12: Updating supports: $s^{(k)} = (\theta^{(k)} > T_h^{(k)})$.
13: Parameter Estimation: $\gamma_i^{(k+1)} = \frac{1+2a}{\gamma_i^{(k)} + 2 + 2b}$, for $i = 1, 2, \cdots, M$.
14: $\Sigma_{0}^{(k+1)} = \text{diag}(\gamma_1^{(k+1)}, \gamma_2^{(k+1)}, \cdots, \gamma_M^{(k+1)})$.
15: $\sigma_n^{(k+1)} = \frac{\mu^{(k+1)}}{\gamma_i^{(k+1)}}$ using (3.25).
16: $p^{(k+1)} = \frac{\|s^{(k)}\|_M}{M} \cdot P_{01} = \frac{\sum_{i=1}^{M-1} s_i^{(k)}(1-s_{i+1})}{\sum_{i=1}^{M-1} s_i^{(k)}}$.
17: $w^{(k)} = s^{(k)} \odot \theta^{(k)}$.
18: Compute difference $\Delta \triangleq \frac{||w^{(k+1)} - w^{(k)}||_2}{||w^{(k+1)}||_2}$, $k \leftarrow k + 1$.
19: end while

Output: $\hat{w} = w^{(k)}$
even in the noiseless scenario when $\sigma_n^2$ approaches zero. This is confirmed by the experimental results in Section 3.7.1.

In Algorithm 3.1, we use $\sigma_0^{(k+1)} = \alpha \sigma_0^{(k)}$, $0.6 < \alpha < 1$ and $\sigma_0^{(0)} = 1$ to decrease the variance $(\sigma_0^{(k)})^2$ in each iteration of the M-step to avoid the local maxima of (3.11), and use finite, e.g. $k_{\text{max}} = 5$, iterations for the convergence of $L(s)$ defined in (3.34). Hence, $\sigma_0^{(k)} \geq (0.6)^5 = 0.0778$ for all $k \leq k_{\text{max}} = 5$. Let $s_i^{(\text{inact})}$, $i \notin S$, be an inactive element of $s^{(k)}$ vector. Since all $s_i^{(\text{inact})}$'s follow a continuous Gaussian distribution with zero mean and variance $(\sigma_0^{(k)})^2$, the probability for $|s_i^{(\text{inact})}|$ to be greater than a constant $L_b > 0$ is 
\[ p_b = p(|s_i^{(\text{inact})}| > L_b) = 2Q\left(\frac{L_b}{\sigma_0}\right), \]
where $Q(\cdot)$ is the Gaussian Q-function. As $\sigma_0^{(k)} \geq 0.0778$, we have $p_b \geq 2Q(L_b/0.0778)$. Because $2Q(L_b/0.0778) = 0.999$ when $L_b = 10^{-4}$, it then follows that $p_b = p(|s_i^{(\text{inact})}| > 10^{-4}) \geq 0.999$, for all $i \notin S$. Therefore, the $|s_i^{(\text{inact})}|$'s are very small but nonzero and satisfy $|s_i^{(\text{inact})}| > 10^{-4}$ with a probability $p_b \geq 0.999$. As a result, none of the elements of matrix $\Psi^{(k)} = \Phi S^{(k)}$ is zero. Moreover, none of the diagonal elements of $\Sigma_0^{(k)}$, $\gamma_i^{(k)}$, is zero. This is because $\gamma_i^{(0)}$'s are initialized with 1 and thus for all $k \in [0, k_{\text{max}} = 5]$, the $\Sigma_0^{(k)}$ in Step 4 and $\theta^{(k)}$ in Step 3 are always finite. Notice that the $\gamma_i^{(k)}$ are estimated using the parameter estimation equation in Step 13, where $a$ and $b$ are small constants, e.g. $a = b = 10^{-4}$, and $\mu_{\theta,i}^{(k)}$, the $i$th entry of $\theta^{(k)}$, and $\Sigma_{\theta,ii}^{(k)}$, the $i$th diagonal element of the covariance matrix $\Sigma_{\theta}^{(k)}$, are all finite. The denominator of $\gamma_i^{(k+1)}$ in Step 13 is therefore always finite and $\gamma_i^{(k+1)} \neq 0$. Consequently, $\Sigma_0^{(k)}$ is invertible for all $k \in [0, k_{\text{max}} = 5]$. The $\Psi^{(k)}$ and $\Sigma_0^{(k)}$ analyzed above guarantee the invertibility of $\Sigma_0^{(k)}$ and $(\sigma_n^{(k)} I_N + \Psi^{(k)} \Sigma_0^{(k)-1} \Psi^{(k)^T})$ at each $k$. 
3.6 Analysis of Global Maximum and Local Maxima

To ensure the convergence of Block-IBA, it is essential to examine the global maximum of the cost function $\mathcal{L}(\mathbf{w}) \triangleq \log p(\mathbf{w} | \mathbf{y})$. Furthermore, as the steepest ascent is used in the M-step of Block-IBA, it is necessary to analyze whether there is a global maximum for the cost function (3.34) which guarantees the convergence of the steepest-ascent method. This analysis also reveals the proper interval for the step-size parameter $\mu$. Finally, we show that there exist a unique local maxima for the cost function $\mathcal{L}(\mathbf{w})$ and this local maxima is equal to the global maximum. Consequently, the convergence of the overall Block-IBA is guaranteed.

3.6.1 Analysis of Global Maxima

The cost function $\mathcal{L}(\mathbf{w})$ which is called the log posterior probability function can be expressed as

$$\mathcal{L}(\mathbf{w}) \propto \log p(\mathbf{w}) + \log p(\mathbf{y} | \mathbf{w}).$$  \hspace{1cm} (3.42)

Further manipulation of (3.42) gives

$$\mathcal{L}(\mathbf{w}) = \log p(\mathbf{w}_1) + \sum_{i=1}^{M-1} \log p(\mathbf{w}_{i+1} | \mathbf{w}_i) - \frac{\| \mathbf{y} - \Phi \mathbf{w} \|^2}{2\sigma^2_n}. \hspace{1cm} (3.43)$$

**Lemma 1.** The cost function $\mathcal{L}(\mathbf{w})$ in (3.43) is concave with respect to $\mathbf{w}$.

*Proof.* See Appendix C.

From the concavity of (3.43), it can be concluded that the cost function $\mathcal{L}(\mathbf{w})$ has a unique global maximum.
3.6.2 Analysis of Local Maxima

In this section, we show that there exist a unique local maxima for the cost function \( L(s) \) in (3.34), which in turn asserts that the M-step (steepest-ascent) converges to this maximum point. To this end, we provide the following lemma.

**Lemma 2.** The iterative steepest-ascent process in (3.35) converges to the optimal solution (local maxima) if step-size \( \mu \) is in the interval \( 0 < \mu < \frac{2}{\sigma_0 + \frac{MM^*}{\sigma_n^2}} \), where \( M^* = \sigma_\theta Q^{-1}(\frac{1 - \sqrt{0.99}}{2}) \) and \( Q^{-1}(\cdot) \) is the inverse Gaussian Q-function.

**Proof.** See Appendix D.

3.6.3 Analysis of Global Maximum of Overall Block-IBA

To prove the existence of the global maximum for the proposed Block-IBA, we should prove that the log posterior probability sequence \( L(w^{(k)}) \) in (3.43) is an increasing sequence. Notice that this condition should be true in both E-step and M-step of the Block-IBA algorithm. As the log posterior \( L(w) \) is equivalent to \( L(s) \) throughout the M-step, it is clear that \( L(w) \) is increasing with respect to the sequence \( w^{(k)} \).

Moreover, in E-step, the estimation of \( \theta \) is performed either through (3.28) or (3.29), which is the MAP estimation of amplitude vector \( \theta \). This MAP estimation implies the maximization of the log posterior \( L(w) \). This is because the logarithm function is concave and monotonically increasing. Hence, the increasing characteristic of the log posterior \( L(w) \) is guaranteed in both E-step and M-step in each iteration. As a result, the sequence \( L(w^{(k)}) \) always converges to a local maxima \( w^* \). We have proved in 3.6.1 that the \( L(w) \) is a concave function. Hence, this unique local maximum attained by the MAP estimate of block-sparse sources in Block-IBA is the global maximum.
3.7 Numerical Evaluation

This section presents the experimental results to demonstrate the performance of Block-IBA. All the experiments are conducted for 400 independent simulation runs. In each simulation run the elements of the matrix $\Phi$ are chosen from a uniform distribution in \([-1,1]\) with columns normalized to unit $\ell_2$-norm. The block-sparse sources $w_{\text{gen}}$ are synthetically generated using BGHMM in (3.5) which is based on Markov chain process. Unless otherwise stated, in all experiments $p = 0.9$, $p_{01} = 0.09$, and $\sigma_0 = 1$ which are the parameters of BGHMM. The measurement vector $y$ is constructed by $y = \Phi w_{\text{gen}} + n$ where $n$ is zero-mean AWGN with a variance tuned to a specified value of SNR which is defined as

$$\text{SNR (dB)} \triangleq 20 \log_{10} \left( \frac{\| \Phi w_{\text{gen}} \|_2}{\| n \|_2} \right). \tag{3.44}$$

In addition, we utilize the following Normalized Mean Square Error (NMSE) as a performance metric

$$\text{NMSE} \triangleq \frac{\| \hat{w} - w_{\text{gen}} \|_2^2}{\| w_{\text{gen}} \|_2^2},$$

where $\hat{w}$ is the estimate of the true signal $w_{\text{gen}}$.

In the numerical studies we compare the proposed Block-IBA with the following block-sparse reconstruction algorithms.

- EBSBL-BO and BSBL-EM, which are the two algorithms from the BSBL framework proposed in [4]. In all the simulations, for EBSBL-BO, we set $h = 4$ the block size parameter, $\text{noise\_Flag} = 1$ (suitable for strongly noisy signal, e.g. $\text{SNR} < 20\text{dB}$), as suggested by the authors. For BSBL-EM, we set $h = 4$ and divide the signal into equal block size in which the start of each block is known. Moreover, we set $\text{Learn\_Lambda} = 1$ for noisy cases.
- BM-MAP-OMP, an algorithm proposed in [3] where the sparsity pattern is modeled by Boltzmann machine. Throughout our experiments, we use the default values for $k_0$, the prior belief on the average cardinality of the supports and $L$, the number of nonzero diagonals in the upper triangle part of the interaction matrix, as suggested by the authors.

- CluSS-MCMC, a hierarchical Bayesian model that uses Markov Chain Monte Carlo (MCMC) sampling approach, proposed in [2]. In all the experiments, we use all the default values suggested by the authors.

- PC-SBL, an algorithm from a coupled hierarchical Gaussian framework proposed in [5], [77]. In all our experiments, we set $\beta = 1$ the relevance parameter between neighboring coefficient. Also, we used the 100 maximum number of iterations for the algorithm, as suggested by the authors.

We also compare the expectation maximization (EM)-SBL algorithm proposed in [58] with the proposed Block-IBA algorithm to demonstrate the benefit of exploiting the block-sparse structure. EM-SBL is a Bayesian learning framework which uses only the sparsity information of the signal without considering the block structure.

### 3.7.1 Performance of Block-IBA versus Block size

Evidently, the strategy of selecting the blocks significantly affects the estimation performance. To examine the influence of the block size on the estimation performance of the Block-IBA under the unknown block partition, we use simulation to compare the Block-IBA with all the other algorithms described above in the noisy and noiseless environments. The size of matrix $\Phi$ is $192 \times 512$, SNR = 15dB, and $\sigma_\theta = 1$. To initialize the steepest ascent method in (3.36) not far from the actual maximum, the value of $\sigma_0$ should decrease slowly. Consequently, the value of $\alpha$ should be in $[0.6, 1]$. 


In this experiment, we choose $\alpha = 0.98$ and the initial value of $\sigma_0 = 1$. For these settings, the suitable interval of $\mu$ in (3.39) is $0 < \mu < 2.1434 \times 10^{-6}$. Hence, we select $\mu = 10^{-6}$ and $Th^{(0)} = 0.5$. Extensive experimental studies demonstrate that for these parameters the $\mathcal{L}(s)$ converges to its maximum value within 4 or 5 iterations, where the stopping criterion is $\frac{|\hat{s}^{(k+1)} - \hat{s}^{(k)}|_2}{||\hat{s}^{(k+1)}||_2} < 0.001$ and $\hat{s}$ is the estimate of the true support $s$. Thus, 5 iterations are used for M-step. Also, we stop the overall Block-IBA when the convergence criterion $\frac{|\hat{w}^{(k+1)} - \hat{w}^{(k)}|_2}{||\hat{w}^{(k+1)}||_2} < 0.001$ is satisfied, where $\hat{w}$ is the estimate of the true signal $w_{gen}$ and $k$ is the iteration number.

Recall from Section 3.2 that the block size and the number of blocks of $w$ are proportional to $1/p_{01}$. When $p_{01}$ is small $w$ comprises small number of blocks with big sizes and vice versa. Hence, we vary the value of $p_{01}$ between 0.09 and 0.9 to obtain the NMSE for various algorithms. The results of NMSE versus $p_{01}$ is shown in Figure 3.1(a) for noisy case. We have also obtained the results of MMSE estimation which is almost a flat line of NMSE around $3.5 \times 10^{-1}$ and have excluded them from Figure 3.1(a) due to the poor performance. As seen from the figure, for $p_{01} \geq 0.36$ the Block-IBA outperforms all other algorithms. As observed from Figure 3.1(a), the proposed Block-IBA significantly outperforms the EM-SBL algorithm [58], which demonstrates the advantage of exploiting the block-sparse structure. We have also obtained the performance of Block-IBA in noiseless scenario. In this case, we have used success rate as the performance metric. The success rate is calculated as the ratio of the number of successful trials to the total number of independent runs. A trial is considered successful if $\text{NMSE} \leq 10^{-5}$. The results of the success rate versus $p_{01}$ are shown in Figure 3.1(b) for the same parameters. The value of $p_{10}$ varies between 0.01 and 0.1. We see that the proposed Block-IBA achieves the highest success rate among all algorithms for $p_{01} \geq 0.27$. Also, for small value of $p_{01}$, e.g. $p_{01} = 0.09$, the signal comprises a few number of blocks with a large number of samples and hence
most of algorithms are unable to provide 100% of success rate.

As \( p_{01} \) increases towards 0.9, the number of samples in a block decreases towards 1, and the blocks gradually reduce to almost unclustered samples of sparse signal without block structure. This trend is illustrated in Figure 3.2 with \( p_{01} = 0.09 \), \( p_{01} = 0.45 \) and \( p_{01} = 0.9 \). However, Block-IBA still outperforms other algorithms. Hence, Block-IBA also performs well for the sparse signals without block structure.

3.7.2 Performance of Block-IBA versus algorithm parameters

The performance of the Block-IBA is affected by the parameters \( \alpha \) and \( Th^{(0)} \) (see Figure 3.3 and Figure 3.4). In this experiment, we use some simulations to examine the effects of these two parameters on the performance of the Block-IBA.

3.7.2.1 Performance versus Parameter \( \alpha \)

As discussed in Section 5.4, the parameter \( \alpha \) controls the decay rate of \( \sigma_0 (\sigma_0^{(k+1)} = \alpha \sigma_0^{(k)}, 0.6 < \alpha < 1) \) to avert the local maxima of (3.11). To investigate its influence on the performance, we set up simulations to observe the NMSE versus parameter \( \alpha \) for different values of SNR(dB) defined in (5.59) and the average number of active sources \( k = \mathbb{E} [|S|] = M(1 - p) \). Note that \( k \) also specifies the sparsity level of active sources. The results are illustrated in Figure 3.3, where Figure 3.3(a) and Figure 3.3(b) represent the NMSE versus parameter \( \alpha \) for different values of SNR and \( k \), respectively.

As seen from Figure 3.3(a), the suitable range for the value of \( \alpha \) is \([0.9, 1]\). In the experiment of Figure 3.3(a), we set \( \mu = 10^{-6} \), \( k = 50 \), \( M = 512 \) and the average number of nonzero blocks to 5 (i.e, \( p_{01} = 0.1 \)). Although the performance of the Block-IBA increases slightly when \( \alpha \) is too close to one (e.g. \( \alpha > 0.9 \)), it is observed that the performance shows little dependency on this parameter. Extensive
Figure 3.1: Performance of all algorithms vs $p_{01}$ for $N = 192, M = 512$. (a) NMSE versus $p_{01}$ in noisy scenario, SNR = 15dB. (b) Success rate versus $p_{01}$ in noiseless scenario. Results are averaged over 400 simulations.
simulation studies show that $\alpha = 0.98$ is an appropriate choice for block-sparse signal reconstruction. In Figure 3.3(b), the results of NMSE versus parameter $\alpha$ for different values of sparsity levels ($30 \leq k \leq 80$) are illustrated. It can be seen that the Block-IBA still shows a low dependency on parameter $\alpha$ when sparsity level $k$ changes. In addition, the appropriate choice for $\alpha$ is in the range $[0.9,1)$. Hence we chose $\alpha = 0.98$.

3.7.2.2 Performance versus threshold ($Th^{(0)}$)

In this experiment, we investigate the influence of the initial value of the threshold, $Th^{(0)}$, on the performance of the Block-IBA. We vary the value of $Th^{(0)}$ between 0 and 1 to obtain the NMSE versus $Th^{(0)}$ at different values of SNR(dB). The results are shown in Figure 3.4. Although the performance of the Block-IBA demonstrates a low dependency on $Th^{(0)}$, extensive simulation studies show that the optimal choice of $Th^{(0)}$ is $Th^{(0)} = 0.5$. 

Figure 3.2: Support vector $s$ samples for $p_{01} = 0.09$, $p_{01} = 0.45$, and $p_{01} = 0.9$. 
Figure 3.3: Performance of the Block-IBA versus parameter $\alpha$ for $N = 192$, $M = 512$ and $\sigma_\theta = 1$. In (a), the number of active sources, $k$, is fixed to 50 and the effect of SNR is investigated. In (b), SNR is fixed to 15 dB and the effect of sparsity factor is assessed. Values of $k$ are 30, 60, 80. Results are averaged over 400 simulations.
3.7.3 Effect of Sparsity Level on the Performance

Sparsity $|S|$ of the underlying signal $w$ is one of the key elements that has a considerable effect on any Compressed Sensing (CS) algorithm. When $w$ is measured by the $N \times M$ measurement matrix $\Phi$ with its elements randomly chosen from a uniform distribution and with columns normalized to unit $\ell_2$-norm, $|S| < N/2$ is the theoretical upper bound limit that guarantees the uniqueness of the sparsest solution [189], [16]. However, most of the algorithms hardly achieve this limit in practice [189]. Hence, we can gain a lot of insight into an algorithm by manipulating this element and investigating the upcoming changes in the performance. To this end, in this experiment, we study the performance of the Block-IBA in terms of normalized sparsity ratio, $\eta \triangleq \frac{E[|S|]}{N}$. For this experiment, the parameters of the signal model are set at $N = 96$, $M = 256$, $p_{01} = 0.45$, $\sigma_0^2 = 1$, and $\text{SNR} = 15\text{dB}$. The value of $p$ is set based on the specific value of $\eta$ and $p_{10}$ is set so that the expected number of active sources remains constant.
Figure 3.5 illustrates the resulting NMSE versus normalized sparsity ratio ($\eta$) for various algorithms. The results are averaged over 400 trials. It is observed that, the proposed Block-IBA presents the best performance among all the algorithms compared, for $\eta \leq 0.35$. For low sparsity level (e.g. $\eta \geq 0.4$) only PC-SBL outperforms Block-IBA.

3.7.4 Effect of Signal to Noise Ratio (SNR) on the Performance

This subsection compares the performance of Block-IBA with that of the other algorithms at different noise levels. In this experiment, $N = 192$, $M = 512$, $p_{01} = 0.45$, and $\sigma_\theta^2 = 1$. We add the AWGN noise $n$ so that the SNR, defined in (5.59), varies between 5 dB and 25 dB for each generated signal.

The results of NMSE versus SNR (dB) for all algorithms are illustrated in Figure 3.6. It is seen that, the Block-IBA outperforms the other algorithms for SNR $\geq 10$. 
Figure 3.6: NMSE vs. SNR (dB) for different algorithms. Simulation parameters are $M = 512$, $N = 192$, $\sigma_\theta = 1$, and $p_{01} = 0.45$. Results are averaged over 400 runs.

3.7.5 Real-World Data Experiment

We have shown the effectiveness of the Block-IBA for recovering the synthetic data thus far. In this subsection, we evaluate the performance of Block-IBA for recovering an MRI image [190], [191]. Images usually demonstrate the block-sparsity structures, particularly on over-complete basis such as wavelet or discrete cosine transform (DCT) basis. The coefficients of the image in the wavelet or DCT domain tend to appear in clustered structures. Hence, images are appropriate datasets for testing the performance of block-sparse signal reconstruction algorithms. In this experiment, we consider an MRI image $I$ of brain with the dimension of $256 \times 256$ pixels. To simulate MRI data acquisition process, the measurement matrix $\Phi$ is obtained by the linear operation $\Phi = F_1 W^T$. The first operation $W^T$ is the 2-D and 2-level Daubechies-4 discrete wavelet transform (DWT) matrix. The second operation, $F_1$, is a 2-D partial discrete Fourier transform (DFT) matrix. In the experiment, we first randomly extract 216 rows from the 256 rows in the spatial frequency of the image $I$. Therefore, the partial DFT matrix $F_1$ is a $216 \times 256$ compressed sensing matrix consisting
of the randomly selected 216 rows of the $256 \times 256$ DFT matrix. To reduce the computational complexity, we reconstruct the image $I$ column by column. We compare Block-IBA with the other algorithms described in this section using the same parameter setups. The performance of various algorithms is summarized in Table 3.1.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>NMSE (dB)</th>
<th>Runtime</th>
</tr>
</thead>
<tbody>
<tr>
<td>EBSBL-BO</td>
<td>-0.0518</td>
<td>27.09 sec</td>
</tr>
<tr>
<td>BM-MAP-OMP</td>
<td>-2.37</td>
<td>1341.6 sec</td>
</tr>
<tr>
<td>CluSS-MCMC</td>
<td>-19.93</td>
<td>1784.4 sec</td>
</tr>
<tr>
<td>PC-SBL</td>
<td>-24.13</td>
<td>54.66 sec</td>
</tr>
<tr>
<td>BSBL-EM</td>
<td>-24.23</td>
<td>354.6 sec</td>
</tr>
<tr>
<td>Block-IBA</td>
<td>-26.78</td>
<td>75.39 sec</td>
</tr>
</tbody>
</table>

3.1. The Block-IBA outperforms all the other algorithms in respect of NMSE. Although EBSBL-BO algorithm appears to be as the fastest algorithm, it shows a very poor performance. Considering Runtime with reasonable performance, only PC-SBL performs faster than Block-IBA. Figure 3.7 compares the original MR image reconstructed by these algorithms and the corresponding error images. The simulations have been performed in MATLAB environment using an Intel 3.10-GHz processor with 8 GB of RAM and under Windows operating system. We have excluded the image reconstructed by EBSBL-BO algorithm because of its severe distortion. We observe that Block-IBA presents the best performance among all the algorithms.

3.8 Conclusion

This chapter has presented a novel Block-IBA to recover the block-sparse signals with completely unknown structure of block sparsity. Unlike the existing algorithms, we have modeled the cluster pattern of the signal using Bernoulli-Gaussian hidden Markov model (BGHMM), which better represents the non-i.i.d. block-sparse sig-
The proposed Block-IBA utilizes diminishing thresholding to effectively select the nonzero elements of signal $w$, and takes advantages of the iterative MAP estimation of sources and the EM algorithm to reduce the complexity of the Bayesian methods. The MAP estimation approach in the Block-IBA renders learning all the signal model parameters automatically from the available data. We have optimized the M-step of the EM algorithm with the steepest-ascent method and provided an analytical solution for the step-size parameter $\mu$ of the steepest-ascent that guarantees the convergence of the overall Block-IBA. We have presented a theoretical analysis to show the global convergence and optimality of the proposed Block-IBA.

Experimental results demonstrate that the Block-IBA has a low dependency on the algorithm parameters and hence is computationally robust. In empirical studies on synthetic data, the Block-IBA outperforms many state-of-the-art algorithms when the block-sparse signal comprises a large number of blocks with short lengths, i.e. non-i.i.d. Numerical experiment on real-world data shows that Block-IBA achieves the best performance among all the algorithms compared at a very low computational cost. Empirical studies also show that the non-i.i.d. block-sparse signals modeled by BGHMM include the sparse signals without block structure as a special case when the model parameter $p_{01}$ approaches 1, and the proposed Block-IBA still outperforms the other state-of-the-art algorithms in such a situation.

The proposed Block-IBA is an estimation-based algorithm, which is a blend of MAP and EM methods. In the next chapter, we develop a novel block Bayesian hypothesis testing algorithm (BBHTA) that uses a joint detection-and-estimation procedure for block-sparse signal recovery.
Figure 3.7: MRI image reconstruction performance for different algorithms accompanied by the corresponding error images.
Chapter 4
Bayesian Hypothesis Testing for Block Sparse Signal Recovery

4.1 Introduction

In Chapter 2, we have introduced the canonical block-sparse model, whose mathematical representation is given by

\[ y = \Phi w + n, \tag{4.1} \]

where \( \Phi \in \mathbb{R}^{N \times M} \) \((N \ll M)\) is a known measurement matrix with columns having unit \( \ell_2 \)-norm, \( y \in \mathbb{R}^N \) is the measurement vector, and \( n \in \mathbb{R}^N \) is the additive white Gaussian noise (AWGN) satisfying \( n \sim \mathcal{N}(0, \sigma_n^2 I_N) \). We aim to estimate the unknown signal \( w \in \mathbb{R}^M \), with the cluster structure

\[ w = \begin{bmatrix} w_1, \ldots, w_{d_1}, \ldots, w_{d_g-1+1}, \ldots, w_{d_g} \end{bmatrix}^T, \tag{4.2} \]

where \( w[i] \) denotes the \( i \)th block with length \( d_i \) which are not necessarily identical. In the block partition (4.2), only \( k \ll g \) vectors \( w[i] \) have nonzero Euclidean norm. Exploiting the block structure of the block-sparse model can achieve better performance than using basic single measurement vector (SMV) model in (2.1) \([65],[68],[70]\).

Most existing sparse- and block-sparse signal recovery algorithms utilize the signal estimation rather than support detection. However, recent research studies show that the existing estimation-based algorithms, e.g. Lasso \([20]\), leaves a potentially large gap between their best performances and the theoretical limit for the
noisy support recovery [78]–[80]. Hence, it is important to develop algorithms that use joint detection-and-estimation procedure [81], [82] to provide a robust support recovery against noise for block-sparse signals.

In this chapter, we propose a novel block Bayesian hypothesis testing (BBHTA) for structure-agnostic block-sparse signal recovery. BBHTA jointly detects the supports by Bayesian hypothesis testing (BHT) [192] and estimates the amplitudes by linear minimum mean-square error estimation (LMMSEE). BHT was first proposed in [7] for a Bayesian pursuit algorithm (BPA) in sparse representations. Unlike BPA that uses the i.i.d. Bernoulli-Gaussian (BG) model to represent the sparse signal, BBHTA uses a more practical Bernoulli-Gaussian hidden Markov model (BGHMM) [1] to describe the block-sparse signal. Recently, an algorithm using BHT with belief propagation, called BHT-BP, has been introduced in noisy sparse recovery [57]. This algorithm also uses i.i.d. BG model to represent the sparse signal. BHT-BP utilizes the nonparametric belief propagation (n-BP) for the detection of the supports. This algorithm uses the low-density parity-check codes (LDPC)-like measurement matrix whose compressing capability performance is worse than that of the dense matrices. However, LDPC-like matrices are capable of fast generation of CS measurements. Although the proposed Block-IBA in Chapter 3 has used BGHMM to describe the block sparse signal, its signal recovery is based on MAP estimation and iterative Expectation Maximization, which is different from the BHT used in BBHTA and requires more computation.

Inspired by BPA [7], we adopt a BHT-based approach and extend BPA to BBHTA. BPA uses the correlation between the measurement vector $y$ and the columns of matrix $\Phi$, in (4.1), in advance to obtain the activity rules. In contrast, BBHTA searches for the start and termination of the blocks of the supports in the block-sparse signal $w$ without using the correlation. This search, performed by BHT, leads to two
ultimate activity rules which manifest the correlations between the measurement vector $y$ and the columns of matrix $\Phi$. In these two activity rules, the correlations are compared with two simple thresholds to detect and recover the supports. Given the detected and recovered supports, BBHTA then uses LMMSEE to estimate the nonzero amplitudes. Such a simple mechanism allows BBHTA to reliably recover the block-sparse signals.

The proposed BBHTA is a double-looped and turbo-like approach (see Algorithm 4.1). The inner loop is a serial procedure that detects the supports. The outer loop is similar to turbo iterative algorithms, estimating the amplitudes of the signal using LMMSEE. The inner loop refines and reuses the LMMSEE of $w$ by combining the block sparsity information in successive iterations. This novel implementation offers more accurate recovery of block-sparse signals.

The rest of the chapter is organized as follows. In Section 4.2, we briefly present the signal model. In Section 4.3, the support detection and amplitude estimation methods of BBHTA are presented. Experimental results are presented in Section 4.4. Finally, conclusions are drawn in Section 4.5.

### 4.2 Signal Model

Consider the block-sparse sources, $w$, in the linear block-sparse model (4.1). Define two hidden random processes, $s$ and $\theta$ [53], [184], where $s \in \{0, 1\}^M$ is a binary vector describing the supports of $w$, denoted as $S$, and $\theta \in \mathbb{R}^M$ is the vector representing the amplitudes of the active elements of $w$. Hence, each element of the source vector $w$ can be characterized as

$$w_i = s_i \cdot \theta_i,$$

(4.3)
where $s_i = 0$ gives $w_i = 0$ for $i \notin S$ and $s_i = 1$ gives $w_i = \theta_i$ for $i \in S$. In vector form, (4.3) can be written as $w = S\theta$, where $S = \text{diag}(s) \in \mathbb{R}^{M \times M}$.

To describe the block-sparsity of the source vector $w$, we assume that its supports $s$ is a stationary first-order Markov process with transition probabilities: $p_{10} \triangleq \Pr\{s_{i+1} = 1|s_i = 0\}$ and $p_{01} \triangleq \Pr\{s_{i+1} = 0|s_i = 1\}$. In the steady state, $\Pr\{s_i = 0\} = p = \frac{p_{01}}{p_{10} + p_{01}}$ and $\Pr\{s_i = 1\} = 1 - p = \frac{p_{10}}{p_{10} + p_{01}}$. Given the two parameters $p$ and $p_{10}$ the remaining transition probability can be determined as $p_{01} = \frac{pp_{10}}{1-p}$. Here, the average number of consecutive samples of ones, i.e. the length of the blocks, is specified by $1/p_{01}$ in Markov chain.

We further assume that the amplitude vector $\theta$ satisfies $\theta \sim N(0, \sigma^2_\theta I_M)$. Hence, the PDF of the $w_i$'s is given as

$$p(w_i) = p\delta(w_i) + (1-p)N(w_i; 0, \sigma^2_\theta),$$  \hspace{1cm} (4.4)

where $\sigma^2_\theta$ is the variance of $\theta$.

Equation (4.4) is Bernoulli-Gaussian hidden Markov model (BGHMM) [1] which is a special form of Gaussian Mixture Hidden Markov model. The hidden variables $s_i$ with the first-order Markov chain model in BGHMM allow implicit expression of the block-sparsity of the signal $w$ to be estimated.

4.3 The Proposed Algorithm

The proposed BBHTA consists of support detection and amplitude estimation. Using BHT, we first detect and recover the Block-sparse support $s$. Then, using a linear MMSE estimator, we estimate the nonzero amplitudes of the detected supports (i.e., estimating $\theta$).
4.3.1 Support Detection Using Bayesian Hypothesis Testing

We determine the activity of the \( j \)th element of the block-sparse signal \( w \) by searching the start and termination of active blocks in \( w \). First, we assume that \( w_i \) is inactive (i.e., \( s_i = 0 \)) and we intend to determine whether \( w_{i+1} \) is active (i.e., \( s_{i+1} = 1 \)). This case is equivalent to searching the start of the active blocks. Next, we assume that \( w_i \) is active (i.e., \( s_i = 1 \)) and we intend to determine whether \( w_{i+1} \) is inactive (i.e., \( s_{i+1} = 0 \)). This corresponds to searching the end of active blocks. Full details are given below.

4.3.1.1 Searching The Start of Active Blocks

To detect the start of an active block we choose one between the hypotheses \( H_{01} : s_i = 0, s_{i+1} = 1 \) and \( H_{00} : s_i = 0, s_{i+1} = 0 \), given the measurement vector \( y \). The Bayesian hypothesis test is

\[
\hat{s}_j = \begin{cases} 
1 & p(H_{01,j} \mid y) > p(H_{00,j} \mid y) \\
0 & \text{Otherwise,}
\end{cases}
\]

where \( y \) is the measurement vector. Using Bayesian hypothesis test, we compute the posteriors \( p(H_{01,j} \mid y) \) and \( p(H_{00,j} \mid y) \). The posterior probability \( p(H_{01,j} \mid y) \) is given as

\[
p(H_{01,j} \mid y) = p(s_is_{i+1} = 01 \mid y) \propto p(s_i = 0) \\
\times p(s_{i+1} = 1 \mid s_i = 0) \times p(y \mid s_is_{i+1} = 01) \\
= p \times p_{10} \times p(y \mid s_is_{i+1} = 01),
\]
where \( y \mid H_{01,j} = \sum_{j=1,j\neq i}^{M} \varphi_j w_j + n \) and \( \varphi_j \) represents the \( j \)th column of matrix \( \Phi \). Similarly, the posterior probability \( p(H_{00,j} \mid y) \) is given by

\[
p(H_{00,j} \mid y) = p(s_is_{i+1} = 00 \mid y) \propto p(s_i = 0) \\
\times p(s_{i+1} = 0 \mid s_i = 0) \times p(y \mid s_is_{i+1} = 00) \\
= p \times p_{00} \times p(y \mid s_is_{i+1} = 00),
\]

(4.7)

where \( p_{00} = p(s_{i+1} = 0 \mid s_i = 0) = 1 - p_{10} \). Hence, from (4.5)–(4.7) the activity rule for \( w_{i+1} \) is

\[
p_{10} \times p(y \mid s_is_{i+1} = 01) > p_{00} \times p(y \mid s_is_{i+1} = 00).
\]

(4.8)

Assume that we have all the estimates of \( w_j \) except for \( j \neq i + 1 \) and we intend to estimate \( w_{i+1} \). We have

\[
p(y \mid s_is_{i+1} = 00, w_{j \neq i,i+1}) = \frac{\exp\left(-\frac{1}{2\sigma_n^2}\|y - \mu_y\|^2\right)}{\sqrt{(2\pi\sigma_n^2)^N}},
\]

(4.9)

where \( \mu_y \triangleq \sum_{j=1,j\neq i,i+1}^{M} \varphi_j w_j \). When \( s_is_{i+1} = 01 \), we have

\[
y \mid H_{01,j} = y \mid H_{00,j} + \varphi_{i+1} w_{i+1} + n = \sum_{j=1,j\neq i,i+1}^{M} \varphi_j w_j + n',
\]

(4.10)

where \( n' = \varphi_{i+1} w_{i+1} + n \). Hence, the likelihood \( p(y \mid s_is_{i+1} = 01) \) is a multivariate Gaussian with the mean \( \mu_y \) and covariance

\[
\Sigma_y = \text{Cov}(n') = \sigma_n^2 I_N + \sigma_\theta^2 \varphi_{i+1} \varphi_{i+1}^T.
\]

(4.11)
Therefore, we can write the likelihood function as

\[ p(y \mid s_is_{i+1} = 01, w_{j \neq i, i+1}) = \frac{\exp\left(-\frac{1}{2}(y - \mu_y)^T \Sigma_y^{-1}(y - \mu_y)\right)}{\sqrt{(2\pi)^N \det(\Sigma_y)}}. \] (4.12)

Using the matrix inversion lemma ([193], p. 571), we can express \( \Sigma_y^{-1} \) as

\[ \Sigma_y^{-1} = \sigma_n^{-2} I_N - \frac{\varphi_{i+1}^T \sigma_n^{-2}}{1 + \left(\frac{\sigma_n}{\sigma_\theta}\right)^2}. \] (4.13)

The determinant of \( \Sigma_y \) can be calculated as

\[ \det(\Sigma_y) = (\sigma_\theta)^{2N} \det\left(\sigma_n^2 I_N + \varphi_{i+1} \varphi_{i+1}^T \sigma_n^{-2}\right) \]
\[ = (\sigma_\theta)^{2N} \left(1 + \varphi_{i+1}^T \frac{\sigma_n^2}{\sigma_\theta^2} \varphi_{i+1}\right) \det\left(\sigma_n^2 I_N\right) \]
\[ = (\sigma_\theta)^{2N} \left(1 + \frac{\sigma_n^2}{\sigma_\theta^2} \varphi_{i+1}^T \varphi_{i+1}\right). \] (4.14)

Using (4.9)-(4.14), the Bayesian hypothesis test in (4.8) can be simplified to give the final activity rule for \( w_{i+1} \) as

\[ \text{Activity}_{\text{START}}(w_{i+1}) \triangleq x^T \varphi_{i+1}^T \varphi_{i+1} x > \text{Th}_{1,i+1}, \] (4.15)

where \( \text{Th}_{1,i+1} \) is defined as

\[ \text{Th}_{1,i+1} \triangleq 2\sigma_n^2 \left(1 + \frac{\sigma_n^2}{\sigma_\theta^2}\right) \ln \left(\frac{p_{10}}{p_{00}} \sqrt{\left(1 + \frac{\sigma_n^2}{\sigma_\theta^2}\right)}\right), \] (4.16)

and \( x = y - \Phi w - \varphi_i w_i - \varphi_{i+1} w_{i+1} \). It is seen that in the activity rule \( \text{Activity}_{\text{START}}(w_{i+1}) \) in (4.15) the correlation between the columns of matrix \( \Phi \) and measurement vector

\[ ^1\text{We have used matrix determinant lemma, i.e. } \det(A + uv^T) = (1 + v^T A^{-1} u) \det(A), \text{ where } A \text{ is an invertible square matrix and } u, v \text{ are column vectors.} \]
x decides $H_{01}$ or $H_{00}$.

### 4.3.1.2 Searching The Termination of Active Blocks

The detection of the end of an active block is performed by choosing one between the hypotheses $H_{10} : s_i = 1, s_{i+1} = 0$ and $H_{11} : s_i = 1, s_{i+1} = 1$, given the measurement vector $y$. The Bayesian hypothesis test is given as

$$
\hat{s}_j = \begin{cases} 
0 & p(H_{10,j} \mid y) > p(H_{11,j} \mid y), \\ 
1 & \text{Otherwise}.
\end{cases}
$$

(4.17)

Using Bayesian hypothesis test, we compute the posteriors $p(H_{10,j} \mid y)$ and $p(H_{11,j} \mid y)$. The posterior probability $p(H_{10,j} \mid y)$ is given as

$$
p(H_{10,j} \mid y) = p(s_is_{i+1} = 10 \mid y) \propto p(s_i = 1) \\
\times p(s_{i+1} = 0 \mid s_i = 1) \times p(y \mid s_is_{i+1} = 10) \\
= (1 - p) \times p_{01} \times p(y \mid s_is_{i+1} = 10),
$$

(4.18)

where $y \mid H_{10,j} = \sum_{j=1,j\neq i+1}^M \varphi_j w_j + n$. Likewise, the posterior probability $p(H_{11,j} \mid y)$ is given by

$$
p(H_{11,j} \mid y) = p(s_is_{i+1} = 11 \mid y) \propto p(s_i = 1) \\
\times p(s_{i+1} = 1 \mid s_i = 1) \times p(y \mid s_is_{i+1} = 11) \\
= (1 - p) \times p_{11} \times p(y \mid s_is_{i+1} = 11),
$$

(4.19)
where \( p_{11} = p(s_{i+1} = 1 \mid s_i = 1) = 1 - p_{01} \) and \( y \mid \mathcal{H}_{11,j} = \sum_{j=1}^{M} \varphi_j w_j + n \). Hence, from (4.17)–(4.19) the inactivity rule for \( w_{i+1} \) is

\[
p_{01} \times p(y \mid s_i s_{i+1} = 10) > p_{11} \times p(y \mid s_i s_{i+1} = 11).
\]

(4.20)

Similar to (4.9), the likelihood function \( p(y \mid s_i s_{i+1} = 10) \) is calculated as

\[
p(y|s_i s_{i+1} = 10, w_{j(j\neq i+1)}) = \exp\left(-\frac{1}{2\sigma_n^2}\|y - \mu'_y\|^2\right)\sqrt{(2\pi\sigma_n^2)^N},
\]

(4.21)

where \( \mu'_y = \sum_{j=1, j\neq i+1}^{M} \varphi_j w_j \). Also, given \( s_i s_{i+1} = 11 \), we have

\[
y \mid \mathcal{H}_{11,j} = y \mid \mathcal{H}_{10,j} + \varphi_{i+1} w_{i+1} + n = \sum_{j=1, j\neq i+1}^{M} \varphi_j w_j + n',
\]

(4.22)

where \( n' = \varphi_{i+1} w_{i+1} + n \). Hence, the likelihood \( p(y \mid s_i s_{i+1} = 11) \) is a multivariate Gaussian with the mean \( \mu'_y \) and its covariance given by (4.11). Therefore, the likelihood function \( p(y \mid s_i s_{i+1} = 11) \) can be evaluated as

\[
p(y \mid s_i s_{i+1} = 11) = \frac{\exp\left(-\frac{1}{2} (y - \mu'_y)^T \Sigma_y^{-1} (y - \mu'_y)\right)}{\sqrt{(2\pi)^N \det(\Sigma_y)}},
\]

(4.23)

where \( \Sigma_y^{-1} \) and \( \det(\Sigma_y) \) are given in (4.13) and (4.14), respectively. Substituting (4.21) and (4.23) into (4.20), the final inactivity rule for \( w_{i+1} \) can be expressed as

\[
\text{Inactivity}_{\text{END}}(w_{i+1}) \triangleq z^T \varphi_{i+1} \varphi_{i+1}^T z > \Theta_{2,i+1},
\]

(4.24)
where $T_{2,i+1}$ is defined as

$$T_{2,i+1} \triangleq 2\sigma_n^2 \left(1 + \frac{\sigma_n^2}{\sigma_\theta^2}\right) \ln \left(\frac{p_{01}}{p_{11}} \sqrt{1 + \frac{\sigma_n^2}{\sigma_\theta^2}}\right), \quad (4.25)$$

and $\mathbf{z} = \mathbf{y} - \Phi \mathbf{w} - \mathbf{v}_{i+1}$. It is seen that in the inactivity rule $\text{Inactivity}_{\text{END}}(w_{i+1})$ in (4.24) the correlation between the columns of matrix $\Phi$ and measurement vector $\mathbf{z}$ decides $\mathcal{H}_{10}$ or $\mathcal{H}_{11}$.

Now consider the unknown parameters $\sigma_\theta$, $\sigma_n$, $p$, $p_{10}$, and $p_{01}$ in (4.16) and (4.25). We can estimate $\sigma_\theta$ by assuming that the elements of $\Phi$ are chosen randomly from a uniform distribution in [-1,1] with columns having unit $\ell_2$-norm. We can estimate the parameters $\sigma_n$ and $p$ by MAP estimation and assuming the other parameters are known. Thus, we can estimate the parameters $\sigma_\theta$, $\sigma_n$, and $p$ with the updates

$$\hat{\sigma}_\theta = \sqrt{\frac{N \mathbb{E}(y_j^2)}{M(1 - \hat{p})}}, \quad \hat{\sigma}_n = \frac{\|\mathbf{y} - \Phi \hat{\mathbf{w}}\|_2}{\sqrt{N}}, \quad \hat{p} = \frac{\|\mathbf{s}\|_0}{M}, \quad (4.26)$$

where $\mathbb{E}(\cdot)$ is the expectation of a random variable. The details about the estimation of $\sigma_n$ and $p$, using MAP estimation method when the other parameters are known, are provided in [184]. A complete derivation of the estimate of $\sigma_\theta$ is presented in Appendix E. To calculate the update equations for parameter $p_{10}$ and $p_{01}$, we also use the MAP estimation approach, assuming the other parameters are known. Hence, the estimates of $p_{10}$ and $p_{01}$ are given by the following simple updates

$$\hat{p}_{10} = \frac{\sum_{i=1}^{M-1} s_{i+1} (1 - s_i)}{\sum_{i=1}^{M-1} (1 - s_i)}, \quad \hat{p}_{01} = \frac{\sum_{i=1}^{M-1} s_i (1 - s_{i+1})}{\sum_{i=1}^{M-1} s_i}. \quad (4.27)$$

The complete derivations of the estimates of $p_{10}$ and $p_{01}$ are presented in Appendix F and Appendix B.
4.3.2 Amplitude Estimation Using LMMSE

Given the detection and recovery information of the binary support vector $s$ by BHT, we complete the estimation of the original unknown signal $w$ by estimating the amplitude samples of the $\theta$ vector. Denote $\hat{s}$ the detected vector $s$. We can obtain the LMMSE estimate $\hat{\theta}$ for $\theta$ [193]

$$\hat{\theta} = \sigma_\theta^2 \hat{S} \Phi^T (\sigma_n^2 I_N + \sigma_\theta^2 \hat{S} \Phi \hat{S}^T)^{-1} y, \quad (4.28)$$

where $\hat{S} = \text{diag}(\hat{s})$.

Algorithm 4.1 provides a pseudo-code implementation of our proposed BBHTA. The inner loop is a consecutive scanning, based on BHT, to detect the supports of the signal in a sequential manner. Similar to turbo iterative algorithms, the outer loop of the algorithm reuses and refines the LMMSE estimate to reconstruct the block-sparse signal $w$ in successive iterations. Also, $k_{\text{max}}$ is a termination condition for the outer loop of the algorithm. Extensive simulation studies show that $k_{\text{max}} = 100$ is sufficient for the overall convergence of the algorithm. Note that $\odot$ in the line 13 of Algorithm 4.1 denotes the Hadamard (element-wise) product.

4.4 Simulation Results

This section presents the experimental results to demonstrate the performance of BBHTA. Two experimental results are presented. We first compare the performance of the proposed BBHTA with that of BPA [7] versus signal to noise ratio (SNR). We then evaluate the performance of BBHTA versus number of nonzero blocks and compare the performance with some block-sparse signal reconstruction algorithms.

All the experiments are conducted for 400 independent simulation runs. In each
Algorithm 4.1 The overall BBHTA estimation.

\begin{algorithm}
\textbf{Input}: $y$, $\Phi$, $k_{\max}$, and $\epsilon$ ($\epsilon < 1$)
\textbf{Initialize}: Choose $p^{(0)} \in [0.5, 1]$, $\sigma^0_\theta = \sqrt{\frac{\text{N_E}(y_j^2)}{\pi(1-p)}}$, $\sigma^0_\sigma = \sigma^0_\sigma / 5$, $w^{(0)} = \Phi^T (\Phi \Phi^T)^{-1} y$. 
\begin{enumerate}
\item while (difference $> \epsilon$ and $k < k_{\max}$) do
\item BHT-detection:
\item for $i = 0, \cdots, M-1$ do
\item if Activity$_{\text{START}}(w_{i+1}) > \text{Th}_{1,i+1}$ in (4.15) then
\item set $s_i = 1$,
\item else if Inactivity$_{\text{END}}(w_{i+1}) > \text{Th}_{2,i+1}$ in (4.24) then
\item set $s_i = 0$,
\item end if
\item end for
\item LMMSE estimation: $S^{(k)} = \text{diag}(s^{(k)})$,
\item $\theta^{(k)} = \sigma^2_\sigma \Phi^T (\sigma^2_\sigma I_N + \sigma^2_\theta \Phi \Phi^T)^{-1} y$.
\item Parameter Estimation: using (4.26) and (4.27)
\item $w^{(k)} = s^{(k)} \odot \theta^{(k)}$.
\item Compute difference $\Delta \left\| \frac{w^{(k+1)} - w^{(k)}}{\left\|w^{(k+1)}\right\|_2} \right\|_2$, $k \leftarrow k + 1$
\item end while
\end{enumerate}
\textbf{Output}: $\hat{w} = w^{(k)}$
\end{algorithm}

Simulation run, the elements of the matrix $\Phi$ are chosen from a uniform distribution in [-1,1] with columns normalized to unit $\ell_2$-norm. The Block-sparse sources $w_{\text{gen}}$ are synthetically generated using BGHMM in (4.4). Unless otherwise stated, in all experiments $p = 0.9$, $p_{01} = 0.09$, and $\sigma_y = 1$ which are the parameters of BGHMM. The measurement vector $y$ is constructed by $y = \Phi w_{\text{gen}} + n$, where $n$ is zero-mean AWGN with a variance tuned to a specified value of SNR which is defined as $\text{SNR}(\text{dB}) \triangleq 20 \log_{10}(\|\Phi w_{\text{gen}}\|_2/\|n\|_2)$. We use the Normalized Mean Square Error (NMSE (dB)) as a performance metric, defined by $\text{NMSE}(\text{dB}) \triangleq 10 \log_{10}(\|\hat{w} - w_{\text{gen}}\|_2/\|w_{\text{gen}}\|_2^2)$, where $\hat{w}$ is the estimate of the true signal $w_{\text{gen}}$. We compare BBHTA and BPA at different noise levels. In this experiment $N = 192$ and $M = 512$. We add additive Gaussian noise so that SNR (dB) varies between 10 dB and 30 dB for each generated signal. Recall from Section 4.2 that decreasing $p$ implies the larger sparsity levels. Hence, in this experiment we evaluate the performance of BPA and BBHTA for $p = 0.9$, $p = 0.8$, and $p = 0.7$.

Figure 4.1 shows the NMSE (dB) versus SNR for both BBHTA and BPA and for different values of $p$. There we observe that BBHTA outperforms BPA for $p = 0.9$,
Figure 4.1: NMSE (dB) versus SNR for BBHTA and BPA with $p = 0.9$, $p = 0.8$, and $p = 0.7$. The results are averaged over 400 trials.

$p = 0.8$, and $p = 0.7$. It is also seen that BBHTA exhibits significant performance gain (almost 5 dB) over BPA for $p = 0.9$.

In the second experiment, we examine the influence of the block size and the number of blocks on the estimation performance of BBHTA where the block partition is unknown. We set up a simulation to compare BBHTA with some recently developed algorithms for block sparse signal reconstruction, such as the block sparse Bayesian learning algorithms (BSBL and EBSBL) [4], the cluster-structured MCMC algorithm (CluSS-MCMC) [2], and the pattern-coupled sparse Bayesian learning algorithm (PC-SBL) [5]. The size of matrix $\Phi$ is $256 \times 512$, SNR = 15dB, and $\sigma_0 = 1$. Recall from Section 4.2 that the block size and the number of blocks of $w$ are proportional to $1/p_{01}$. That is, when $p_{01}$ is small $w$ comprises small number of blocks with big sizes and vice versa. Hence, we vary the value of $p_{01}$ between 0.09 and 0.9 to obtain the NMSE (dB) for various algorithms. The results of NMSE (dB) versus $p_{01}$ is shown in Figure 4.2. As seen from the figure, for $p_{01} \geq 0.36$ BBHTA outperforms all other algorithms.
4.5 Conclusion

This chapter has presented a novel BBHTA to recover the block-sparse signals whose structure of block sparsity is completely unknown. The proposed BBHTA uses a Bayesian hypothesis testing to detect and recover the support of the block sparse signal. For amplitude recovery, BBHTA utilizes an LMMSEE to estimate the nonzero amplitudes of the detected supports. Different to Bayesian pursuit algorithm (BPA) in [7], which models the signal by i.i.d. Bernoulli-Gaussian (BG) model, BBHTA represents the block-sparse signal by a more practical Bernoulli Gaussian hidden Markov model (BGHMM). Unlike Bayesian pursuit algorithm (BPA) in [7], which uses the correlation between measurement $y$ and the columns of matrix $\Phi$ to detect the supports of the signal, BBHTA seeks the start and termination of the blocks of supports in signal $w$ without using the correlation. However, in the activity rules, obtaining by this search, the correlations between measurement $y$ and the columns of matrix $\Phi$ are manifested. These correlations are compared with two simple thresholds to detect the supports. BBHTA is a double-looped and turbo-like approach. The inner
loop is a serial procedure handling the block sparsity information and detection of the supports. The outer loop is similar to turbo iterative algorithm, handling the information related to the amplitudes of the signal using LMMSEE. The function of the inner loop is to refine and reuse the LMMSEE of $w$ by combining the block sparsity information in the successive iterations. This novel implementation of BBHTA results in more accurate recovery of block-sparse signals. Simulation results show that BBHTA outperforms BPA with almost 5 dB performance gain for $p = 0.9$. BBHTA also outperforms many state-of-the-art algorithms when the block-sparse signal comprises a large number of blocks with short lengths.

In many wireless and wireline communication systems, e.g. Power Line Communication (PLC), the additive noise is impulsive and also block-sparse and thus the noise samples are correlated. This block structure can be represented by BGHMM. Since this block-sparse impulsive noise is a major impairment to wireless and PLC networks, it is necessary to design a method to mitigate its effect on communication performance. Therefore, in the next chapter, using Block-IBA presented in Chapter 3, we present a novel and specific receiver for block-sparse impulsive noise cancellation in OFDM-based communication systems.
Chapter 5
Block-Sparse Impulsive Noise Reduction in OFDM Systems - A Novel Iterative Bayesian Approach

5.1 Introduction

A major impairment to Power Line Communication (PLC) and wireless networks is impulsive noise, which is usually generated by various sources such as partial discharges, corona noise, motor ignitions, radio broadcasting, microwave ovens and switching power supplies. The additive white Gaussian noise (AWGN) model is inadequate for representing the characteristics of the impulsive noise and results in suboptimal receivers. Different statistical models have been reviewed in Section 2.5, which better capture the statistical properties of the impulsive noise. Such models facilitate the design and analysis of the performance of receivers in impulsive noise channels.

As discussed in Section 2.6, orthogonal frequency division multiplexing (OFDM) reduces the effect of impulsive noise by spreading the impulses energy over all frequency tones by applying the discrete Fourier transform (DFT) at the receiver [156]. Hence, OFDM receivers are more resilient against impulsive noise than single carrier (SC) receivers. Due to this advantage and other advantages discussed in Section 2.6, OFDM modulation has been widely adopted in many wireless communication standards, including IEEE802.11a and LTE, and narrowband PLC standards, including PRIME and G3-PLC. However, the samples of impulsive noise across subcarriers in frequency domain are no longer statistically independent, and different to AWGN, the independent decoding of subcarriers are suboptimal. More importantly, in many
applications, e.g. PLC and power substations, the impulsive noise in not only highly impulsive with power spectral density (PSD) reaching up to 50 dB higher than background noise [1], but also block-sparse (or bursty) [1]. Hence, the conventional OFDM systems might still suffer from the performance degradation caused by bursty impulsive noise. Hence, it is important to redesign the OFDM receiver to cope with impulsive noise.

As discussed in Section 2.7, the prior work on robust receivers for impulsive noise channels can generally be categorized into two main categories: i) preprocessing mitigation techniques and ii) sparse signal recovery techniques. In preprocessing techniques, the received time-domain signal is preprocessed by clipping or blanking [162]–[165] or nonlinear MMSE estimation [158] and the result is passed to the conventional OFDM receiver for decoding and detecting. The preprocessing techniques demonstrate limited performance gain, particularly at low signal to impulsive noise ratio (SINR) [159] and also for higher order modulations [158]. The sparse signal recovery techniques assume the sparsity of the unknown impulsive noise vector and attempt to reconstruct the impulsive noise by exploiting the information in known tones [8], [9]. The method in [8], requires the time domain interleaving (TDI) which is not standard-compliant. TDI also introduces a significant delay in the receiver detection procedure and requires significant memory to store the continuous time-domain signal. Also, these techniques are mostly non-parametric and they only consider the sparsity of the impulsive noise without taking into account a particular model for impulsive noise [8], [182]. In contrast, the parametric methods select a particular model for impulsive noise and design the receiver according to its statistical properties. Given a good match for modeling the underlying impulsive noise the parametric methods generally outperform the nonparametric methods [8]. Although the technique in [9] considers a statistical model for sparse impulsive noise, the parameters
of the impulsive noise are assumed to be known.

Using the novel Block Iterative Bayesian Algorithm (Block-IBA) presented in Chapter 3, this chapter presents a new impulsive noise reduction method for OFDM systems. The method utilizes the guard band null subcarriers and data subcarriers for the impulsive noise estimation and cancellation. Unlike some other general OFDM transceivers [8], [131], [132], which use time-domain interleaving (TDI) to cancel impulsive noise, we design a specific receiver for bursty impulsive noise channels that removes the delay due to TDI and saves memory space. The proposed receiver is a parametric approach, which represents the bursty impulsive noise by the practical Bernoulli-Gaussian Hidden Markov model (BGHMM) [1]. The Block-IBA first automatically estimates the variance and the transition matrix of Markov chain model for the impulsive noise. Towards that end, it uses a maximum a posteriori (MAP) approach, which averts complicated tuning updates. It then iteratively estimates the amplitudes and positions of the block-sparse impulsive noise using the steepest-ascent based Expectation-Maximization (EM), and optimally selects the nonzero elements of the block-sparse impulsive noise by adaptive thresholding. Numerical experiments show that the proposed receiver outperforms existing receivers under the block-sparse impulsive noise environment.

5.2 System Model

5.2.1 OFDM Transmission Model

Consider the conventional coded discrete Fourier Transform (DFT)-based OFDM system whose complex baseband equivalent representation shown in Figure 5.1. Each OFDM symbol has $N$ tones consisting of $N_d$ data tones and $N_n$ non-data tones, with $N = N_d + N_n$. The non-data tones comprise either null-tones or pilot tones. Null-tones
are usually used for spectral shaping or mitigation of inter-carrier or radio service interference, while pilot tones are utilized for channel estimation and synchronization. At the transmitter, the information bit stream $b = [b_1, \ldots, b_{M_i}]^T$ with length $M_i = R_c R_m N_d$ is encoded to $c = [c_1, \ldots, c_{M_c}]^T$ with length $M_c = R_m N_d$, where $R_c$ and $R_m$ are the code and modulation rates, respectively. The output of the encoder (i.e. $c$ vector) is then mapped into $N_d$ signal points $q = [q_1, \ldots, q_{N_d}]^T$. The elements of $q$ are chosen from $2^{R_m}$-ary constellation $\mathcal{Q}$. The $N_n$ non-data tones are then inserted to expand $q$ and form the OFDM symbol $x = [x_1, \ldots, x_N]^T$.

The OFDM modulation is performed by applying the inverse of the unitary $N \times N$ (IDFT) matrix $F$ to OFDM symbol $x$ to convert it to the time domain signal $x_t = F^H x$. Finally, a cyclic prefix (CP) which is longer than the channel delay spread is prepended to $x_t$ to eliminate the inter-symbol interference (ISI) from the received signal.

The time domain signal $x_t$ propagates through the discrete-time complex baseband channel with impulse response $h_{ch}$ and then is contaminated by both additive white Gaussian noise (AWGN) and impulsive noise. At the receiver, removing the CP results in

$$r = H x_t + e + z = H F^H x + e + z,$$

where $H \in \mathbb{C}^{N \times N}$ is the column-circulant matrix due to cyclic prefix prepending with the first column, $h$, formed by zero-padded discrete-time channel impulse response $h_{ch}$, and $e, z \in \mathbb{C}^N$ denote impulsive noise and background AWGN, respectively. From the matrix decomposition, the column-circulant convolution matrix $H$ is represented by $H = F^H D F$, where $D \triangleq \text{diag}(\sqrt{N} F h)$.

The OFDM demodulation is performed by applying a DFT to obtain the
frequency-domain received signal as

\[ y = \mathbf{F}^H \mathbf{F} \mathbf{x} + \mathbf{F} \mathbf{e} + \mathbf{F} \mathbf{z} = \mathbf{D} \mathbf{x} + \mathbf{F} \mathbf{e} + \tilde{\mathbf{z}}, \]  

(5.2)

where \( \tilde{\mathbf{z}} \triangleq \mathbf{F} \mathbf{z} \) is the DFT of \( \mathbf{z} \) which is also AWGN satisfying \( \tilde{\mathbf{z}} \sim \mathcal{CN}(0, \sigma_n^2 \mathbf{I}_N) \).

Because of the term \( \mathbf{F} \mathbf{e} \), (5.2) is different from the standard AWGN OFDM system. If we can accurately estimate the impulsive noise \( \mathbf{e} \) and subtract it from the received signal \( y \), then the decoder output will be the same as the output of the standard AWGN OFDM system without the influence of block-sparse impulsive noise \( \mathbf{e} \). To do this, a model for estimation of \( \mathbf{e} \) is needed and will be developed in the next subsection.

### 5.2.2 Impulsive Noise Model

In general, impulsive noise model is controlled by a process that switches from zero outputs to burst outputs of certain lengths. Therefore, the appropriate model of the impulsive noise can be completely determined by the switching law and the sample distributions. In addition, the sampling frequency controls the number of samples in impulses which could be one sample or a burst of samples. This in turn results in
an i.i.d. model or non-i.i.d. model (block sparse or bursty model) of the impulsive noise. Many memoryless models (see e.g., Table 2.1) consider the sparse samples of the impulsive noise to be i.i.d. However, in many applications such as PLC, the noise has bursty structure with correlated samples and hence, the noise model requires some memory to represent the correlation between the samples that in turn captures the bursty structure of the noise. The Markov chain as an appropriate statistical tool satisfies the non-i.i.d. property of the block-sparse impulsive noise by performing the suitable switching process and varying the shape of the impulses. The switching process between the zero output and impulsive output is performed by the transition probabilities between the 0 and 1 states in Markov chain (see Figure 5.2). Also, the shape of the impulses is determined by the probability of remaining in the impulse state (State 1 in Figure 5.2). In this chapter, we also use Markov chain to model the realistic bursty structure of the impulsive noise.

To model the block-sparse impulsive noise $\mathbf{e}$, we introduce two hidden random processes, $\mathbf{s}$ and $\mathbf{\theta}$ [53], [184]. The binary vector $\mathbf{s} \in \{0,1\}^N$ describes the support of $\mathbf{e}$, denoted as $\mathcal{S}$, while the vector $\mathbf{\theta} \in \mathbb{C}^N$ represents the amplitudes of the active elements of $\mathbf{e}$. Hence, each element of the impulsive noise vector $\mathbf{e}$ can be characterized as

$$e_i = s_i \cdot \theta_i,$$  \hspace{1cm} (5.3)

where $s_i = 0$ gives $e_i = 0$ for $i \notin \mathcal{S}$ and $s_i = 1$ gives $e_i = \theta_i$ for $i \in \mathcal{S}$. In vector form,(5.8) can be written as

$$\mathbf{e} = \mathbf{S} \mathbf{\theta}, \quad \mathbf{S} = \text{diag}(\mathbf{s}) \in \mathbb{R}^{N \times N}.$$  \hspace{1cm} (5.4)

To generate the bursty impulsive noise $\mathbf{e}$, we follow the following two steps.
In the first step, the binary vector $s$ is produced by a stationary first-order Markov process defined by two transition probabilities: $p_{10} \triangleq \Pr \{s_{i+1} = 1 | s_i = 0\}$ and $p_{01} \triangleq \Pr \{s_{i+1} = 0 | s_i = 1\}$ (see Figure 5.2), where $i = 1, \cdots, N - 1$. In other words, the elements of the vector $s$ which are indexed by $i \in [1, N - 1]$, are iteratively and stochastically specified to fall into the sets of zero or non-zero (one) based on the index $(i - 1)$. Particularly, $s_2$ is determined by $s_1$, which could be zero or one (see Figure 5.3 for detail) \(^1\). In fact, the memory of the noise is approximated by $1/p_{01}$ when the noise comprises a few impulses, i.e., $0.9 \leq p < 1$, which also represents the average number of consecutive ones in Markov model. In the second step, the amplitudes $\theta_i$ of nonzero elements in vector $s$ are drawn independently and identically from Gaussian distribution $\theta \sim \mathcal{CN}(0, \sigma_\theta^2 I_N)$, where $\sigma_\theta^2$ is the variance of the Gaussian amplitudes $\theta_i$.

Based on the Markov process, the sparsity and block size of the impulsive noise $e$ is summarized in the following property.

Property 1: For a given impulsive noise $e$ with parameters $p_{10}$ and $p_{01}$, the average percentage of zero and nonzero coefficients, the average block size of nonzero and zero coefficients of the generated impulsive noise, denoted by $\overline{S_z}, \overline{S_{nz}}, \overline{B_{nz}}$ and

\(^1\)In the trellis diagram in Figure 5.3 the first sample (i.e. $s_1$) is considered zero which could be one too.
\( B_z, \text{ respectively, are given as} \)

\[
\overline{S}_z = p = \frac{p_{01}}{p_{10} + p_{01}}, \quad \overline{S}_{nz} = 1 - p = \frac{p_{10}}{p_{10} + p_{01}}, \quad (5.5)
\]

\[
\overline{B}_{nz} = \frac{1}{p_{01}}, \quad \overline{B}_z = \frac{1}{p_{10}}, \quad (5.6)
\]

Hence, we can define the noise memory as \( \eta = \frac{1}{p_{10} + p_{01}} \) [146]. It can be shown that when \( \eta = 1 \) the noise is memoryless, which can happen for example when \( p = 0.9, \ p_{10} = 0.1 \) and consequently \( p_{01} = 0.9 \). In this case, the two-state Markov chain is equivalent to Bernoulli-Gaussian model with i.i.d samples. On the other hand, it can be shown that when \( \eta > 1 \), the noise has a persistent memory. In this case, the average permanence in a given state is longer compared to the memoryless case. For instance, when \( p = 0.9, \ p_{10} = 0.01 \) and \( p_{01} = 0.09 \), the noise memory is \( \eta = 10 \). If block sparse impulsive noise comprises a few impulses, i.e. \( 0.9 \leq p < 1 \), then \( p_{01} \gg p_{10} \) and the noise memory can be approximated by \( \eta \approx 1/p_{01} \). Figure 5.4 shows the realizations of the BGHMM for different values of noise memory \( \eta = \frac{1}{p_{10} + p_{01}} \), when \( p \) and impulsive to background noise power ratio (INR) are fixed (i.e., \( p = 0.9 \) and INR = 20dB). It is seen that, while the memoryless (i.i.d.) model of noise with \( \eta = 1 \) is unable to describe the bursty structure of the noise, BGHMM with \( \eta > 1 \) is able to do that. In fact, BGHMM with \( \eta > 1 \) allows to set the average duration of the bursts by modifying

\[ \text{Figure 5.3: Trellis diagram of block-sparse impulsive noise.} \]
Figure 5.4: Three realization of the noise model generated by BGHMM for different values of noise memory $\eta$. The impulsive to background noise power ratio (INR) is 20 dB.

the parameter $\eta$. When block sparse impulsive noise comprises a few impulses (i.e. $0.9 \leq p < 1$), $p_{01} \gg p_{10}$ and the noise memory can be approximated by $\eta \approx 1/p_{01}$.

From (5.8) it is obvious that
\[ p(e_i|s_i, \theta_i) = \delta(e_i - s_i\theta_i), \]
where $\delta(\cdot)$ is the Dirac delta function. Removing $s_i$ and $\theta_i$ by the marginalization rule, we can find the PDF of the impulsive noise as
\[ p(e_i) = p\delta(e_i) + (1 - p)\mathcal{CN}(e_i; 0, \sigma_\theta^2), \]  
(5.7)

where $\sigma_\theta^2$ is the variance of $\theta$. Equation (5.7) shows that the distribution of the sources is a Bernoulli-Gaussian hidden Markov model (BGHMM) which is utilized to implicitly express the block sparsity of the noise model due to the point-mass
distribution at $e_i = 0$ and the hidden variables $s_i$. BGHMM is a special form of Gaussian Mixture Hidden Markov model (GHMM). It is shown in [178] that with properly tuned parameters, GHMM can approximate rather precisely the PDF of additive non-Gaussian noise in many communication systems such as PLC.

Unlike the memoryless models such as Bernoulli-Gaussian model [148] which consider the impulsive noise samples to be i.i.d., the BGHMM [143], [187], [146] with the first-order Markov chain model allows to better describe the typical bursty nature of impulsive noise with non-i.i.d. samples.

Although higher-order Markov processes [144] and/or more complex GM model can be utilized to model the impulsive noise $e$, we focus on the first-order Markov processes and Bernoulli-Gaussian model to reduce the complexity in the development of the proposed impulsive noise reduction algorithm.

5.3 Formulation of Impulsive Noise Receiver

5.3.1 Problem Statement

Denote the index sets of the non-data (null and pilot) tones and the data tones as $\mathcal{T}$ with cardinality $|\mathcal{T}| = M < N$ and $\overline{\mathcal{T}}$ with cardinality $|\overline{\mathcal{T}}| = N - M$, respectively. Denote $(\cdot)_\mathcal{T}$ the sub-matrix or sub-vector corresponding to the rows or elements indexed by the set $\mathcal{T}$. Since the focus of this paper is not the channel estimation, the pilot tones are treated as zero tones, i.e. null tones. We can utilize the null tones to estimate the bursty impulsive noise $e$ from the following equation

$$y_\mathcal{T} = F_\mathcal{T} e + \tilde{z}_\mathcal{T},$$

(5.8)
where the AWGN $\tilde{z}_T$ is a subsampling of $\tilde{z}$ and is an AWGN vector of length $M$, i.e. $\tilde{z}_T \sim CN(0, \sigma_n^2 I_M)$.

Estimating the impulsive noise vector $e$ with length $N$ from the underdetermined system of linear equations (5.8) is generally an ill-posed problem. It is a challenging problem with infinite number of solutions and has been investigated as a central problem in the Compressed Sensing (CS) literature. To recover $e$, appropriate prior knowledge such as the sparsity of $e$ (i.e. the majority of the elements of the time-domain vector $e$ are zero or near zero while only a few components are nonzero) is needed. Knowing the sparsity of vector $e$ \textit{a priori}, a theoretically proven and practically effective approach to recover the signal $e$ is to solve the following optimization problem

$$\hat{e} = \arg \min_{e} \beta \| y_T - F_T e \|^2_2 + \tau \| e \|_1,$$

(5.9)

where $\beta = \frac{1}{2} \sigma_n^{-2}$ and $\tau$ is the regularization parameter controlling the degree of the sparsity of the solution. Some popular optimization algorithms have been developed to solve (5.9) [13], [20], [28], [29]. In some works such as [28], [29] the developed sparse reconstruction algorithms used the $\ell_p$-norm to replace the $\ell_1$-norm where $0 < p \leq 1$. The drawback of these approaches is that they only utilized the sparsity of the impulsive noise $e$ without considering any \textit{a priori} statistical information or specific structure on the impulsive noise to be estimated. If the structure of the impulsive noise is further exploited, the better recovery performance can be expected. A block-sparse signal, in which the nonzero samples manifest themselves as clusters, is an important structured sparsity. The CS problem for block-sparse signals is to estimate $e \in \mathbb{C}^N$ with the cluster structure

$$e = \left[ e^{T[1] \{1\}}, \ldots, e^{T[1] \{d_g - 1\} + 1}, \ldots, e^{T[1] \{g\}} \right]^T,$$

(5.10)
where \( e[i] \), \( i = 1, \ldots, g \), denote the blocks with length \( d_i \) which are not necessarily identical. In the block partition (5.10), only \( k \ll g \) vectors \( e[i] \) have nonzero Euclidean norm.

We aim to improve the detection of the frequency domain signal \( x \) with more accurate estimation of impulsive noise \( e \). Consider \( \hat{e} \) as the estimate of \( e \) obtained from a block-sparse reconstruction algorithm which solves (5.8). The impulsive noise estimate \( \hat{e} \) can be subtracted from the received signal so that the decision metric received by the channel decoder is expressed as

\[
\hat{y} = Dx + F(e - \hat{e}) + \tilde{z}.
\]  \hfill (5.11)

The block diagram of the proposed communication system is shown in Figure 5.5, which includes the block-sparse impulsive noise mitigation in the receiver. If the reconstruction algorithm is capable of accurately estimating the bursty impulsive noise \( e \) (i.e. \( e \approx \hat{e} \)), the decoder decodes the received signal \( \hat{y} \) as if it was the output of the standard AWGN OFDM system without the presence of block-sparse impulsive noise \( e \). This leads to a significant gain when it is compared to (5.2) where the receiver is subject to the received signal corrupted by both AWGN and impulsive noise \( e \). To sum up, the estimation of bursty (block-sparse) impulsive noise \( e \) in (5.2) is cast into solving an underdetermined system of linear equations problem in (5.8) with block-sparsity as a prior knowledge about the impulsive noise \( e \). In the following subsection, we first elaborate on the optimum estimate of \( e \). Then, in Section 5.4 we propose an efficient block-sparse signal reconstruction algorithm (Block-IBA) for bursty impulsive noise estimation.
Figure 5.5: The proposed coded OFDM system with block-sparse impulsive noise (IN) mitigation in the receiver.

5.3.2 Estimation of \( e \)

To obtain the optimum estimate of \( e \) in (5.8), we pursue a MAP approach. It first determines the MAP estimate of \( s \) which maximizes the posterior probability \( p(s|y_T) \). After estimating \( s \), the estimation of bursty impulsive noise \( e \) is accomplished by the estimation of \( \theta \).

5.3.2.1 MAP Estimation of \( s \)

Using the Bayes’ rule, we can rewrite \( p(s|y_T) \) as

\[
p(s|y_T) = \frac{p(s)p(y_T|s)}{\sum_s p(s)p(y_T|s)},
\]

where the summation is over all the possible \( s \) vectors describing the support of \( e \). Note that the denominator in (5.12) is a normalizing constant and thus can be ignored. To evaluate \( p(s) \), it is known that the \( s \) vector is a stationary first-order Markov process with two transition probabilities given in Section 5.2. Therefore, \( p(s) \) is given by

\[
p(s) = p(s_1) \prod_{i=1}^{N-1} p(s_{i+1}|s_i),
\]

(5.13)
where \( p(s_1) = p^{(1-s_1)}(1-p)^{s_1} \) and

\[
p(s_{i+1}|s_i) = \begin{cases} 
(1 - p_{10})^{(1-s_{i+1})}p_{10}^{s_{i+1}} & \text{if } s_i = 0, \\
p_{01}^{(1-s_{i+1})}(1 - p_{01})^{s_{i+1}} & \text{if } s_i = 1,
\end{cases}
\]

(5.14)

where \( p_{10} \) and \( p_{01} \) can be calculated using the MAP estimation approach given in (5.53) and (5.54) in Section IV-C.

It remains to calculate \( p(y_T|s) \). Given the support vector \( s, e \) is Gaussian. It follows from (5.8) that \( y_T \) is also Gaussian with zero mean and the covariance

\[
\Sigma_s = \mathbb{E}[yy^H|s] = \sigma_n^2 I_M + \sigma_0^2 y_T S y_T^H,
\]

(5.15)

where \( S = \text{diag}(s) \) as defined in (5.4). Therefore, we can write the likelihood function as

\[
p(y_T|s) = \frac{\exp \left(-\frac{1}{2}y_T^H \Sigma_s^{-1} y_T \right)}{\det(\Sigma_s)},
\]

(5.16)

up to an inessential multiplicative constant factor. Hence, the MAP estimate of \( s \) is given by

\[
s_{\text{MAP}} = \arg\max_s p(s)p(y_T|s),
\]

(5.17)

where \( p(s) \) is calculated using (5.13) and (5.14), whereas the prior likelihood \( p(y_T|s) \) is given by (5.16). The maximization is performed by an exhaustive search over all \( 2^N \) possible sets of \( s \) vectors (forming a discrete space), which is a computationally daunting task for large values of \( N \). However, by exploiting the block sparsity of the signal and using steepest-ascent method, in Section 5.4 we propose an efficient iterative Bayesian algorithm (i.e., Block-IBA) which intelligently averts this complicated combinatorial search and computation.
5.3.2.2 MAP Estimation of $\theta$ using Gamma Prior

After the binary vector $s$ is estimated, we complete the estimation of the bursty impulsive noise $e$ by estimating the amplitude samples of the $\theta$ vector. To this end, we estimate the amplitudes with considering hyperprior over the inverse of the variance. The details are given below.

Following the Sparse Bayesian Learning (SBL) framework [58], we consider a Gaussian prior distribution for amplitude vector $\theta$

$$p(\theta; \gamma_i) \sim \mathcal{CN}(0, \Sigma_0^{-1}),$$  \hspace{1cm} (5.18)

where $\Sigma_0 = \text{diag}(\gamma_1, \gamma_2, \cdots, \gamma_N)$. Furthermore, $\gamma_i$ are the non-negative real-valued elements of the hyperparameter vector $\gamma$, that is $\gamma \triangleq \{\gamma_i\}$. Based on the SBL framework, we use Gamma distributions as hyperpriors over the hyperparameters $\{\gamma_i\}$

$$p(\gamma) = \prod_{i=1}^{N} \text{Gamma}(\gamma_i | a, b) = \prod_{i=1}^{N} \Gamma(a)^{-1} b^a \gamma_i^{a-1} e^{-b\gamma_i},$$  \hspace{1cm} (5.19)

where $\Gamma(a) = \int_{0}^{\infty} t^{a-1}e^{-t}dt$ is the Gamma function and $a = b = 10^{-4}$ [58]. From (5.4), we can rewrite (5.8) as

$$y_T = F_T S \theta + \bar{z}_T = \Psi \theta + \bar{z}_T,$$  \hspace{1cm} (5.20)

where $\Psi = F_T S$. Therefore, from the linear model of (5.20) and given the support vector $s$, the likelihood function also has Gaussian distribution

$$p(y_T \mid \theta; \sigma_n^2) \sim \mathcal{CN}_{y_T \mid \theta} \left( \Psi \theta, \sigma_n^2 I_M \right).$$  \hspace{1cm} (5.21)

Using the Bayes’ rule the posterior approximation of $\theta$ is found as a multivariate
Gaussian:

\[ p(\theta \mid y_T; \gamma, \sigma_n^2) \sim \mathcal{CN}_\theta (\mu_\theta, \Sigma_\theta), \quad (5.22) \]

with parameters

\[
\begin{align*}
\mu_\theta &= \sigma_n^{-2} \Sigma_\theta \Psi^H y_T, \\
\Sigma_\theta &= (\sigma_n^{-2} \Psi^H \Psi + \Sigma_0)^{-1} \\
&= \Sigma_0^{-1} - \Sigma_0^{-1} \Psi^H \left( \sigma_n^{-2} I_M + \Psi \Sigma_0^{-1} \Psi^H \right)^{-1} \Psi \Sigma_0^{-1}.
\end{align*}
\quad (5.23) \quad (5.24) \quad (5.25)
\]

Therefore, given the hyperparameters \( \gamma_i \) and noise variance \( \sigma_n^2 \), the MAP estimate of \( \theta \) is the posterior mean \( \mu_\theta \), i.e. \( \hat{\theta}_{MAP} = \mu_\theta \).

The hyperparameters \( \gamma_i \) control the sparsity of the amplitudes \( \theta_i \). Sparsity in the samples of the amplitudes occur when particular variables \( \gamma_i \rightarrow \infty \), whose effect forces the \( i \)th sample to be pruned out from the amplitude estimate. To calculate \( \hat{\theta}_{MAP} \) from (5.23), we have two options for obtaining the covariance matrix \( \Sigma_\theta \) using (5.24) and (5.25). Note that, the computational complexity for estimation of \( \Sigma_\theta \) is different in (5.24) and (5.25). An \( N \times N \) matrix inversion is required using (5.24), whereas an \( M \times M \) matrix inversion is needed in (5.25).

When the noise variance \( (\sigma_n^2) \) is also unknown, we can place conjugate gamma prior on the inverse of the variance (i.e. \( \beta \triangleq \sigma_n^{-2} \)) as \( p(\beta) = \text{Gamma}(\beta \mid c, d) \), where \( c = d = 10^{-4} \). To estimate the hyperparameters, we utilize the Relevance Vector Learning (RVL) which is maximization of the product of the marginal likelihood (Type-II maximum likelihood) and the priors over the hyperparameters \( \gamma \) and \( \beta \) \( (\sigma_n^{-2}) \) [58]. Given the priors, the likelihood of the observations can be given as

\[
p(y_T \mid \gamma, \beta) = \mathcal{CN} (y_T \mid 0, \Psi \Sigma_0^{-1} \Psi^H + \beta^{-1} I_M) \times p(\gamma) \times p(\beta).
\quad (5.26)
\]
A maximum likelihood (ML) estimator which maximizes (5.26) can be used to find the unknown hyperparameter $\gamma$ and $\beta$. Utilizing the ML estimator [58], the updates for hyperparameters $\gamma$ and $\beta$ can be expressed, respectively as

$$
\gamma^{(k+1)}_i = \frac{1 + 2a}{\left( \mu^{(k)}_{\theta,i} \right)^2 + \Sigma^{(k)}_{\theta,ii} + 2b},
$$

and

$$
\frac{1}{\beta^{(k+1)}} = \frac{1}{M + 2c} \left\{ \left\| Y_T - \Psi \mu^{(k)}_{\theta} \right\|^2_2 + \left( \beta^{(k)} \right)^{-1} \sum_{i=1}^{N} \left[ 1 - \gamma^{(k)}_i \Sigma^{(k)}_{\theta,ii} \right] + 2d \right\},
$$

where $\mu_{\theta,i}$ denotes the $i$th entry of $\mu_{\theta}$ in (5.23) and $\Sigma_{\theta,ii}$ denotes the $i$th diagonal element of the covariance matrix $\Sigma_{\theta}$ in (5.24) or (5.25).

Having estimated the posterior probability of $s$ and MAP estimate of amplitude vector $\theta$, the estimation of the impulsive noise $e$ is complete. However, the evaluation of (5.17) over all $2^N$ possible sets of $s$ vectors is a computationally daunting task when $N$ is large. The difficulty of this exhaustive search is obvious from (5.12)-(5.17). Hence, in the following section, we propose an Iterative Bayesian Algorithm, referred to as Block-IBA, which reduces the complexity of the exhaustive search.

5.4 Block Iterative Bayesian Algorithm for Impulsive Noise Estimation

5.4.1 Block Iterative Bayesian Algorithm (Block-IBA) Using Null Tones

Finding the solution for (5.17) through combinatorial search is computationally intensive. This is because the computation should be done over the discrete space. One way to avoid this exhaustive search is to convert the maximization problem into a continuous form. In this section, we propose a method to convert the problem into
a continuous maximization problem and apply a steepest-ascent algorithm to find the maximum value. To this end, we model the elements of \( \mathbf{s} \) vector as a Gaussian Mixture (GM) with two Gaussian variables centered around 0 and 1 having variances sufficiently smaller than 1, e.g. \( 0.05^2 \), so that the two Gaussian distributions have small overlap. Hence, each discrete element of \( \mathbf{s} \) vector, \( s_i \), can be given as

\[
p(s_1) \approx p \mathcal{N}(0, \sigma_0^2) + (1 - p) \mathcal{N}(1, \sigma_0^2).
\]

(5.29)

Moreover, the other elements of \( \mathbf{s} \) vector, \( s_{i+1} \) \((i = 1, \ldots, N - 1)\), can be expressed as

\[
p(s_{i+1}) \approx \begin{cases} 
(1 - p_{10}) \mathcal{N}(0, \sigma_0^2) + p_{10} \mathcal{N}(1, \sigma_0^2) & \text{if } s_i = 0, \\
p_{01} \mathcal{N}(0, \sigma_0^2) + (1 - p_{01}) \mathcal{N}(1, \sigma_0^2) & \text{if } s_i = 1.
\end{cases}
\]

(5.30)

In order to find the global maximum of (5.17) we decrease the variance \( \sigma_0^2 \) in each iteration of the algorithm gradually to avert the local maximum of (5.17). Implementation of this process is discussed after (5.38). Although we have converted the discrete variables \( s_i \) to the continuous form, finding the optimal value of \( \mathbf{s} \) using (5.17) is still complicated. Thus, we propose an algorithm that estimates the impulsive noise \( \mathbf{e} \) by estimating its components (\( \mathbf{s} \) and \( \mathbf{\theta} \) in (5.4)) iteratively. We follow a two-step approach to estimate the \( \mathbf{e} \) vector. In the first step, we estimate the amplitude vector \( \mathbf{\theta} \) (i.e. \( \hat{\mathbf{\theta}} \)) based on the known estimation of \( \mathbf{s} \) (i.e. \( \hat{\mathbf{s}} \)) vector and the mixing observation vector \( \mathbf{y}_\tau \) (see Algorithm 5.1). We call this expectation step (E-step). Given the estimate of \( \mathbf{s} \) denoted as \( \hat{\mathbf{s}} \), the estimate of amplitude vector \( \mathbf{\theta} \) can be derived as

\[
\hat{\mathbf{\theta}} = \left( \hat{\mathbf{\Psi}}^\mathsf{H} \hat{\mathbf{\Psi}} + \sigma_n^2 \Sigma_\theta \right)^{-1} \hat{\mathbf{\Psi}}^\mathsf{H} \mathbf{y}_\tau,
\]

(5.31)

where \( \hat{\mathbf{\Psi}} = F_\tau \hat{\mathbf{S}}. \)
We call the second step of our approach the maximization step (M-step). In this step, we find the estimate of $s$ with the assumption of known vector $\hat{\theta}$ and the observation vector $y_T$. From (5.17), the MAP estimate of $s$ can be written as

$$
\hat{s}_{\text{MAP}} = \arg\max_s p(s \mid y_T, \hat{\theta}) \equiv \arg\max_s p(s \mid \hat{\theta}) p(y_T \mid s, \hat{\theta}) \equiv \arg\max_s p(s) p(y_T \mid s, \hat{\theta}) = \arg\max_s \log(p(s)) + \log(p(y_T \mid s, \hat{\theta})).
$$

(5.32)

After calculating the two summands in (5.32) (see Section 3.4.1 for details), we can express the M-step as

$$
\hat{s} = \arg\max_s \mathcal{L}(s),
$$

(5.33)

where

$$
\mathcal{L}(s) = \log(p(s_1)) + \sum_{i=1}^{N-1} \log(p(s_{i+1} \mid s_i)) - \frac{\|y_T - \Psi\hat{\theta}\|^2}{2\sigma_n^2}.
$$

(5.34)

We can find the optimal solution of (5.33) by performing the steepest-ascent method. The expression for obtaining the sequence of optimal solutions in this method can be given as

$$
s^{(k+1)} = s^{(k)} + \mu \frac{\partial \mathcal{L}(s)}{\partial s} \bigg|_{s=s^{(k)}},
$$

(5.35)

where $\mu$ is the step-size parameter of the steepest-ascent method. The gradient term in (5.35) can be expressed in a closed form (see Appendix G). Therefore (5.35) can be rewritten as

$$
s^{(k+1)} = s^{(k)} + \frac{\mu}{\sigma_0^2} g(s) + \frac{\mu}{\sigma_n^2} \text{diag}(\Psi\hat{\theta} - y_T) \cdot \hat{\theta},
$$

(5.36)

where $g(s) = g_1(s) + g_2(s)$ as derived in Appendix G. Moreover, the two scalar
functions \( g_1(s_1) \) and \( g_2(s_{i+1}) \), \( i = 1, 2, \ldots, N - 1 \), can be given as

\[
g_1(s_1) = \frac{p s_1 \exp\left(-\frac{s_1^2}{2\sigma_0^2}\right) + (1-p)(s_1 - 1) \exp\left(-\frac{(s_1-1)^2}{2\sigma_0^2}\right)}{p \exp\left(-\frac{s_1^2}{2\sigma_0^2}\right) + (1-p) \exp\left(-\frac{(s_1-1)^2}{2\sigma_0^2}\right)},
\]

\( (5.37) \)

\[
g_2(s_{i+1}) = \frac{q_1 s_{i+1} \exp\left(-\frac{s_{i+1}^2}{2\sigma_0^2}\right) + q_2(s_{i+1} - 1) \exp\left(-\frac{(s_{i+1}-1)^2}{2\sigma_0^2}\right)}{q_1 \exp\left(-\frac{s_{i+1}^2}{2\sigma_0^2}\right) + q_2 \exp\left(-\frac{(s_{i+1}-1)^2}{2\sigma_0^2}\right)},
\]

\( (5.38) \)

where \( q_1 = p_{01} + (1-p_{10}) \) and \( q_2 = p_{10} + (1-p_{01}) \). To guarantee the global maxima of (5.34) in the computation, we decrease \( \sigma_0 \) in the consecutive iterations using \( \sigma_0^{(k+1)} = \alpha \sigma_0^{(k)} \), with a constant \( \alpha \) chosen from \( \alpha \in [0.6, 1] \). Also, the range for step-size parameter \( \mu \) can be expressed as (see Section 3.6.2)

\[
0 < \mu < \frac{2}{\sigma_0^2 + \frac{NM^2}{\sigma_0^2}},
\]

\( (5.39) \)

where \( M^* = \sigma_0 Q^{-1}(\frac{\sqrt{0.99}}{2}) \) and \( Q^{-1}(\cdot) \) is the inverse Gaussian Q-function.

We initialize the proposed Block-IBA with the minimum \( \ell_2 \)-norm solution and use a decreasing threshold \( Th^{(k+1)} = \alpha Th^{(k)} \), with a constant \( \alpha \) chosen from \( \alpha \in [0.6, 1] \), so that the subsampling matrix \( F_T \) maximally samples the nonzero elements of the impulsive noise \( e \). In fact, the value of \( Th \) optimally selects the number of nonzero elements in \( s \) vector.

### 5.4.2 Extension of Block-IBA using All Tones

Simulation results demonstrate the improvement of the estimation using null tones with increased number of non-data subcarriers (see Figure 5.6). However, increasing the number of non-data tones is limited by the bandwidth and the throughput of the OFDM system. Hence, to avoid increasing the number of non-data tones and
still have a reasonable estimation performance, it is helpful to utilize the available information in all subcarriers in order to perform a more accurate impulsive noise estimation. To this end, we use the similar method in [8] which has been used for impulsive noise estimation. Define $g \triangleq Dx + z$, then (5.2) can be rewritten as

$$
\begin{bmatrix}
y_T \\
y_T
\end{bmatrix} = Fe + \begin{bmatrix}
g_T \\
g_T
\end{bmatrix},
$$

(5.40)

where $g_T \sim \mathcal{CN}(0, \sigma^2_n I_M)$ and $g_T \sim \mathcal{CN}((Dx)_T, \sigma^2_I I_M)$.

Assuming the same Gaussian prior distribution for amplitude vector $\theta$, i.e. $p(\theta; \gamma_i) \sim \mathcal{CN}(0, \Sigma^{-1}_0)$ and Gamma distribution as hyperpriors over hyperparameters $\gamma$ and $\beta$, the likelihood of the observations $y_T$ remains the same as (5.26). However, the likelihood of observations $y_T$ which is the observations of the received signal based on the data subcarriers is given as

$$
p(y_T | \gamma, \beta) = \mathcal{CN}(y_T | (Dx)_T, \Sigma_{y_T}) \times p(\gamma) \times p(\beta),
$$

(5.41)

where $\Sigma_{y_T} = \Lambda \Sigma^{-1}_0 \Lambda^H + \beta^{-1} I_{N-M}$ and $\Lambda = F_{\tau} S$. Hence, along with the unknown hyperparameters $\gamma$ and $\beta$, the unknown transmitted data tones $x_T$ should be estimated in Block-IBA algorithm. Since $x_T$ comprises discrete constellation points, we temporarily relax it to be continuous for carrying out EM algorithm. After the convergence of EM algorithm is obtained, hard decision (on $(Dx)_T$) is used in the receiver before feeding it into the decoder. As a result, keeping the hyperparameters $\gamma$ and $\beta$ constant and treating $(Dx)_T$ as an unknown hyperparameter the update
equation for \((\mathbf{Dx})_T\) is given as

\[
\begin{align*}
(\mathbf{Dx})^{(k+1)}_T &= \arg\max_{(\mathbf{Dx})_T} \mathbb{E}_{\theta,y,\Theta^{(k)}} [\log(p(y \mid \theta, \beta, x) \times p(\theta \mid \gamma) p(\gamma) p(\beta)) ] \\
&= \arg\max_{(\mathbf{Dx})_T} \mathbb{E}_{\theta,y,\Theta^{(k)}} [\log(p(y \mid \theta, \beta, x))] \\
&= \arg\min_{(\mathbf{Dx})_T} \left\| y_T - (\mathbf{Dx})_T - F_T \mu^{(k)}_{\theta} \right\|_2^2 \\
&= y_T - F_T \mu^{(k)}_{\theta},
\end{align*}
\]

(5.42)

where \(\Theta \triangleq \{\gamma, \beta\}\) and \(\mu^{(k)}_{\theta}\) is given in (5.45).

Similarly, from (5.42), keeping \(x\) and \(\beta\) as the fixed parameters, the same equation as (5.27) can be obtained for \(\gamma\), i.e.

\[
\gamma_i^{(k+1)} = \frac{1 + 2a}{\left( \mu_{\theta,i}^{(k)} \right)^2 + \Sigma_{\theta,ii}^{(k)} + 2b},
\]

(5.43)

where \(\mu_{\theta,i}\) denotes the \(i\)th entry of \(\mu_{\theta}\) in (5.45) and \(\Sigma_{\theta,ii}\) denotes the \(i\)th diagonal element of the covariance matrix \(\Sigma_{\theta}\) in (5.46). Further, an update for \(\beta\) can be given with a minor change in (5.28) as

\[
\frac{1}{\beta^{(k+1)}} = \frac{1}{N + 2c} \left\{ \left\| y - (\mathbf{Dx})^{(k)} - \Psi' \mu^{(k)}_{\theta} \right\|_2^2 \\
+ (\beta^{(k)})^{-1} \sum_{i=1}^{N} \left[ 1 - \gamma_i^{(k)} \Sigma_{\theta,ii}^{(k)} \right] + 2d \right\},
\]

(5.44)

where \(\Psi' = FS\). After estimating the hyperparameters \(\gamma\) and \(\beta\), the MAP estimate of \(\theta\) is given as

\[
\hat{\theta}_{MAP}^{(k)} = \mu_{\theta}^{(k)} = \left( \sigma_n^{-2} \right)^{(k)} \Sigma_{\theta}^{(k)} \Psi^H \left( y - (\mathbf{Dx})^{(k)} \right),
\]

(5.45)
where
\[ \Sigma_{\theta}^{(k)} = (\Sigma_0^{-1})^{(k)} - (\Sigma_0^{-1})^{(k)} \Psi^H \Sigma_y^{-1} \Psi' (\Sigma_0^{-1})^{(k)}, \]
and \[ \Sigma_y = \Psi' \Sigma_0^{-1} \Psi^H + \beta^{-1} I_N. \] To apply the Block-IBA to block-sparse impulsive noise estimation \( e \) using all tones, we use the same method as in Subsection 5.4.1 and follow an iterative EM algorithm. In the E-step, given the estimate of \( s \) vector, the estimate of amplitude vector \( \theta \) is given by (5.45). In M-step, for the estimate of \( s \) using \( \hat{\theta}^{(k)}_{MAP} \) (the MAP estimate of \( \theta \)) and the observation vector \( y \), the steepest-ascent method is given as
\[
s^{(k+1)} = s^{(k)} + \frac{\mu}{\sigma_\theta^2} g(s) + \frac{\mu}{\sigma_n^2} \text{diag}(F^H (\Psi' \hat{\theta} - y - (Dx)^{(k)})) \cdot \hat{\theta},
\]
where \( g(s) = g_1(s) + g_2(s) \) and the two scalar functions \( g_1(s_i) \) and \( g_2(s_{i+1}) \), \( i = 1, 2, \ldots, N-1 \) are given in (5.37) and (5.38), respectively. In the context of all tones the range for step-size parameter \( \mu \) is also given by (5.39).

Thus far, we have presented the first and the second steps of the Block-IBA, i.e. E-step and M-step given in (5.31) or (5.45), and (5.36) or (5.47), respectively. In the next subsection, we complete the Block-IBA by estimating the unknown parameters of the block-sparse impulsive noise model in Section 5.2.

### 5.4.3 Learning The Impulsive Noise Model Parameters

The block-sparse impulsive noise model \( e \) presented in Section 5.2 is characterized by Markov chain parameters \( p, p_{01} \) and \( p_{10} \) for the supports \( s_i \) of \( e \), and the variance parameter \( \sigma_\theta^2 \) for the amplitudes \( \theta_i \) of \( e \). Also, as described in Subsection 5.4.1, the variance \( \sigma_n^2 \) of the (background) AWGN \( \tilde{z} \) is needed in the E-step and M-step for the estimation of the impulsive noise \( e \). These parameters must be estimated in order to estimate the vector \( e \). For this purpose, we develop estimation algorithms
which work together with Block-IBA to learn the model parameters $p_{01}$, $p_{10}$, $p$, $\sigma_\theta^2$ and $\sigma_n^2$ iteratively from the available data. If we define the parameter vector as $\Omega = (p_{01}, p_{10}, p, \sigma_\theta^2, \sigma_n^2)^T$, we aim to estimate the vector $\Omega$.

To obtain an estimate of $\sigma_\theta$ and $p$, the method of moments estimator is used. To simplify the calculation of $\sigma_\theta$ and $p$, we assume the i.i.d. samples for block-sparse impulsive noise $e$. Hence, neglecting the correlation between the samples $e_i$, we consider the Bernoulli-Gaussian process for these samples as explicitly given in (5.7). It is known from (5.8) that $y_j = \sum_{i=1}^{N} f_{ji} e_i + z_j$. By neglecting the noise power, namely, assuming $\sigma_n \ll \sigma_\theta$, and considering that the samples of impulsive noise $e$ are i.i.d. with zero mean, the statistical moments of impulsive noise and observation samples are related as

$$E(y_j^2) \approx \left( \sum_{i=1}^{N} f_{ji}^2 \right) E(e_i^2),$$

$$E(y_j^4) \approx \left( \sum_{i=1}^{N} f_{ji}^4 \right) E(e_i^4) + (6 \sum_{i,i' = 1, i \neq i'}^{N} f_{ji}^2 f_{ji'}^2) E(e_i^2)^2.$$  

(5.48)  

(5.49)
By assuming that all impulsive noise samples $e_i$ have the same moments and knowing $f_{ji}$, the elements of the partial DFT matrix $F_T$, the moments of impulsive noise samples $e_i$ can be calculated based on (5.48) and (5.49). Moreover, it is observed from (5.7) that the samples of the unknown block-sparse impulsive noise $e$ have the special form of Gaussian mixture (Bernoulli-Gaussian) PDF. Hence, it can be shown that the first two even sample moments of $e$ vector can be given as

$$m_2 = \mathbb{E}(e_i^2) = (1 - p)\sigma^2_\theta,$$  \hspace{1cm} (5.50)

$$m_4 = \mathbb{E}(e_i^4) = 3(1 - p)\sigma^4_\theta,$$  \hspace{1cm} (5.51)

where $\mathbb{E}(e_i^2)$ and $\mathbb{E}(e_i^4)$ are calculated from (5.48) and (5.49), respectively. The solutions of equations (5.50) and (5.51) give the estimate of $\sigma_\theta$ and $p$ as

$$\sigma_\theta = \sqrt{\frac{m_4}{3m_2}}, \quad p = 1 - \frac{3m_2^2}{m_4}.$$  \hspace{1cm} (5.52)

In (5.52), the simple updates of $\sigma_\theta$ and $p$ are used as the initial estimates of the parameters $\sigma_\theta^{(0)}$, $p^{(0)}$, $\sigma_n^{(0)}$, and $\sigma_0^{(0)}$.

In Section 5.3.2.2, we have derived an update in (5.28) for $\beta \triangleq \sigma_n^{-2}$. Moreover, using the MAP estimation method we can express the following update equations for the rest of parameters involved in the iterations

$$p_{10}^{(k+1)} = \frac{\sum_{i=1}^{N-1} s_{i+1} (1 - s_i)}{\sum_{i=1}^{N-1} (1 - s_i)},$$  \hspace{1cm} (5.53)

$$p_{01}^{(k+1)} = \frac{\sum_{i=1}^{N-1} s_i (1 - s_{i+1})}{\sum_{i=1}^{N-1} s_i},$$  \hspace{1cm} (5.54)

$$p^{(k+1)} = \frac{\|s\|_0}{N}.$$  \hspace{1cm} (5.55)
where (5.55) is used to update the parameter $p$ in the iterations. The complete derivation of (5.53) and (5.54) are presented in Appendix F and Appendix B, respectively. The details about the estimation of parameter $p$ in (5.55) is provided in [184].

The same results can be derived for parameter estimation of $\sigma_\theta$ and $p$ as in (5.52) for Block-IBA using all tones estimation. Also, the updates for the rest of the parameters remain unchanged as in (5.44), (5.53), (5.54) and (5.55).

Figure 5.7 shows the proposed Block-IBA receiver, where the block-sparse impulsive noise estimator includes the noise model parameter estimation, support estimation, and the overall block-sparse impulsive noise estimation.

Algorithm 5.1 and Algorithm 5.2 provide the pseudo-code implementations of our proposed Block-IBA using null and all tones, respectively. By numerical study, we empirically find that the initial value for the threshold parameter should be $Th(0) = \sigma_\theta^{(0)}/4$ to achieve reasonable performance. This is because the value of $Th$ specifies the number of nonzero elements in $s$ vector. Extensive simulation results also show that $\sigma_\theta^{(0)} = \sigma_\theta^{(0)}$ results in reasonable performance.

### 5.4.4 Null Tone Placement

As explained in Section 5.3, CS [84] aims to recover the sparse signal accurately from the underdetermined system of linear equations. CS considers the underdetermined problem as follows. Let $w \in \mathbb{C}^m$ represent $K$-sparse signal comprising at most $K$ non-
Algorithm 5.1 The overall Block-IBA estimation using null tones.

Input: \( y_T, F_T, k_{max}, r_{max}, \alpha (0.6 < \alpha < 1), \mu, \) and \( \epsilon (\epsilon < 1) \)

Initialize: \( p^{(0)} = 1 - \frac{3\sigma_2^2}{4k^2}, \sigma_\theta^{(0)} = \sqrt{\frac{\sigma_\theta^2}{2k}} = \sigma_\theta^{(0)}/10, \)
\( Th^{(0)} = \sigma_\theta^{(0)}/4, \sigma_0^{(0)} = \sigma_0^{(0)} , \theta^{(0)} = F_T^H (F_T F_T^H)^{-1} y_T, \)
\( s^{(0)} = (\theta^{(0)} > Th^{(0)}), e^{(0)} = s^{(0)} \odot \theta^{(0)}, \) set difference = 1, \( k = 0. \)

1: while (difference > \( \epsilon \) and \( k < k_{max} \)) do
2: \( \text{E-step: } S^{(k)} = \text{diag}(s^{(k)}), \Psi^{(k)} = F_T S^{(k)}, \)
3: \( \theta^{(k)} = \sigma_\theta^{(k) - 2} \Sigma^{(k-1) \Psi^{(k)H}} Y_T, \)
4: \( \Sigma^{(k)} = \Sigma_0^{(k) - 1} - \Sigma_0^{(k) - 1} \Psi^{(k)H} \left( \sigma_\theta^{(k)} I_N + \Psi^{(k)} \Sigma_0^{(k) - 1} \Psi^{(k)H} \right)^{-1} \times \Psi^{(k)} \Sigma_0^{(k) - 1}. \)
5: \( \text{M-step:} \)
6: \( \text{for } r = 0, \ldots, r_{max} - 1 \) do
7: \( s^{(r+1)} = s^{(r)} + \frac{s^{(r)}}{\sigma_0^{(r)}} g(s^{(r)}) + \frac{s^{(r)}}{\sigma_\theta^{(r)}} \text{diag}(F_T^H (\Psi^{(r)} \theta^{(k)} - Y_T) \cdot \theta^{(k)}), \)
8: \( \sigma_0^{(r+1)} = \alpha \sigma_0^{(r)}, \mu^{(r+1)} = \alpha \mu^{(r)}, \)
9: \( S^{(r)} = \text{diag}(s^{(r)}), \Psi^{(r)} = F_T S^{(r)}. \)
10: end for
11: Decreasing Threshold: \( Th^{(k+1)} = \alpha Th^{(k)}, 0.6 < \alpha < 1 \)
12: Updating supports: \( s^{(k)} = (\theta^{(k)} > Th^{(k)}) \)
13: Parameter Estimation: \( \gamma_i^{(k+1)} = \{ \mu_\gamma^{(k)} \} = \frac{1}{2 \epsilon^{(k+1)}} \), for \( i = 1, 2, \ldots, N, \)
14: \( \Sigma^{(k+1)} = \text{diag}(\gamma_1^{(k+1)}, \gamma_2^{(k+1)}, \ldots, \gamma_N^{(k+1)}). \)
15: \( \sigma_\theta^{(k+1)} = \frac{1}{p^{(k+1)}} \text{ using (5.28),} \)
16: \( p^{(k+1)} = \frac{\|s^{(k)}\|}{\min \left( \sum_{i=1}^{N-1} s_i^{(k)} (1-s_{i+1}^{(k)}) / \sum_{i=1}^{N} (1-s_i^{(k)}), 1 \right)}, \)
17: \( p_0^{(k+1)} = \frac{\sum_{i=1}^{N-1} s_i^{(k)} (1-s_{i+1}^{(k)})}{\sum_{i=1}^{N} s_i^{(k)}}, \)
18: \( e^{(k)} = s^{(k)} \odot \theta^{(k)}. \)
19: Compute difference \( \triangle \|s^{(k+1)} - e^{(k)}\|_2, k \leftarrow k + 1 \)
20: end while

Output: \( \hat{e} = e^{(k)} \)
Algorithm 5.2 The overall Block-IBA estimation using all tones.

Input: $y_T, F_T, y, F, k_{max}, r_{max}, \alpha(0.6 < \alpha < 1), \mu,$ and $\epsilon (\epsilon < 1)$

Initialize: $p(0) = 1 - \frac{3n}{m}, \sigma^{(0)} = \sqrt{\frac{3n}{m}}, \sigma_n^{(0)} = \sigma^{(0)}/10,$

$T h^{(0)} = \sigma^{(0)}/4, \sigma_n^{(0)} = \sigma^{(0)}, \theta^{(0)} = F^H(FF^H)^{-1}y,$

$s^{(0)} = (\theta^{(0)} > Th^{(0)}), e^{(0)} = s^{(0)} \odot \theta^{(0)},$ set difference $= 1, k = 0.$

1: while (difference $> \epsilon$ and $k < k_{max}$) do
2: E-step: $S^{(k)} = \text{diag} (s^{(k)}), \Psi^{(k)} = FS^{(k)},$
3: $(Dx)^{(k+1)} = y_T - F_T \theta^{(k)},$
4: $\theta^{(k)} = (\sigma_n^{-2})^{(k)} \Sigma^{(k)} \Psi^{(k)}^H \left( y - (Dx)^{(k)} \right),$
5: $\Sigma_\theta = \Sigma_0^{(k)} - \Sigma_0^{(k)} \Psi^{(k)} \Sigma_0^{(k)} \Psi^{(k)}^H \left( \sigma_n^{2} I_N + \Psi^{(k)} \Sigma_0^{(k)} \Psi^{(k)}^H \right)^{-1}$

$\times \Psi^{(k)} \Sigma_0^{(k)}.$
6: M-step:
7: for $r = 0, \ldots, r_{max}$ do
8: $s^{(r+1)} = s^{(r)} + \frac{\mu^{(r)}}{\sigma_n} \delta^{(r)} + \frac{\mu^{(r)}}{\sigma_n} \text{diag}(F^H(\Psi^{(r)} \tilde{\theta} - y - (Dx)^{(k)})) \cdot \theta,$
9: $s_0^{(r+1)} = \alpha s_0^{(r)}, \mu^{(r+1)} = \alpha \mu^{(r)}.$
10: $S^{(r)} = \text{diag} (s^{(r)}), \Psi^{(r)} = FS^{(r)}.$
11: end for
12: Decreasing Threshold: $Th^{(k+1)} = \alpha Th^{(k)}, 0.6 < \alpha < 1,$
13: Updating supports: $s^{(k)} = (\theta^{(k)} > Th^{(k)})$
14: Parameter Estimation: $\gamma_i^{(k+1)} = \frac{1}{\beta_i^{(k+1)}} \text{ using } (5.44),$
15: $\Sigma_0^{(k+1)} = \text{diag}(\gamma_1^{(k+1)}, \gamma_2^{(k+1)}, \ldots, \gamma_N^{(k+1)}),$
16: $\sigma_n^{(k+1)} = \frac{1}{\beta_0^{(k+1)}} \text{ using } (5.44),$
17: $p^{(k+1)} = \frac{\sum_{i=1}^{N} s_i^{(k+1)}}{N},\quad P_{01}^{(k+1)} = \frac{\sum_{i=1}^{N-1} s_i^{(k+1)}(1-s_i)}{\sum_{i=1}^{N-1} (1-s_i)},$
18: $\epsilon^{(k)} = s^{(k)} \odot \theta^{(k)}.$
19: Compute difference $\Delta = \frac{\|s^{(k+1)} - e^{(k)}\|}{\|s^{(k+1)}\|}, k \leftarrow k + 1$
20: end while

Output: $\tilde{e} = e^{(k)}$
zero coefficients with $K \ll m$, and let $y \in \mathbb{C}^n$ with $n \ll m$ represent the observation vector expressed by

$$
y = \Phi w + v,
\tag{5.56}
$$

where $\Phi \in \mathbb{C}^{n \times m}$ is a known measurement matrix and $v \in \mathbb{C}^n$ is the Gaussian corrupting noise. Given $y$ and $\Phi$, CS aims to reliably and accurately estimate the original unknown signal $w \in \mathbb{C}^m$. To recover the signal vector $w$ correctly, the measurement matrix $\Phi$ should satisfy the restricted isometry property (RIP) in addition to the sparsity of $w$ [194], [84]. In general, for a given measurement matrix $\Phi$, calculating the RIP is NP-hard [195]. Hence, instead of RIP, the efficiently computable mutual coherence which yields bounds on the RIP is utilized to achieve the recovery guarantee [73]. The mutual coherence of a given matrix $\Phi \in \mathbb{C}^{n \times m}$ is expressed as [73]

$$
\mu(\Phi) = \max_{1 \leq i \neq j \leq m} \frac{|\langle \varphi_i, \varphi_j \rangle|}{\|\varphi_i\|_2 \|\varphi_j\|_2},
\tag{5.57}
$$

where $\varphi_i$ denotes the $i$th column of $\Phi$. It can be shown that $\sqrt{\frac{m-n}{n(m-1)}} \leq \mu(\Phi) \leq 1$ [196]. In addition, $\mu(\Phi) \geq \frac{1}{\sqrt{n}}$ is an efficient approximation for the lower bound when $m \gg n$. To recover the sparse signal $w$ accurately, $\mu(\Phi)$ should be as small as possible (the worst case is 1) [73].

In conventional OFDM systems, the null tones (unused subcarriers) are usually located at the edges of the spectrum to avoid the interference with radio services due to out-of-band emission and inter-carrier interference with adjacent channels. The samples of the block-sparse impulsive noise are non-i.i.d. in the frequency domain. This practice of the placement of the null tones dictates the decoding process without considering the null-tones in the receiver. Taking into account the null tones in the decoding process, namely, controlling their locations, will improve the performance.
of the system in impulsive noise estimation.

The location of the null tones significantly affects the mutual coherence of the measurement matrix $F^T$ in (5.8). This is because the block-sparse impulsive noise estimation is treated as a sparse signal reconstruction. Specifically, the random placement of the null tones achieves sufficiently low coherence matrix $F^T$, i.e. $\mu(F^T)$ is small. This in turn results in more accurate block-sparse impulsive noise estimation. However, randomly placing null tones in conventional OFDM systems may decrease data bandwidth. Hence, the high performance of random placement of null tones in impulsive noise estimation is at the price of data bandwidth. There is therefore a trade off between the performance and data bandwidth which needs to be considered in practice according to application requirements.

5.5 Simulation Results

This section presents experimental results to demonstrate the performance of OFDM receiver based on the Block-IBA estimator for bursty impulsive noise estimation and removal. The numerical results demonstrate that the proposed Block-IBA receiver outperforms the existing techniques at a reasonable computational cost, in both uncoded and coded OFDM systems. Moreover, the empirical evaluation of the effect of null tone placement shows further improvement of the Block-IBA receiver performance.

In all experiments, the null tones are placed at the edges of the channel band. Two types of the noise realizations are used for the experiments. The first one is the block-sparse noise realizations $e$ which are synthetically generated using BGHMM in (5.7) with two-state (first-order) Markov chain process. The second one is the block-sparse noise realizations $e$ which are generated using the more general Gaussian
Hidden Markov model (GHMM) with three-state Markov chain process and the state-transition matrix [178]

\[
T = \begin{bmatrix}
0.989 & 0.006 & 0.005 \\
0.064 & 0.857 & 0.079 \\
0.183 & 0.150 & 0.667
\end{bmatrix}
\]

In this case, the components of the impulsive noise have two non-trivial emission states with powers 20 dB and 30 dB above the background noise power that occur 7% and 3% of the time, respectively.

Unless otherwise stated, in all experiments \( p = 0.9, p_{01} = 0.09 \) which are the parameters of BGHMM [1], [146], [187]. For a fair comparison with GHMM noise model, the impulsive to background noise power ratio (INR) for BGHMM is set to \( \text{INR} = 20 \text{dB} \) which describes the power of impulsive noise \( e \) over AWGN \( z \). Moreover, the signal-to-noise ratio (SNR) is defined as the ratio of the received signal power to the total noise power

\[
\text{SNR (dB)} = 10 \log_{10} \left( \frac{\sigma_x^2}{\sigma_t^2} \right),
\]

where \( \sigma_x^2 \) represents the signal power and \( \sigma_t^2 \) denotes the total noise (i.e., \( n_t \triangleq e + z \)) power. Unless otherwise stated, we choose \( \alpha = 0.98 \) and \( \mu = 10^{-6} \) which are the parameters of Block-IBA. Also, 5 iterations are used for M-step. Since the focus of the proposed Block-IBA is on the block-sparse impulsive noise estimation and cancellation, a frequency-flat channel impulse response is assumed throughout the simulations, i.e. \( |d_{jj}|^2 = 1 \ (j = 1, \cdots, N) \) where \( d_{jj} \) are diagonal elements of matrix \( D \).

In the empirical studies, we compare the proposed Block-IBA receiver with the following receivers.

- SBL, which is a sparse reconstruction receiver based on sparse Bayesian learning
(SBL) proposed in [8]. The receiver uses null tones to estimate and remove impulsive noise. Moreover, SBL receiver jointly estimates impulsive noise and detects OFDM data tones using the information available in all subcarriers.

- MMSE detector, a pre-processing method proposed in [158] which uses MMSE-optimal processing prior to conventional OFDM receiver. MMSE detector assumes the perfect knowledge about BGHMM impulsive noise. In the simulations, it is implemented without perfect knowledge about noise state information (NSI) and is denoted as “PP-MMSE”.

- Block-based CS, a receiver proposed in [76] which uses mixed $\ell_2/\ell_1$ norm-minimization to estimate the support of the block-sparse impulsive noise and least-squares (LS) method to estimate the amplitudes. In the simulation results, it is denoted as “Block CS+LS”.

- BSBL-EM, which is a block-sparse signal reconstruction algorithm from the BSBL framework proposed in [4]. In all the simulations, for BSBL-EM, we set the block size parameter $h = 4$, and divide the signal into equal block size in which the start of each block is known.

- Block-SABMP, which is a block-sparse signal reconstruction algorithm proposed in [197]. Block-SABMP does not require any information about the distribution of the active blocks and the sparsity rate of the block-sparse signal.

As Block-IBA learns the noise model parameters automatically, we investigate the difference between the actual and estimated parameters of the noise. We have used 5 iterations in the M-step of Block-IBA, and presented the evolution of iterative estimation of the actual parameters in Table 5.1. The parameter estimation has been performed for SNR = 10dB and INR = 20dB. It is seen that the parameter
Table 5.1: Iterative estimation of actual parameters in the case of $M = 80$, $N = 256$, $p = 0.9$, $p_{10} = 0.01$, $p_{01} = 0.09$, $\sigma_\theta = 0.3147$, $\sigma_n = 0.03$.

<table>
<thead>
<tr>
<th>itr. #</th>
<th>$p$</th>
<th>$p_{10}$</th>
<th>$p_{01}$</th>
<th>$\sigma_\theta$</th>
<th>$\sigma_n$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.8008</td>
<td>0.1010</td>
<td>0.6458</td>
<td>0.8289</td>
<td>0.0812</td>
</tr>
<tr>
<td>2</td>
<td>0.8402</td>
<td>0.0341</td>
<td>0.3032</td>
<td>0.6160</td>
<td>0.0525</td>
</tr>
<tr>
<td>3</td>
<td>0.9050</td>
<td>0.0225</td>
<td>0.1322</td>
<td>0.4081</td>
<td>0.0430</td>
</tr>
<tr>
<td>4</td>
<td>0.9117</td>
<td>0.0122</td>
<td>0.1016</td>
<td>0.3258</td>
<td>0.0365</td>
</tr>
<tr>
<td>5</td>
<td>0.9117</td>
<td>0.0111</td>
<td>0.1016</td>
<td>0.3004</td>
<td>0.0329</td>
</tr>
</tbody>
</table>

estimation step of Block-IBA shows the accurate results for automatic learning of the parameters.

5.5.1 Performance of Uncoded OFDM System

The uncoded 4-QAM modulated OFDM system comprises $N = 256$ subcarriers per OFDM symbol of which $M = 80$ subcarriers are null tones. Figure 5.8(a) illustrates the symbol-error rate (SER) versus SNR for Block-IBA and the other compared receivers under the block-sparse impulsive noise generated by BGHMM. As seen from Figure 5.8(a), the proposed Block-IBA estimator outperforms the conventional OFDM receiver by almost 4 dB SNR gain. Utilizing all tones in the Block-IBA estimator results in additional 1 dB SNR gain. The Block-IBA estimator using null tones also outperforms the state-of-the-art SBL using null tones by 2.5 dB, BSBL-EM and Block-SABMP by 1 dB, and PP-MMSE receiver by 3 dB in the high SNR regime. It is also interesting to see that Block-IBA estimator using null tones even outperforms the SBL receiver using all tones estimation for most of the SNR range. The reason for the significant performance gain of Block-IBA estimator over most other algorithms is that it utilizes the block structure of the impulsive noise and automatically learns the noise model parameters through the algorithm. In contrast, PP-MMSE ignores the OFDM signal and impulsive noise structure for block-sparse impulsive noise estimation and mitigation; and SBL only uses the sparsity property of
impulsive noise and ignores the block structure of impulsive noise. Block-IBA estimator also outperforms Block CS+LS significantly by almost 3 dB SNR gain. Finally, from Figure 5.8(a), the Block-IBA estimator using all tones outperforms the SBL receiver using all tones by a 1 dB SNR gain.

Figure 5.8(b) shows the SER versus SNR for Block-IBA and the other compared receivers under the block-sparse impulsive noise generated by the GHMM with three-state Markov process [178]. Although Block-IBA estimator is derived from the BGHMM impulsive noise model with two states, it still performs well under the more general impulsive noise and outperforms the other compared algorithms in the high SNR regime when all tones are used for impulsive noise rejection.

It is observed in Figure 5.8 that Block CS+LS does not perform as well as the other algorithms. This is because the other algorithms utilize the a priori statistics of the impulsive noise and additive background noise, while Block CS+LS does not. Also, as pointed out in [8] and also in [76], Block CS+LS can only recover highly sparse impulsive noise, typically between 6 and 12 impulses per OFDM symbol, which is not the case in our simulations.

Although the Block-IBA receiver is designed for block-sparse impulsive noise estimation and removal, for a fair comparison, we also compare the uncoded SER performance of Block-IBA estimator with the other receivers under the i.i.d. impulsive noise. Recall from Section 5.2.2 that with $\eta = 1$ the block-sparse impulsive noise is memoryless and it is equivalent to the Bernoulli-Gaussian (BG) model with i.i.d. samples. Hence, we have set $p_{10} = 0.1$ and $p_{01} = 0.9$ to generate the impulsive noise with i.i.d samples. Figure 5.9 shows the SER versus SNR for our proposed receiver and the other receivers under the i.i.d. impulsive noise. It is observed that the Block-IBA receiver still shows a reasonable performance among the other receivers. This is because the Block-IBA estimator considers the sparsity structure of the impulsive
noise and also considers the block-sparsity of the impulsive noise using the parameters of the Markov chain. Moreover, unlike the other receivers, it automatically learns the parameters of the impulsive noise model. This performance of Block-IBA shows its robustness against the impulsive noise distribution.

5.5.2 Performance of Uncoded OFDM System with Random Placement of Null Tones

This subsection presents the performance of Block-IBA estimator under the effect of null tone placement. To this end, the uncoded 4-QAM 256-tone OFDM system is used under the block-sparse impulsive noise. The number of null tones is 80. Figure 5.10 plots the SER versus SNR for the Block-IBA estimator under sideband (denoted as S) and random (denoted as R) placement of the non-data tones. It is seen that randomizing the non-data tones significantly improves the SER performance of the Block-IBA. A sufficient condition for full recovery of sparse signal from linear measurement is the low coherence of the dictionary $F^T$. This recovery condition requires the randomness in the dictionary $F^T$. Therefore, randomizing the locations of non-data tones yields a signal-free rank-deficient subspace on which the impulsive noise is randomly projected. This random projection of impulsive noise results in low coherence of dictionary $F^T$ that improves the reconstruction performance [198]. The improvement of the performance by randomizing the locations of null tones agrees with the analysis of Subsection 5.4.4 that randomized non-data tone placements leads to smaller value of the coherence $\mu(F^T)$ and enhanced recovery performance.
Figure 5.8: SER vs. SNR for 4-QAM uncoded OFDM system with total 256 subcarriers, and 80 null subcarriers under the block-sparse impulsive noise. (a) Impulsive noise with two-state Markov chain. (b) Impulsive noise with three-state Markov chain.
5.5.3 Performance of Uncoded OFDM System Under Different Parameters of Noise

This subsection investigates the performance of the proposed Block-IBA estimator under different parameters $p$ and $p_{10}$ for BGHMM noise model. Figure 5.11(a) shows the SER versus SNR for Block-IBA and the other compared receivers under the block-sparse impulsive noise generated by the BGHMM with $p = 0.8$ and $p_{10} = 0.01$, which results in more impulses in the channel noise compared to the previous experiments. Figure 5.11(b) shows the SER versus SNR for Block-IBA and the other compared receivers under the block-sparse impulsive noise generated by the BGHMM with $p = 0.9$ and $p_{10} = 0.03$. In this case, the impulses last shorter compared to the previous experiments. It is observed that Block-IBA performs better than the other algorithms in both cases.
5.5.4 Performance of Coded OFDM System

This section presents the bit error rate (BER) performance of Block-IBA estimator in the context of coded OFDM system. To this end, a commonly used convolutional coded 4-QAM OFDM system with total 256 tones and 80 null tones is utilized under the block-sparse impulsive noise. The convolutional code has a code rate of 1/2 and a constraint length of 7 with generator polynomial [171, 133] in octal basis. Figure 5.12(a) shows the BER performance versus SNR for Block-IBA and the other compared receivers in the coded OFDM scenario and under the block-sparse impulsive noise generated by BGHMM. It can be seen that Block-IBA estimator outperforms the other algorithms.

Figure 5.12(b) shows the BER performance versus SNR for Block-IBA and the other compared receivers in the coded OFDM scenario and under the block-sparse impulsive noise generated by the GHMM with three-state Markov chain. It can be seen that Block-IBA estimator outperforms the conventional coded OFDM receiver.
Figure 5.11: SER vs. SNR for 4-QAM uncoded OFDM system with total 256 subcarriers, and 80 null subcarriers in impulsive noise environment. (a) BGHMM impulsive noise with $p = 0.8$ and $p_{10} = 0.01$. (b) BGHMM impulsive noise with $p = 0.9$ and $p_{10} = 0.03$. 
by 5 dB, Block CS+LS receiver by 4 dB, and SBL receiver by 2 dB. By exploiting all tones information, Block-IBA estimator results in further SNR gain of 2 dB over Block-IBA estimator using null tones. Moreover, the Block-IBA estimator using null tones slightly outperforms the BSBL-EM and Block-SABMP estimators.

5.5.5 Complexity

This section compares the computation time of Block-IBA with the other algorithms. Table 5.2 presents the average runtime required to recover 1000 OFDM symbols under the BGHMM noise. The simulations have been performed in MATLAB environment using an Intel 3.10-GHz processor with 8 GB of RAM and under Windows operating system. It can be seen that the computation time of the Block-IBA is comparable to that of SBL algorithm, while it is moderately longer than those of other algorithms. Therefore, the high performance of Block-IBA is at the price of moderately increased computation time compared with the other algorithms.

Table 5.2: Comparison of the mean runtime of the algorithms for 1000 OFDM symbols under BGHMM noise

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Runtime</th>
</tr>
</thead>
<tbody>
<tr>
<td>SBL with null tones</td>
<td>248.3 sec</td>
</tr>
<tr>
<td>PP-MMSE</td>
<td>4.7 sec</td>
</tr>
<tr>
<td>SBL with all tones</td>
<td>748.6 sec</td>
</tr>
<tr>
<td>BSBL-EM</td>
<td>14000 sec</td>
</tr>
<tr>
<td>Block CS+LS</td>
<td>167 sec</td>
</tr>
<tr>
<td>Block-SABMP</td>
<td>159.7 sec</td>
</tr>
<tr>
<td>Block-IBA with null tones</td>
<td>280.7 sec</td>
</tr>
<tr>
<td>Block-IBA with all tones</td>
<td>790 sec</td>
</tr>
</tbody>
</table>

5.6 Conclusion

This chapter has presented a novel receiver to improve the performance of OFDM systems subject to block-sparse impulsive noise. Unlike some other general OFDM
Figure 5.12: BER vs. SNR for 4-QAM coded OFDM system with total 256 subcarriers, and 80 null subcarriers in impulsive noise environment. (a) Impulsive noise with two-state Markov chain. (b) Impulsive noise with three-state Markov chain.
receivers that use TDI to cancel impulsive noise, a specific receiver has been designed here for bursty impulsive noise channels to remove the delay due to TDI and save memory space. The proposed receiver utilizes the Block-IBA algorithm to estimate and remove the block-sparse impulsive noise in the received signal. The estimation is performed using the information carried by the null (and pilot) tones or data tones of the received signal. To reduce the complexity of the Bayesian methods, the proposed Block-IBA uses adaptive thresholding to optimally select the nonzero elements of the block-sparse impulsive noise, and takes advantage of EM and iterative MAP algorithms to estimate the supports and amplitudes of block-sparse impulsive noise. Unlike most of general receivers which assume prior knowledge of the noise parameters, the MAP estimation in Block-IBA automatically learns all the model parameters of block-sparse impulsive noise from the available data. The M-step of the EM algorithm has been optimized by the steepest-ascent method, and an appropriate range has been provided for the step-size parameter of the steepest-ascent to guarantee the convergence of overall Block-IBA. Empirical studies have shown that the proposed receiver outperforms many state-of-the-art receivers in the block-sparse impulsive noise environment. The random placement of the null (and pilot) tones has also been investigated analytically and experimentally. It has been shown that the random placement of null tones improves the performance of OFDM system, by 4 dB SNR gain (at BER of $10^{-3}$), under the block-sparse impulsive noise.
Chapter 6

Conclusion

6.1 Summary of Results

This thesis has proposed a novel block iterative Bayesian algorithm (Block-IBA) for block-sparse signal reconstruction with unknown block structure. It has also proposed a new block Bayesian hypothesis testing algorithm (BBHTA) to reconstruct the structure-agnostic block-sparse signals. Block-IBA utilizes an estimation-based method to recover the signal, while BBHTA uses a joint detection-and-estimation procedure to detect the supports and estimate the amplitudes of block-sparse signals. Then, the Block-IBA algorithm has been leveraged in the design of an OFDM receiver for bursty impulsive noise estimation and cancellation. In particular, a novel OFDM receiver has been proposed, which utilizes the Block-IBA algorithm as an impulsive noise estimator to cancel and reject the block-sparse impulsive noise. The conclusions made from the contributions of this work are as follows:

Chapter 3 has presented a novel Block-IBA to reconstruct the structure-agnostic block-sparse signals. Different to existing algorithms [2], [6], [77], the block structure of the signal has been modeled using Bernoulli-Gaussian hidden Markov model (BGHMM) [1], which better represents the non-i.i.d. block-sparse signals, e.g. the bursty impulsive noise in Power Line Communication (PLC). The proposed Block-IBA uses the iterative maximum \textit{a posteriori} (MAP) estimation of sources and the expectation maximization (EM) algorithm to reduce the complexity of the Bayesian methods. Block-IBA is parametric algorithm, which uses the MAP estimation approach to automatically learn all the signal model parameters from the available data.
A steepest-ascent method has been used to optimize the M-step of the EM algorithm. Moreover, an analytical solution has been provided for the step-size parameter $\mu$ of the steepest-ascent that guarantees the convergence of the overall Block-IBA. Numerical experiments on both synthetic and real-life datasets show the superior performance of Block-IBA to many state-of-the-art block-sparse reconstruction algorithms.

Chapter 4 has presented a novel BBHTA to recover the block-sparse signals whose structure of block sparsity is completely unknown. Different to existing estimation-based algorithms, BBHTA utilizes a joint detection-and-estimation structure to detect the supports and estimate the amplitudes of the unknown signal. The proposed BBHTA uses a Bayesian hypothesis testing (BHT) [192] to detect and recover the supports of the block sparse signal. For amplitude recovery, BBHTA utilizes a linear minimum mean-square error estimate (LMMSEE) to estimate the nonzero amplitudes of the detected supports. BBHTA is a double-looped and turbo-like algorithm. The inner loop is a serial procedure controlling the block sparsity information and detecting the supports of the signal. The outer loop is a turbo-like iterative algorithm, controlling the information related to the amplitudes of the signal using LMMSEE. In fact, the inner loop refines and reuses the LMMSEE of the signal (i.e., $\mathbf{w}$ in (4.1)) by combining the block sparsity information in the successive iterations. As a result of this new implementation, BBHTA offers more reconstruction accuracy for block-sparse signals. Simulation results show that BBHTA outperforms many state-of-the-art algorithms when the block-sparse signal comprises a large number of blocks with short lengths.

Chapter 5 has presented a novel receiver to improve the performance of OFDM systems in bursty impulsive noise channels. Unlike some other general OFDM receivers [8], [9] that use time-domain interleaving (TDI) to cancel the impulsive noise, a specific receiver has been designed for bursty impulsive noise channels to remove
the delay due to TDI and save memory space. The proposed receiver utilizes the Block-IBA algorithm to estimate and remove the block-sparse impulsive noise in the received signal. The estimation is performed using the information carried by the null (and pilot) tones or data tones of the received signal. Unlike most existing receivers [8], [9], [178] which assume prior knowledge of the noise parameters, Block-IBA performs an automatic parameter estimation of block-sparse impulsive noise model from the available data. Empirical studies have shown that the proposed receiver outperforms many state-of-the-art receivers in the block-sparse impulsive noise channels. It has also been shown that the random placement of null tones significantly improves the performance of the OFDM system under the bursty impulsive noise channels.

6.2 Future Work

This section presents some promising areas for future research direction regarding block-sparse signal reconstruction and impulsive noise reduction.

The proposed Block-IBA, in Chapter 3, is based on an estimation method which is a blend of MAP and EM algorithm. It is not a rigorous EM method. By using the BGHMM model parameters and treating some variables as hidden variables and some as deterministic parameters, it may be possible to develop the algorithm in a rigorous EM manner.

The proposed BBHTA, in Chapter 4, is based on the assumption that all the samples $w_i$ satisfy (4.4) with the same variance $\sigma_\theta$. If each $w_i$ has different $\sigma_{\theta_i}$, which is unknown and to be estimated, then the PDF of the signal samples $w_i$ can be modeled by a generalized Bernoulli-Gaussian Mixture (GM) with $M$ components i.e.,

$$p(w_i) = p^{(0)}\delta(w_i) + \sum_{i=1}^{M-1} p^{(i)}\mathcal{N}(w_i; 0, \sigma_{\theta_i}^2),$$

where $\sum_{i=0}^{M-1} p^{(i)} = 1$. This model has been successfully used in [178] to model the impulsive noise in the communication
systems. The parameters $p^{(0)}$, $p^{(i)}$, and $\sigma^2_{\theta_i}$ ($i = 1, \ldots, M - 1$) can be estimated using the expectation-maximization (EM) algorithm. This will be an interesting topic for future work.

In Section 5.5.2, it has been shown that the locations of the null and pilot tones have a significant effect on the communication performance in the bursty impulsive noise channels. The placement of these tones construct an effective measurement matrix for block-sparse impulsive noise recovery. This opens new research directions for known tone allocations in block-sparse impulsive noise channels, compared to the conventional known tone allocations for AWGN channels, which fails to accurately estimate the impulsive noise. In Section 5.4.4, dictionary coherence has been proposed as an appropriate metric to optimize the placement of null and pilot tones. In particular, it has been discussed that the random placement of these tones results in a small value of dictionary coherence, which in turn significantly improves the communication performance under block-sparse impulsive noise environment. However, the discussion lacks the analytical results related to optimality of the dictionary coherence for communication performance. In addition, maximizing this metric is a difficult problem to solve since it is generally a combinatorial problem. Also, the random locations of both null and pilot tones will provide a trade-off between the performance and data bandwidth which needs to be considered in practice according to application requirements. These problems can be addressed in future work.
Bibliography


Appendix A  Derivation of Steepest Ascent Formulation

The first derivative of (3.34) can be written as

$$\frac{\partial L(s)}{\partial s} = \frac{\partial}{\partial s} \log (p(s_1)) + \frac{\partial}{\partial s} \sum_{i=1}^{M-1} \log(p(s_{i+1} | s_i))$$

\[ (A.1) \]

$$- \frac{1}{2\sigma^2} \frac{\partial}{\partial s} (y - \Phi S \theta)^T (y - \Phi S \theta).$$

Define $g(s) \triangleq -\sigma^2_0 \frac{\partial}{\partial s} \log (p(s_1)) - \sigma^2 \frac{\partial}{\partial s} \sum_{i=1}^{M-1} \log(p(s_{i+1} | s_i)) = g_1(s) + g_2(s)$ and $n(s) \triangleq (y - \Phi S \hat{\theta})^T (y - \Phi S \hat{\theta})$, where the two scalar functions $g_1(s_1)$ and $g_2(s_{i+1})$ ($i = 1, 2, \cdots, M - 1$) are given in (3.37) and (3.38), respectively. It can be shown that (see the complete proof in [184])

$$\frac{\partial n(s)}{\partial s} = 2 \cdot \text{diag}(\Phi^T \Phi S \hat{\theta} - \Phi^T y) \cdot \hat{\theta}. \quad (A.2)$$

Therefore using (A.2), (A.1), (3.35) and the definitions of $g(s)$ and $n(s)$, the main steepest-ascent iteration in (3.36) can be obtained.

Appendix B  MAP Update Equation for the Signal Model Parameter $p_{01}$

To calculate the update equation for parameter $p_{01}$, we use the MAP estimation approach, assuming the other parameters are known. Hence, we should maximize the posterior probability $p \left( p_{01} | \hat{s}, \hat{\theta}, \hat{\sigma}, \hat{\sigma}_n, \hat{p}, y \right)$. This probability is equivalent to $p (\hat{s} | p_{01}, p) \times p (y | \hat{s}, \hat{\theta}, \hat{\sigma}_n)$, where only $p (\hat{s} | p_{01}, p)$ depends on $p_{01}$. Therefore, the
MAP estimate of parameter $p_{01}$ can be given as

$$
\hat{p}_{01,MAP} = \arg\max_{p_{01}} p ( s_1 \mid p_{01}, p )
= \arg\max_{p_{01}} p ( s_1 ) \prod_{i=1}^{M-1} p ( s_{i+1} \mid s_i ),
$$

where we have used the equation (3.7). As $p(s_1)$ is independent of $p_{01}$, we can rewrite (B.1) as

$$
\hat{p}_{01,MAP} = \arg\max_{p_{01}} \prod_{i=1}^{M-1} p ( s_{i+1} \mid s_i )
\equiv \arg\max_{p_{01}} \sum_{i=1}^{M-1} \log ( p ( s_{i+1} \mid s_i ) ).
$$

Define $\Gamma \triangleq \sum_{i=1}^{M-1} \log ( p ( s_{i+1} \mid s_i ) )$. Then, differentiating $\Gamma$ with respect to $p_{01}$ and using (3.8) gives

$$
\frac{\partial \Gamma}{\partial p_{01}} = \sum_{i=1}^{M-1} \frac{\partial}{\partial p_{01}} \log ( p ( s_{i+1} \mid s_i ) )
= \sum_{i=1}^{M-1} \left( p_{01}^{-1} s_i (1 - s_{i+1}) - s_i (1 - p_{01})^{-1} s_{i+1} \right).
$$

Equating (B.3) to zero and solving for $p_{01}$ result in the desired MAP update

$$
p_{01}^{(k+1)} = \hat{p}_{01,MAP} = \frac{\sum_{i=1}^{M-1} s_i (1 - s_{i+1})}{\sum_{i=1}^{M-1} s_i}
$$

Appendix C  Proof of Lemma 1

First, from (3.43) it is obvious that the quadratic function $L (w) \triangleq \| y - \Phi w \|^2_2$ is convex. Then, it remains to prove the concavity of $\log (p (w_{i+1} \mid w_i))$ and $\log (p (w_1))$. 
Finally, as the sum of concave functions is concave, the proof is completed. The latent variables $s_i$ has been modeled by first-order Markov chain, thus we have

\[
p(w_{i+1} \mid w_i) = p(s_i = 0) p(w_{i+1} \mid w_i, s_i = 0) \]
\[+ p(s_i = 1) p(w_{i+1} \mid w_i, s_i = 1), \tag{C.1}
\]

where $p(s_i = 0) = p$ and $p(s_i = 1) = 1 - p$. Using the Markov chain model with the transition probabilities $p_{01}$ and $p_{10}$ defined in Section 3.2 we can rewrite the conditional probabilities $p(w_{i+1} \mid w_i, s_i = 0)$ and $p(w_{i+1} \mid w_i, s_i = 1)$ in (C.1), respectively as

\[
p(w_{i+1} \mid w_i, s_i = 0) = \left[ \frac{(1 - p_{10})}{\sigma_1 \sqrt{2\pi}} \exp \left( \frac{-w_{i+1}^2}{2\sigma_1^2} \right) + \frac{p_{10}}{\sigma_2 \sqrt{2\pi}} \exp \left( \frac{-w_{i+1}^2}{2\sigma_2^2} \right) \right], \tag{C.2}
\]

\[
p(w_{i+1} \mid w_i, s_i = 1) = \left[ \frac{p_{01}}{\sigma_1 \sqrt{2\pi}} \exp \left( \frac{-w_{i+1}^2}{2\sigma_1^2} \right) + \frac{(1 - p_{01})}{\sigma_2 \sqrt{2\pi}} \exp \left( \frac{-w_{i+1}^2}{2\sigma_2^2} \right) \right], \tag{C.3}
\]

where $\sigma_1 \ll \sigma_2$. Substituting (C.2) and (C.3) in (C.1) with some simplifications, results in

\[
p(w_{i+1} \mid w_i) = \frac{t_1}{\sigma_1 \sqrt{2\pi}} \exp \left( \frac{-w_{i+1}^2}{2\sigma_1^2} \right) \]
\[+ \frac{t_2}{\sigma_2 \sqrt{2\pi}} \exp \left( \frac{-w_{i+1}^2}{2\sigma_2^2} \right), \tag{C.4}
\]

where $t_1 = p (1 - p_{10}) + p_{01} (1 - p)$ and $t_2 = pp_{10} + (1 - p) (1 - p_{01})$. To prove that $p(w_{i+1} \mid w_i)$ is log-concave, we should prove that the second derivative of $\log(p(w_{i+1} \mid w_i))$ with respect to $w_{i+1}$ is negative, i.e. $h_1(w_{i+1} \mid w_i) \triangleq \frac{p(w_{i+1} \mid w_i) p''(w_{i+1} \mid w_i) - (p'(w_{i+1} \mid w_i))^2}{(p(w_{i+1} \mid w_i))^2} < 0$ or equivalently $h(w_{i+1} \mid w_i) \triangleq p(w_{i+1} \mid w_i) p''(w_{i+1} \mid w_i) - (p'(w_{i+1} \mid w_i))^2 < 0$. 
Calculating the $h(w_{i+1} \mid w_i)$ function results in the following expression

$$
\frac{1}{2\pi} h(w_{i+1} \mid w_i) = -\frac{t_i^2}{\sigma_1^4} \exp\left(-\frac{w_{i+1}^2}{\sigma_1^2}\right) - \frac{t_i^2}{\sigma_2^4} \exp\left(-\frac{w_{i+1}^2}{\sigma_2^2}\right) + \frac{t_i t_2}{\sigma_1 \sigma_2} \left[ \left(\frac{w_{i+1}}{\sigma_1^2} - \frac{w_i}{\sigma_2^2}\right)^2 - \frac{1}{\sigma_1^2} - \frac{1}{\sigma_2^2} \right] \times \exp\left(-\frac{w_{i+1}^2}{2\sigma_1^2} - \frac{w_i^2}{2\sigma_2^2}\right).
$$

(C.5)

If $w_{i+1} = 0$, then $\frac{1}{2\pi} h(0 \mid w_i) = -\frac{t_i^2}{\sigma_1^4} - \frac{t_i t_2}{\sigma_1 \sigma_2} \left(\frac{1}{\sigma_1^2} + \frac{1}{\sigma_2^2}\right) < 0$. If $w_{i+1} \neq 0$, then

$$
\lim_{\sigma_1 \to 0} h(w_{i+1} \mid w_i) = -\frac{t_i^2}{\sigma_2^4} \exp\left(-\frac{w_{i+1}^2}{\sigma_2^2}\right) < 0,
$$

which proves the concavity of $\log\left(p(w_{i+1} \mid w_i)\right)$ for $i = 1, \cdots, M-1$. Moreover, we can rewrite $p(w_1)$ in (3.43) as

$$
p(w_1) = \frac{1 - p}{\sigma_2 \sqrt{2\pi}} \exp\left(-\frac{w_1^2}{2\sigma_2^2}\right),
$$

where $\sigma_1 \ll \sigma_2$. In a very similar way, it can be shown that $p(w_1)$ is log-concave.

### Appendix D Proof of Lemma 2

First we define $d^{(k)} = \frac{\partial \mathcal{L}(s)}{\partial s} \bigg|_{s=s^{(k)}}$, hence we can rewrite (3.35) as

$$
s^{(k+1)} = s^{(k)} + \mu d^{(k)},
$$

(D.1)

where $\mathcal{L}(s) = \sum_{i=0}^{M-1} \log\left(p\left(s_{i+1} \mid s_i\right)\right) - \frac{1}{2\sigma_0^2} (y - \Phi w)^T (y - \Phi w)$ and $p(s_1) = p(s_1 \mid s_0)$ for $i = 0$. We can rewrite $p(s_{i+1} \mid s_i) = \sum_{j=1}^2 \pi_j g_j(s_{i+1} \mid s_i)$, where $g_j(s_{i+1} \mid s_i) = \frac{1}{\sigma_0 \sqrt{2\pi}} \exp\left(-\frac{(s_{i+1} - m_j)^2}{2\sigma_0^2}\right)$, $m_1 = 0$, $m_2 = 1$, $\pi_1 \triangleq [p_{10} + (1 - p_{10})]$ and $\pi_2 \triangleq [p_{10} + (1 - p_{10})]$ for $i = 1, 2, \ldots, M - 1$. Also, for $i = 0 \pi_1 \triangleq p$ and $\pi_2 \triangleq 1 - p$. Since $d = \frac{\partial \mathcal{L}(s)}{\partial s}$, we
can write

\[ d_{i+1} = \frac{\partial}{\partial s_{i+1}} \left( \sum_{i=0}^{M-1} \log \left( \sum_{j=1}^{2} \pi_j g_j \left( s_{i+1} \mid s_i \right) \right) \right) \]

\[ - \frac{1}{2\sigma_n^2} \frac{\partial}{\partial s_{i+1}} \left( (y - \Phi w)^T (y - \Phi w) \right), \tag{D.2} \]

where \( d_{i+1} \) is the \( i + 1 \)th \((i = 0, \ldots, M - 1)\) element of \( d \). We define \( L_1 \triangleq \sum_{i=0}^{M-1} \log \left( \sum_{j=1}^{2} \pi_j g_j \left( s_{i+1} \mid s_i \right) \right) \) and \( L_2 \triangleq (y - \Phi w)^T (y - \Phi w) \), hence we have

\[ \frac{\partial L_1}{\partial s_{i+1}} = -s_{i+1} \sum_{j=1}^{2} \frac{r_j \left( s_{i+1} \mid s_i \right)}{\sigma_0^2} + \frac{r_2 \left( s_{i+1} \mid s_i \right)}{\sigma_0^2}, \tag{D.3} \]

where \( r_j \left( s_{i+1} \mid s_i \right) \triangleq \frac{\pi_j g_j \left( s_{i+1} \mid s_i \right)}{\sum_{j=1}^{2} \pi_j g_j \left( s_{i+1} \mid s_i \right)} \) and \( r_2 \left( s_{i+1} \mid s_i \right) \triangleq \frac{\pi_2 g_2 \left( s_{i+1} \mid s_i \right)}{\sum_{j=1}^{2} \pi_j g_j \left( s_{i+1} \mid s_i \right)} \). Moreover, using the chain rule we can express \( \frac{\partial L_2}{\partial s_{i+1}} = (\partial L_2 / \partial w_{i+1}) (\partial w_{i+1} / \partial s_{i+1}) = \left[ 2\Phi^T (\Phi w - y) \right]_{i+1} \theta_{i+1} \). Substituting this partial derivative and (D.3) into (D.2) results in

\[ d^{(k)} = -z^{(k)} - x^{(k)} + \frac{r_2 (s^{(k)})}{\sigma_0^2}, \tag{D.4} \]

where \( r_2 (s) = [r_2 (s_1), r_2 (s_2 \mid s_1), \ldots, r_2 (s_M \mid s_{M-1})]^T \), \( z_{i+1} \triangleq \left( 1/2\sigma_n^2 \right) \left[ 2\Phi^T (\Phi w - y) \right]_{i+1} \theta_{i+1} \), and \( x_{i+1} \triangleq s_{i+1} \sum_{j=1}^{2} \left( r_j (s_{i+1} \mid s_i) / \sigma_0^2 \right) \). We can also use (3.2) to rewrite \( z \) as

\[ z = \frac{1}{2\sigma_n^2} \Theta_1 (2AS\theta - b), \tag{D.5} \]

where \( \Theta_1 \triangleq \text{diag} (\theta_{i+1}) \), \( A \triangleq \Phi^T \Phi \), and \( b \triangleq 2\Phi^T y \). Hence, by substituting (D.4) into (D.1) the steepest-ascent iteration becomes

\[ s^{(k+1)} = s^{(k)} + \mu \left( -z^{(k)} - x^{(k)} + \frac{r_2 (s^{(k)})}{\sigma_0^2} \right). \tag{D.6} \]

Now, to determine the value of the step-size parameter \( \mu \) that will ensure the
convergence of the iterative process of steepest-ascent in (D.6), we first define an error vector at the kth iteration as
\[ c^{(k)} = s^{(o)} - s^{(k)}, \]  
where \( s^{(o)} \) is the optimum value of \( s \) that maximizes the log-posterior function \( L(s) \).

From (D.7) and (D.6), we have
\[ c^{(k+1)} = c^{(k)} - \mu \left( -z^{(k)} - x^{(k)} + \frac{r_2(s^{(k)})}{\sigma_0^2} \right). \]  

We know that the optimum value \( s^{(o)} \) satisfies the equation\[ d^{(o)} = 0 \quad \text{or} \quad \frac{r_2(s^{(o)})}{\sigma_0^2} = z^{(o)} - x^{(o)}. \] Moreover, \( \lim_{k \to \infty} \frac{r_2(s^{(k)})}{\sigma_0^2} = \frac{r_2(s^{(o)})}{\sigma_0^2} \); therefore, substituting these into (D.8) with some manipulation yields
\[ c^{(k+1)} = c^{(k)} - \mu \left[ (z^{(o)} - z^{(k)}) + (x^{(o)} - x^{(k)}) \right]. \]  

From (D.5), we have \( z^{(o)} - z^{(k)} = \frac{1}{\sigma_n^2} \Theta_1 A \Theta_1 c^{(k)} \). Also, we have \( x = \Theta s \), where \( \Theta = \text{diag} \left( \sum_{j=1}^{2} (r_j s_{i+1} \mid s_i) / \sigma_0^2 \right) \). Hence, we have \( x^{(o)} - x^{(k)} = \Theta s^{(o)} - \Theta s^{(k)} = \Theta c^{(k)}. \)

Substituting these equations into (D.9) with some manipulation leads to
\[ c^{(k+1)} = \left( I - \frac{\mu}{\sigma_n^2} \Theta_1 A \Theta_1 - \mu \Theta \right) c^{(k)} = (I - \mu E) c^{(k)}, \]  
where \( E = \frac{1}{\sigma_n^2} \Theta_1 A \Theta_1 + \Theta \). A necessary condition for iterative procedure in (D.10) to converge is that every eigenvalue of \( D = I - \mu E \) matrix should be less than one in absolute value. If we define the eigenvalues of \( E \) to be \( \lambda_1, \lambda_2, \cdots, \lambda_M \), where \( \lambda \)'s are all greater than zero, then the eigenvalues of \( D \) are found to be equal to \( (1 - \mu \lambda_{i+1}) \), for \( i = 0, 1, \cdots, M-1 \). Now, the necessary condition for the steepest-ascent algorithm to converge to optimum value \( s^{(o)} \) is that \(|1 - \mu \lambda_{i+1}| < 1 \) for all \( i \). That is, the step-size
parameter $\mu$ must satisfy the condition $-1 < 1 - \mu \lambda_{i+1} < 1$ for all $i$ which results in the following interval for step-size parameter $\mu$ as

$$0 < \mu < \frac{2}{\lambda_{\text{max}}(E)}, \quad (D.11)$$

where $\lambda_{\text{max}}(E)$ is the largest eigenvalue of $E$. We know that $\lambda_{i+1}(E) = \frac{1}{\sigma_{\theta}^2} \lambda_{i+1}(\Theta_1 A \Theta_1) + \lambda_{i+1}(\Theta)$, where $\lambda_{i+1}(\Theta) = \frac{1}{\sigma_{\theta}^2}$ because $\Theta$ is diagonal. Hence, the upper bound for $\lambda_{\text{max}}(E)$ is $\lambda_{\text{max}}(E) < \frac{1}{\sigma_{\theta}^2} \lambda_{\text{max}}(\Theta_1 A \Theta_1) + \frac{1}{\sigma_{\theta}^2}$. Therefore, it remains to find an upper bound for the largest eigenvalue of $\Theta_1 A \Theta_1 = (\Phi \Theta_1)^T (\Phi \Theta_1)$ which is a Positive Definite (PD) matrix with all positive eigenvalues. Therefore, we have

$$\lambda_{\text{max}}(\Theta_1 A \Theta_1) \leq \sum_{i=0}^{M-1} \lambda_{i+1} = \text{Tr}(\Theta_1 A \Theta_1) = \text{Tr}(\Theta_1^2 A),$$

where $\Theta_1 \triangleq \text{diag}(\theta_{i+1})$. As the elements of diagonal matrix $\Theta_1$ are chosen from a Gaussian distribution, i.e. $\theta_{i+1} \sim \mathcal{N}(0, \sigma_{\theta}^2)$, we have $p(|\theta_{i+1}| < M_v) = 1 - 2Q(M_v/\sigma_{\theta})$. Hence, when $|\theta_{i+1}| < M_v$, $\text{Tr}(\Theta_1^2 A) = \sum_{i=0}^{M-1} \theta_{i+1}^2 a_{i+1,i+1} < M_v^2 \sum_{i=0}^{M-1} a_{i+1,i+1} = M_v^2 \text{Tr}(\Phi^T \Phi)$ with probability $\gamma_v = (1 - 2Q(M_v/\sigma_{\theta}))^M$ which should be very close to one (e.g., $\gamma_v = 0.99$). Therefore, we must select the value of $M_v$ as $M^* = \sigma_{\theta} Q^{-1}(\frac{1-\frac{M_v}{2}}{\gamma_v})$, where $Q^{-1}(\cdot)$ is the inverse Gaussian Q-function. To ensure the convergence of steepest-ascent algorithm with probability $\gamma_v$ close to one we must choose the interval of the step-size parameter $\mu$ as $0 < \mu < \frac{1}{\sigma_{\theta}^2 + \frac{M^*}{2} \text{Tr}(\Phi^T \Phi)}$. Since the columns of $\Phi$ are normalized to have unit $\ell_2$-norms, we have $\text{Tr}(\Phi^T \Phi) = M$. As a result, the condition (D.11) on step-size parameter $\mu$ can be rewritten as (3.39), which completes the proof of Lemma 2. Consequently, the convergence of M-step of steepest-ascent algorithm is guaranteed.
Appendix E  MAP Update Equation for the Signal Model Parameter $\sigma_\theta$

in (4.26)

To obtain an estimate of $\sigma_\theta$, we use the method of moments estimator. We assume
that the $\Phi$ matrix has the columns with unit $\ell_2$-norms and its elements have a uniform
distribution in [-1,1]. To simplify the calculation of $\sigma_\theta$, we assume the i.i.d. samples
for block-sparse signal $w$. Hence, neglecting the correlation between the samples $w_i$, we
consider the Bernoulli-Gaussian process for these samples as explicitly given in
(4.4). Hence, the second sample moments of $w$ vector can be given as

$$E(w_i^2) = (1 - p)\sigma_\theta^2, \quad (E.1)$$

From (4.1), we know that $y_j = \sum_{i=1}^M \varphi_{ji} w_i + n_j$, and by neglecting the noise power,
only assuming $\sigma_n \ll \sigma_\theta$, we have

$$E(y_j^2) = M E(\varphi_{ji}^2) E(w_i^2). \quad (E.2)$$

Moreover, we know that $\sum_{j=1}^N \varphi_{ji}^2 = 1$, hence $E(\varphi_{ji}^2) = 1/N$. Finally, by substituting
(E.1) into (E.2), we can obtain a simple update for $\sigma_\theta$ as

$$\hat{\sigma}_\theta = \sqrt{\frac{N E(y_j^2)}{M(1 - p)}}. \quad (E.3)$$

Appendix F  MAP Update Equation for the Signal Model Parameter $p_{10}$

To calculate the update equation for parameter $p_{10}$, we need to maximize the posterior
probability $p(p_{10} \mid \mathbf{s}, \hat{\theta}, \hat{\sigma}_\theta, \hat{\sigma}_n, \hat{p}, y)$, i.e. maximizing $p(\mathbf{s} \mid p_{10}, p) \times p(y \mid \mathbf{s}, \hat{\theta}, \hat{\sigma}_n)$, where
only \( p(\hat{s} \mid p_{10}, p) \) depends on \( p_{10} \). From Section 3.2, \( p(s) \) is given by

\[
p(s) = p(s_1) \prod_{i=1}^{M-1} p(s_{i+1} | s_i),
\]

where \( p(s_1) = p^{1-s_1}(1-p)^{s_1} \) and

\[
p(s_{i+1} | s_i) = \begin{cases} (1 - p_{10})^{(1-s_i+1)p_{i+1}^{s_i+1}} & \text{if } s_i = 0, \\ p_{01}^{(1-s_i+1)}(1 - p_{01})^{s_i+1} & \text{if } s_i = 1. \end{cases}
\]

Therefore, the MAP estimate of parameter \( p_{10} \) is given as

\[
\hat{p}_{10MAP} = \arg\max_{p_{10}} p(\hat{s} \mid p_{10}, p) = \arg\max_{p_{10}} p(s_1) \prod_{i=1}^{M-1} p(s_{i+1} | s_i)
\]

\[
\equiv \arg\max_{p_{10}} \sum_{i=1}^{M-1} \log(p(s_{i+1} | s_i)),
\]

where (F.1) is used and \( p(s_1) \) is removed since it is independent of \( p_{10} \). Define \( \Gamma \triangleq \sum_{i=1}^{M-1} \log(p(s_{i+1} | s_i)) \). Then, differentiating \( \Gamma \) with respect to \( p_{10} \) and using (F.2) gives

\[
\frac{\partial \Gamma}{\partial p_{10}} = \sum_{i=1}^{M-1} \frac{\partial}{\partial p_{10}} \log(p(s_{i+1} | s_i))
\]

\[
= \sum_{i=1}^{M-1} (p_{10}^{-1}s_{i+1}(1 - s_i) - (1 - s_{i+1})(1 - p_{10})^{-1}(1 - s_i)).
\]

Equating (F.4) to zero and solving it for \( p_{10} \) result in the desired MAP update

\[
\hat{p}_{10} = \frac{\sum_{i=1}^{M-1} s_{i+1}(1 - s_i)}{\sum_{i=1}^{M-1} (1 - s_i)}.
\]
Appendix G  Derivation of Steepest Ascent Formulation in (5.36)

The first derivative of (5.34) can be written as

$$\frac{\partial L(s)}{\partial s} = \frac{\partial}{\partial s} \log(p(s_1)) + \frac{\partial}{\partial s} \sum_{i=1}^{N-1} \log(p(s_{i+1} | s_i)) - \frac{1}{2\sigma^2_0} \frac{\partial}{\partial s} (y_T - F_T S \hat{\theta})^H (y_T - F_T S \hat{\theta}).$$  \hfill (G.1)

Define $g(s) \triangleq -\sigma^2_0 \frac{\partial}{\partial s} \log(p(s_1)) - \sigma^2_0 \frac{\partial}{\partial s} \sum_{i=1}^{N-1} \log(p(s_{i+1} | s_i)) = g_1(s) + g_2(s)$ and $n(s) \triangleq (y_T - F_T S \hat{\theta})^H (y_T - F_T S \hat{\theta})$, where the two scalar functions $g_1(s)$ and $g_2(s_{i+1})$ ($i = 1, 2, \cdots, N-1$) are given in (5.37) and (5.38), respectively.

It can be shown that (see the complete proof in [184])

$$\frac{\partial n(s)}{\partial s} = 2 \cdot \text{diag}(F^H_T F_T S \hat{\theta} - F^H_T y_T) \cdot \hat{\theta}. \quad \hfill (G.2)$$

Therefore using (G.2), (G.1), (5.35) and the definitions of $g(s)$ and $n(s)$, the main steepest-ascent iteration in (5.36) can be obtained.